Modeling and Addressing Uncertainty in Artificial Intelligence

Contributions to Uncertainty-Aware Machine Learning and Explainable AI

DISSERTATION

zur Erlangung des Grades eines
Doktors der Wirtschaftswissenschaft

eingereicht an der
Fakultät für Wirtschaftswissenschaften
der Universität Regensburg

vorgelegt von
Thomas Krapf, M.Sc. (Mathematik)

Berichterstatter:
Prof. Dr. Bernd Heinrich
Prof. Dr. Günther Pernul

Tag der Disputation: 22.07.2026

Acknowledgements

I would like to thank everyone who has supported me throughout the work on this thesis. First of all, I want to thank Bernd Heinrich and express my deep gratitude to him for his outstanding supervision of my dissertation and exceptional guidance over the past few years. I really appreciate his innovative ideas and impulses, the intensive and fruitful discussions, and the ongoing encouragement he provided, which not only supported this work but also shaped my development and growth as a young researcher.

Second, I would like to thank Günther Pernul for his excellent supervision and the valuable feedback provided during the course of this work.

Further, I would like to thank all the people who accompanied and supported me during my time as a doctoral candidate at the Chair of WI2. I would like to thank my co-authors for the thriving discussions and conversations, their tremendous and sustained efforts, and the productive and supportive working atmosphere. Moreover, I would like to thank all my other colleagues and co-workers at the University of Regensburg for the constructive exchanges and the collaborative environment. In particular, I would like to give special thanks to Armin Steinwender and Heike Gorski, who promptly and reliably helped me with any problems and questions I came up with. Further, I would like to acknowledge Alexander Schiller, who supported and encouraged me from my very first day at the chair and who provided me with countless invaluable pieces of advice throughout the years. In addition, I would like to thank Paul Miethaner, with whom sharing an office was both pleasant and enriching. Also, our coffee breaks often led to interesting discussions and new ideas, while also providing moments of relaxed conversation and good company.

I would like to finish by thanking my family and friends for their continuous support and encouragement. Most notably, I would like to express my heartfelt gratitude towards my parents and my sister for all their love and care. Finally, I would like to thank my wonderful girlfriend Verena for all her love and unwavering support. I am deeply grateful for having you in my life.

Thomas Krapf May 2026

Summary of Contents

Contents

Remark: To facilitate selective reading, each paper in the dissertation is treated as its own manuscript with respect to abbreviations, figures, tables, references, reference style, as well as general numbering and is thus self-contained.

1 Introduction

In this chapter, the motivation for the dissertation is presented. In this context, a theoretical framework in which the contributions of the dissertation are situated is introduced. Thereafter, the focal topics of the dissertation are discussed, along with the research questions that are examined. Finally, the structure of the dissertation is outlined, and an overview of the included papers is provided.

1.1 Motivation

In recent years, rapid technological progress has fundamentally reshaped societies and transformed the way businesses and organizations operate. Advances in digital technologies have altered competitive dynamics and complex decision-making processes across industries and domains. Within this ongoing transformation, artificial intelligence (AI) has emerged as a key driver, enabling new forms of automation, analysis, and decision-making that were previously unattainable. Importantly, this development is far from complete. In fact, there is no sign that the pace of AI-driven innovation will slow down in the foreseeable future. However, as AI systems are increasingly deployed in complex and high-stakes application settings, uncertainty becomes a central and ubiquitous challenge. Uncertainty can arise from multiple sources and can fundamentally affect the validity and transparency of AI-based decisions. Against this background, the aim of this dissertation is to enable the validity and transparency of AI in society and organizations alike by explicitly modeling and addressing uncertainty. In the following, the relevance of AI in the context of the ongoing digital transformation is discussed. Subsequently, a theoretical framework for uncertainty is introduced, before its role as a central challenge for deploying AI in a valid and transparent manner is outlined. Concluding the motivation, the aims and contributions of the dissertation are summarized.

1.1.1 The Era of Artificial Intelligence

AI is a broad and interdisciplinary field that seeks to enable machines to analyze, simulate, explore, and exploit aspects of human cognition and behavior to solve complex tasks (Lu, 2019; Russell & Norvig, 2021; Zha et al., 2025). Due to its groundbreaking successes and the ongoing innovation, AI has become a central focus of research across various disciplines even beyond computer science, such as, e.g., philosophy, economics, mathematics, neuroscience, psychology, information systems, cybernetics, and linguistics (Berente et al., 2021; Russell & Norvig, 2021). Moreover, it is widely adopted across organizational and societal contexts as well as domains in practice such as retail, supply chain, logistics, agriculture, manufacturing, finance, healthcare, the public sector, and many more (Kronblad et al., 2024; Li et al., 2024a; Milana & Ashta, 2021; Razzaq & Shah, 2025). In particular, machine learning (ML), which focuses on algorithms that learn patterns from data, has emerged as a central methodological paradigm of many modern AI systems and provides the foundation of many highly influential and still constantly evolving

fields such as deep learning, natural language processing, and computer vision (Razzaq & Shah, 2025; Russell & Norvig, 2021).

Many tasks that are characterized by large-scale data availability and clearly definable objectives, such as, e.g., information extraction from text, material quality control in production, medical diagnostics, or credit scoring, are highly amenable to automation, and contemporary AI systems have reached or even surpassed human-level performance in such tasks (Cardellicchio et al., 2024; Grauslund, 2022; Jehangir et al., 2023; Li et al., 2024a). These advances offer seminal opportunities but also exert increasing pressure on organizations and businesses to adopt and integrate AI-based technologies in order to remain economically competitive by realizing efficiency gains, cost reductions, and new sources of value creation. Many businesses already leverage AI to enhance their decision-making by extracting insights from large, complex datasets, enabling faster and evidence-based decisions (Jansen et al., 2025; Nuthalapati, 2024; Senoner et al., 2022). Consequently, a profound transformation of the business landscape is already underway, in which AI systems are progressively augmenting or replacing human labor across a wide range of sectors and tasks, with far-reaching implications not only for the organizations themselves with their work practices and employment patterns, but also for society as a whole (Savolainen et al., 2026; Shen & Zhang, 2024).

This development is further reflected by the following recent figures. According to the 2025 Stanford AI index report, AI adoption in business is accelerating rapidly: 78% of surveyed organizations worldwide reported using AI in 2024, a substantial increase from 55% in the previous year (Maslej et al., 2025). Beyond its widespread adoption, empirical evidence also suggests substantial productivity gains associated with the use of AI across various domains. For example, reported improvements include productivity increases of between 21–40% in software engineering (Cui et al., 2024), more than 30% in customer support (Brynjolfsson et al., 2025), and up to 43% in consulting (Dell'Acqua et al., 2023). With already unprecedented investments of roughly $370 billion in 2025, Alphabet, Amazon, Meta, and Microsoft project a further rise of combined capital expenditures to over $650 billion in 2026 (MLQ, 2026). Much of this spending is allocated to AI infrastructure such as data centers, GPUs, servers, and power systems highlighting the central role of computational capacity as a key strategic resource. Alongside the growing attention in businesses, governments also invest at scale. For example, Taiwan has committed approximately $3.2 billion to its "AI Island" initiative, aiming to develop advanced AI hardware, data centers, and computing capabilities to position the country as a global hub for AI (James, 2025). Saudi Arabia announced "Project Transcendence", a $100 billion AI initiative that includes a $5–10 billion partnership with Alphabet to develop Arabic-language AI models (Maslej et al., 2025). Further, the European Commission launched "InvestAI", aiming to mobilize €200 billion in AI investments to strengthen European AI infrastructure and support the development and deployment of advanced AI technologies across the EU (European Commission, 2025).

Overall, this discussion demonstrates that AI has evolved from an experimental technology into a central component of organizational and societal infrastructures. In particular, ML models are

increasingly relied upon in high-stakes decision-making, such as law enforcement (Brayne, 2017; Haley & Burrell, 2025; Raaijmakers, 2019), medical diagnosis (Jussupow et al., 2022; Kim et al., 2023; Wang et al., 2025), or credit scoring (Hurlin et al., 2024; Li et al., 2024a; Visani et al., 2022) where their predictions often have direct consequences on human lives. As a result, the validity and transparency of AI systems have emerged as vital desiderata in such contexts. Among other factors, the validity of ML models and their predictions heavily depends on the quality and (un)certainty of the underlying data used for both training and inference (e.g., Gong et al., 2023; Padmanabhan et al., 2022). On the other hand, transparency is required for ML models and their predictions to be understandable and interpretable, which has critically driven the rise of explainable AI (XAI) in recent years (e.g., Barredo Arrieta et al., 2020; Brasse et al., 2023). Importantly however, XAI methods themselves can be affected by uncertainty, hampering their effectiveness in enhancing transparency. The focal topics of this dissertation directly correspond to the desiderata of validity and transparency in the context of uncertainty and will be examined in detail in Section 1.2. The following section discusses the sources of uncertainty in ML and presents a theoretical framework within which the contributions of this dissertation are situated.

1.1.2 Uncertainty in Machine Learning

In typical ML-based data analyses, an input data instance $x \in X$ is mapped by an ML model $f: X \to Y$ to a target value $f(x) = y \in Y$ which may subsequently be complemented by an explanation $e(x, f)$ generated by an XAI method e. In the context of ML, the term uncertainty commonly refers to a "lack of knowledge about some outcome of interest" (Bhatt et al., 2021, p. 402). Uncertainty can arise from multiple sources and manifest at different stages of the ML-based data analysis pipeline. Accordingly, multiple possible categorizations of uncertainty are considered in literature.

Literature often distinguishes between aleatoric and epistemic uncertainty, which are typically characterized as irreducible and reducible, respectively (Der Kiureghian & Ditlevsen, 2009; Gal, 2016; Hora, 1996; Hüllermeier & Waegeman, 2021). More specifically, aleatoric uncertainty arises from the non-deterministic nature of the dependency between input and output, as well as random noise in the available data (e.g., as a result of measurement imprecision) and cannot be reduced by additional data or observations (Gal, 2016; Hüllermeier & Waegeman, 2021). In contrast, epistemic uncertainty in ML arises from limited knowledge about which model or function best describes the observed data. Importantly, epistemic uncertainty can be reduced, e.g., by collecting additional data or observations, or by using a different ML model type (Bhatt et al., 2021; Gal, 2016). Crucially however, it has been recognized that aleatoric and epistemic uncertainty cannot always be clearly separated as (ir)reducibility is often context-dependent, rendering their boundary inherently blurred (Der Kiureghian & Ditlevsen, 2009; Hüllermeier & Waegeman, 2021).

Therefore, in this dissertation, uncertainty is primarily categorized according to the stage of the ML-based data analysis pipeline at which it manifests (cf., e.g., Förster et al., 2025; Förster et

al., 2026; Kläs & Vollmer, 2018). Chiaburu et al. (2024) propose a framework based on a three-stage ML-based data analysis pipeline where uncertainty can arise at each stage. Förster et al. (2026) suggest a similar framework which additionally incorporates human uncertainty stemming from individual cognitive reasoning, judgment processes, or incomplete domain knowledge by users as a fourth stage. Although human uncertainty is recognized, it is outside the scope of this dissertation which particularly focuses on the first three pipeline stages. In the first stage (input stage), an input data instance $x \in X$ is observed or selected from the given dataset and presented to the ML model f. In the second stage (model stage), the ML model f generates a prediction $f(x)$ for the given input instance x. In the third stage (explanation stage), an XAI method e provides an explanation $e(f, x)$ for the prediction, which can subsequently be presented to a user. Figure 1-1 summarizes and illustrates this three-stage ML-based data analysis pipeline.

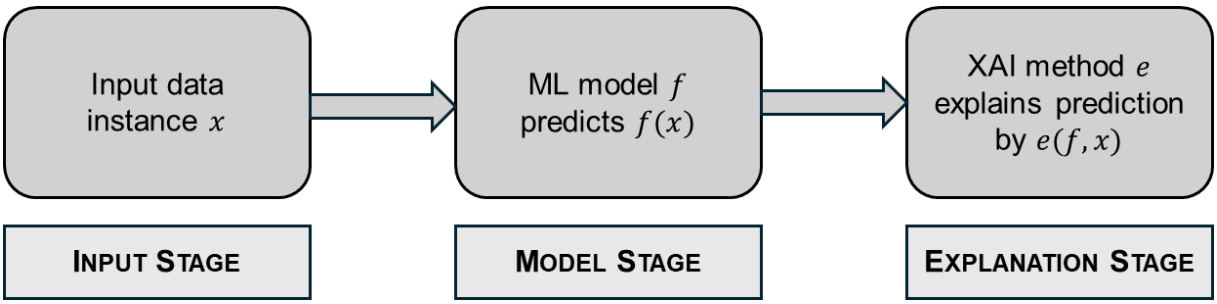


Figure 1-1. Three-Stage ML-based Data Analysis Pipeline (based on Chiaburu et al. (2024))

As uncertainty reflects a lack of knowledge about specific quantities of interest, it is commonly modeled in probabilistic terms (Bhatt et al., 2021; Heinrich & Klier, 2015; Hüllermeier & Waegeman, 2021). Therefore, the deterministic, pointwise formulation of the ML-based data analysis pipeline in Figure 1-1 needs be extended to a probabilistic framework in order to adequately discuss uncertainty in this context. This framework comprises three main sources of uncertainty—data uncertainty, model uncertainty, and explanation uncertainty—which are described in the subsequent sections. Note that although each source of uncertainty is discussed separately in the following, they commonly occur simultaneously and jointly affect the outcomes of the data analysis pipeline.

1.1.2.1 Data Quality and Data Uncertainty

Data uncertainty describes the indefiniteness of data stemming from potential data quality (DQ) defects. Conceptually, DQ characterizes the agreement between the true state of the world and the representing data, and DQ defects arise from disagreements between them (Heinrich et al., 2025; Parssian et al., 2004). On this basis, data uncertainty refers to the lack of knowledge about the true data values resulting from such defects and is subsequently modeled in probabilistic terms. In literature, DQ is commonly understood as a multidimensional concept entailing multiple dimensions that reflect different perspectives on DQ (Batini et al., 2009; Redman, 1996; Wand & Wang, 1996; Wang & Strong, 1996). Among the various DQ dimensions, accuracy, currency, and completeness are frequently examined and are generally regarded as particularly

important (Batini et al., 2009; Cichy & Rass, 2019; Wang & Strong, 1996). In this sense, DQ defects can occur in different forms and thus induce uncertainty through different underlying mechanisms. For example, a potentially outdated data value that indeed was once current introduces temporal uncertainty, as the true value may have drifted over time. The associated probability distribution can therefore be modeled based on the previously observed value and a decline rate that depends on its shelf life or other metadata (e.g. Heinrich & Klier, 2015; Zak & Even, 2017). Similarly, inaccurate (or noisy) data, such as imprecise measurements, are often modeled by Gaussian distributions around the observed value, with the variance reflecting the degree of uncertainty (e.g., Gast & Roth, 2018; Kendall & Gal, 2017). In contrast, regarding the DQ dimension of completeness, missing values introduce uncertainty about the unknown true value and are commonly modeled by a probability distribution estimated from the observed dataset, e.g., based on likelihood-based modeling or multiple imputation (Little & Rubin, 2020).

This discussion highlights how different DQ defects can lead to different forms and modeling of data uncertainty. Instead of a deterministic feature vector, an uncertain instance x is therefore represented by a probability distribution P_X over the input space X, i.e., $x \sim P_X$. For $N_x \in \mathbb{N}$ random samples $x_1, \dots, x_{N_x}$ drawn from P_X representing possible realizations of x and a fixed (i.e., already trained) ML model f, Figure 1-2 illustrates how data uncertainty affects the inference process in ML-based data analyses.

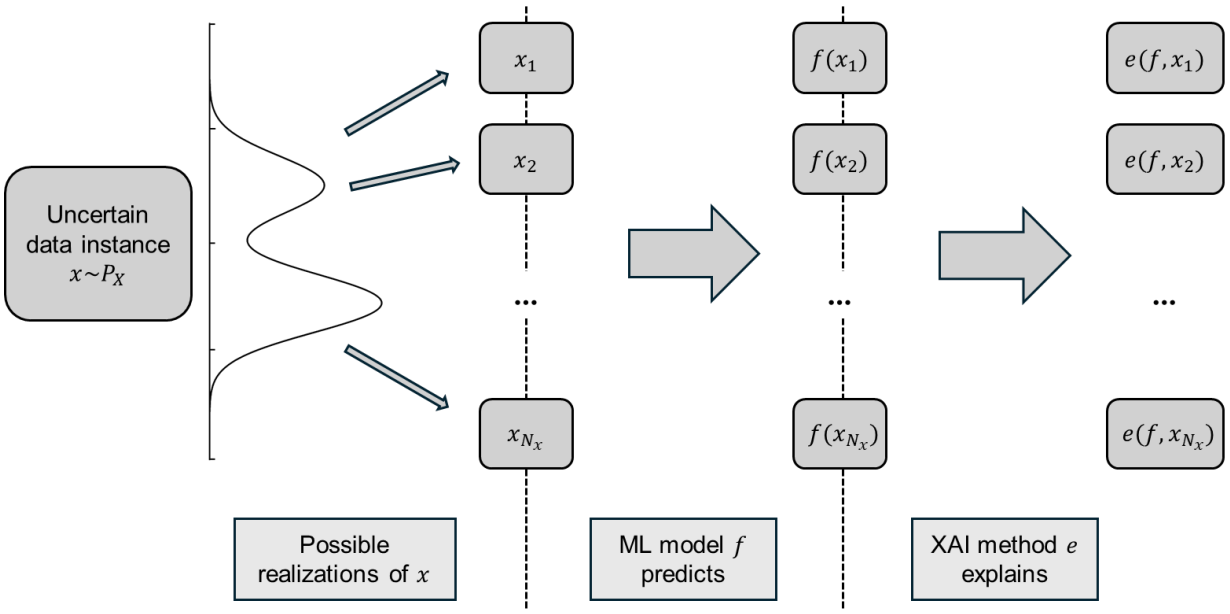


Figure 1-2. Data Uncertainty in ML-based Data Analysis (based on Chiaburu et al. (2024))

Importantly, Figure 1-2 shows how data uncertainty propagates through the subsequent stages of the data analysis pipeline, inducing a variety of possible predictions $f(x_1), \dots, f(x_{N_x})$ and explanations $e(f, x_1), \dots, e(f, x_{N_x})$ for the same uncertain instance $x \sim P_X$. In particular, this can result in incorrect and contradicting model predictions and explanations, which significantly undermines the validity and trustworthiness of the results (Alasalmi et al., 2015; Alvarez-Melis & Jaakkola, 2018; Mohammed et al., 2025). Notably, DQ defects and data uncertainty not only affect inference but also impact model training. Indeed, models trained on noisy, incomplete, or

imputed data can exhibit systematic bias, poor generalizability, and reduced predictive accuracy (Gong et al., 2023; Shadbahr et al., 2023; Song et al., 2023). For these reasons, data uncertainty is a critical factor that must be considered in ML-based data analysis. Hence, addressing data uncertainty in this context constitutes a focal topic of this dissertation, which is further motivated in Section 1.1.3, and the associated research questions are presented in Section 1.2.1.

1.1.2.2 Model Uncertainty

Typically, ML models can only approximate the unknown ground truth relationship between inputs and outputs based on finite training data and may therefore fail to fully capture the true underlying data-generating process. Model uncertainty arises from this lack of knowledge about the perfect prediction model, including uncertainty regarding the appropriate model type, structure, and parameters (Gal, 2016; Hüllermeier & Waegeman, 2021). In principle, this source of uncertainty is often considered reducible through more representative training data or by selecting a model type and capacity that match the complexity of the underlying task, e.g., by selecting a deep convolutional neural network with a high number of learnable parameters for difficult computer vision tasks (Chiaburu et al., 2024; Hüllermeier & Waegeman, 2021; Kendall & Gal, 2017). Nevertheless, some aspects of model uncertainty often remain irreducible in reality: In particular, ML models of all types are subject to inductive biases and structural assumptions that constrain the relationships they can represent (e.g., Goodfellow et al., 2016). Moreover, stochastic elements in the training process, such as parameter initialization or data shuffling can lead to different models with different parameter configurations, even when trained under otherwise identical conditions (Lakshminarayanan et al., 2017).

Consequently, in many ML application scenarios, multiple models can achieve comparable predictive performance on a given dataset while still exhibiting substantially different decision behaviors and predictions on individual instances (Breiman, 2001; D'Amour et al., 2022; Laberge et al., 2023). This widely observed phenomenon is referred to as the Rashomon effect in ML and suggests the existence of a variety of plausible but different models. Formally, this can be represented by a set of (plausible) models $\{f_1, \dots, f_{N_f}\} \subset \mathcal{F}$ with $N_f \in \mathbb{N}$ and $\mathcal{F}$ denoting the set of all possible models $f: X \to Y$ that can be learned from a given dataset $D \subset X \times Y$. As a result, for a deterministic (i.e., no data uncertainty is present) data instance $x \in X$, model uncertainty induces a variety of possible model predictions $f_1(x), \dots, f_{N_f}(x)$ and explanations $e(f_1, x), \dots, e(f_{N_f}, x)$. Critically, these predictions and explanations can be inconsistent or even contradictory, raising fundamental questions about which model, prediction, or explanation to select and trust, thereby posing a serious challenge to valid ML-based decision making (D'Amour et al., 2022; Fisher et al., 2019; Jaki et al., 2025; Laberge et al., 2023). Figure 1-3 illustrates this impact of model uncertainty within the ML-based data analysis pipeline for a deterministic data instance.

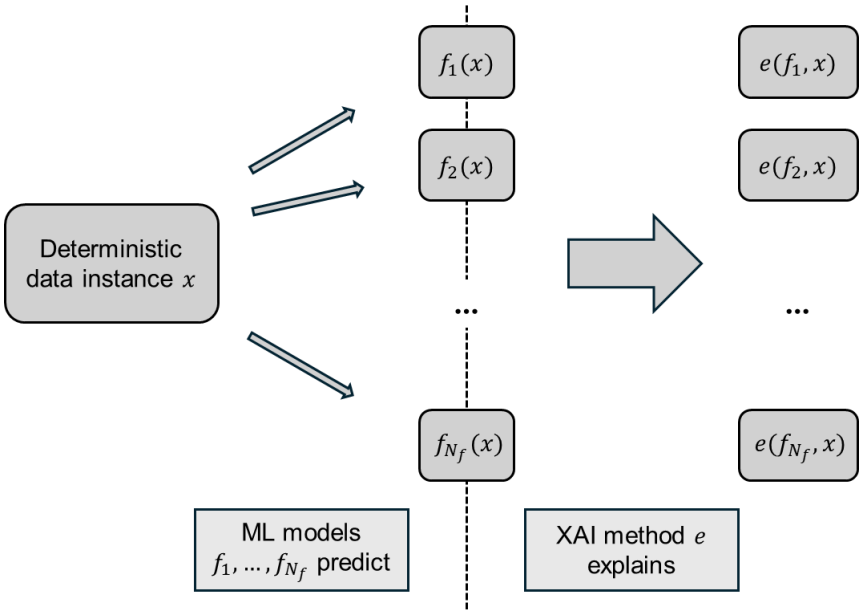


Figure 1-3. Model Uncertainty in ML-based Data Analysis (based on Chiaburu et al. (2024))

1.1.2.3 Explanation Uncertainty

In the ML-based data analysis pipeline, uncertainty can arise not only at the data and model stages but also at the explanation stage when XAI methods are applied (Laugel et al., 2018; Marx et al., 2023; Slack et al., 2021). Explanation uncertainty manifests itself as a variety of possible explanations for the same data instance with respect to the same underlying ML model. It emerges because XAI methods aim to approximate the behavior of complex ML models to convey their predictions in a form that is understandable and interpretable for human users. As a result, the true model behavior is represented only in a simplified form and in many cases cannot be reflected in its full complexity (Jia et al., 2019; Laugel et al., 2018; Petch et al., 2022). Since different XAI methods rely on different approximation strategies, heuristics, and assumptions, they often produce different explanations for the same instance and model (Decker et al., 2024; Krishna et al., 2022; Pirie et al., 2023). For example, the XAI method LIME (Ribeiro et al., 2016) approximates the behavior of the underlying ML model locally by fitting a linear regression as surrogate model, whereas SHAP (Lundberg & Lee, 2017) is grounded in cooperative game theory and assigns each feature an importance value intended to quantify its contribution to the model's prediction relative to a baseline. Hence, different XAI methods $e_1, \dots, e_{N_e}$ ($N_e \in \mathbb{N}$) induce a variety of possible explanations $e_1(f, x), \dots, e_{N_e}(f, x)$. Furthermore, even the same XAI method can yield different explanations across repeated runs due to stochastic elements within the explanation generation process (Visani et al., 2022; Zhou et al., 2021). This particularly concerns surrogate-based XAI methods such as LIME or ROLEX (Kim et al., 2023) which often rely on randomly drawn samples to train their surrogate models. Formally, such a non-deterministic XAI method e_i may yield multiple explanations $e_{i,1}(f, x), \dots, e_{i,N_{e,i}}(f, x)$ when applied $N_{e,i} \in \mathbb{N}$ times to the same instance x and model f. This discussion of explanation uncertainty is summarized and illustrated in Figure 1-4. In this figure, the e_1 exemplarily represents a non-

deterministic XAI method that produces multiple explanations $e_{1,1}(f,x), e_{1,2}(f,x), \dots, e_{1,N_{e,1}}(f,x)$ for the same instance and model.

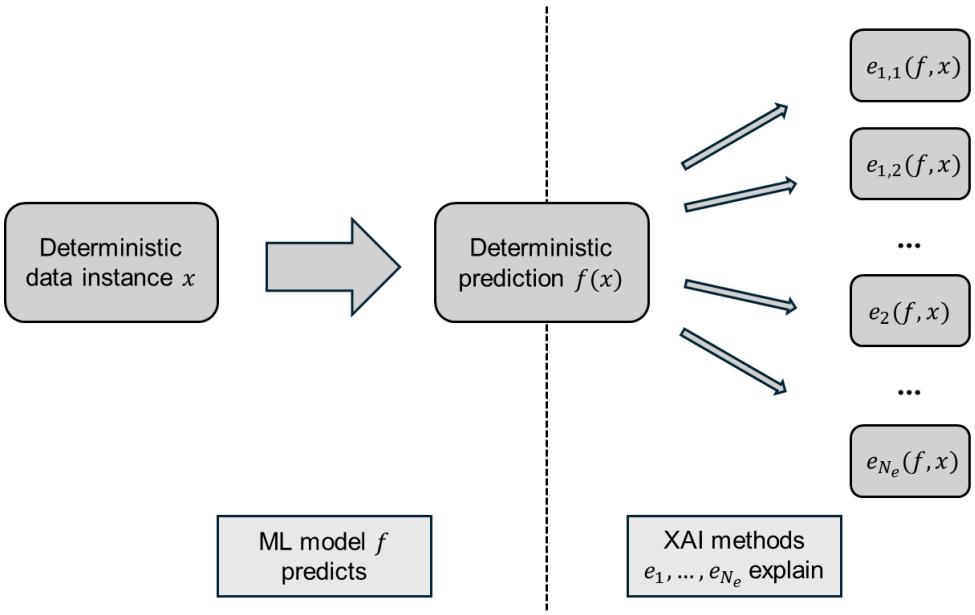


Figure 1-4. Explanation Uncertainty in ML-based Data Analysis (based on Chiaburu et al. (2024))

As a result, explanation uncertainty undermines the ability of explanations to support decision-makers in evaluating ML models and assessing whether individual predictions should be followed or disregarded (Miller, 2019; Saarela & Geogieva, 2022; Visani et al., 2022). Ideally, XAI methods should thus provide robust explanations that correctly reflect the true behavior of the underlying ML model, thereby making them transparent and avoiding ambiguous and misleading information. Against this background, the second focal topic of this dissertation examines the impact of uncertainty across different stages on the explanations provided by XAI methods. The motivation of this focal topic is further elaborated in Section 1.1.4, and the associated research questions are presented in Section 1.2.2.

1.1.3 The Prevalence and Risks of Data Uncertainty

Data uncertainty is a pervasive issue that refers to the indefiniteness of data arising from potential DQ defects. As AI is increasingly employed in data-driven decision-making in various contexts, DQ has been widely recognized as a central determinant for their success. Indeed, poor DQ is a widespread issue which can substantially limit the performance and practical utility of AI systems (e.g., Geiger et al., 2021; Mohammed et al., 2025; Padmanabhan et al., 2022). As a result, data uncertainty and DQ must be considered and addressed in this context. For example, according to a recent report across multiple industries, more than 80% of AI projects fail to move beyond the pilot stage with poor DQ being cited as a primary factor in 68% of the failed cases (Pertama Partners, 2026). However, even when AI systems overcome this initial hurdle—either because they are genuinely valid or because the evaluation process was insufficient and failed to detect issues—and are in active use, they can still suffer from DQ-related problems potentially

leading to severe detrimental consequences. For example, in early 2022, Unity Technologies reported that corrupted datasets used to train advertising-related ML models led to underperforming predictive targeting and bidding algorithms, resulting in an estimated $110 million in lost revenue (Krantz & Jonker, 2026). Importantly however, the consequences of poor DQ can extend beyond mere financial losses but severely impact human lives, particularly when AI is employed in high-stakes domains. For example, a self-driving car failed to recognize a truck due to inaccurately labeled training data in Florida in 2017, resulting in a fatal collision (Sama, 2024).

Aiming to mitigate the negative impact of data uncertainty and poor DQ on AI-driven decision support, and a plethora of methods have been proposed to account for them and address them in ML applications (Gudivada et al., 2017; L'Heureux et al., 2017; Li et al., 2024b). As a result, a broad yet fragmented research landscape has emerged in recent years, encompassing various methods that follow diverse methodological strategies and often even originate from different research strands, each with its own concepts and terminologies. This variety of methods largely reflects the multitude of factors that determine how data uncertainty must be addressed in a given ML application setting, including the ML model used, the type of underlying data, and the specific DQ defects that occur. For example, a method that successfully addresses measurement imprecisions in sensor data, which is analyzed by a random forest for production quality control, is typically not suitable for addressing missing pixel data in the automated assessment of medical scan images by a convolutional neural network. Overall, effectively addressing data uncertainty and DQ defects in ML is highly relevant, but it requires selecting and applying suitable methods for the specific given context to achieve valid ML predictions and reliable decision support despite imperfect data.

As discussed, different DQ defects induce different forms of data uncertainty, which must be modeled and addressed accordingly. For this, a solid theoretical understanding of DQ is essential, providing the foundation for precise definitions and meaningful discussions of DQ defects and other DQ-related phenomena (Haug, 2021; Lee, 2003). Since the early 1990s, multiple efforts for providing theoretical foundations of DQ have been proposed (Aebi & Perrochon, 1993; Kon et al., 1993; Price & Shanks, 2005; Wand & Wang, 1996) and have shaped the trajectory of DQ research. In literature, DQ is often viewed as a multi-dimensional concept comprising a variety of DQ dimensions such as accuracy, currency, completeness, or consistency (Lee et al., 2002; Wang & Strong, 1996). Despite these existing works, research has consistently pointed out that theoretical foundations of DQ and DQ dimensions are still defined in an incoherent or sometimes even inconsistent way (Batini et al., 2009; Guillen-Aguinaga et al., 2025; Haug, 2021; Jayawardene et al., 2013; Miller et al., 2025; Shamala et al., 2017). For example, regarding the DQ dimension currency, Redman (1996, p. 258) defines currency as the "degree to which a datum in question is up to date", Sicari et al. (2016, p. 47) refer to currency as "the interval from the time when the value was sampled to the time instant at which data are received", and Wang and Strong (1996) view currency as a subdimension of timeliness, defining it as the passed time until an information system is updated after a change in the real world. However, without clearly defined DQ fundamentals and dimensions, any efforts to assess and improve, and thus address specific

DQ defects and data uncertainty risk remaining conceptually vague and practically ineffective. Therefore, a coherent and well-founded theoretical grounding of DQ is an indispensable prerequisite for any rigorous discussion of DQ defects and other DQ-related phenomena. In particular, this enables the appropriate modeling of data uncertainty induced by specific DQ defects, and the development of effective methods to address DQ defects and data uncertainty in complex application settings driven by ML.

Another central challenge when addressing data uncertainty in ML is the heterogeneity and increasingly high complexity of ML models, which often necessitate more elaborate and model-specific methods. For instance, due to their groundbreaking success in recent years, neural networks in particular have established themselves as central and indispensable tools for ML-based data analysis across a wide range of domains, including high-stakes areas such as, e.g., healthcare (Kim et al., 2023; Mehbodniya et al., 2022), finance (Hurlin et al., 2024; Mintarya et al., 2023), industrial processes (Jiang et al., 2024; Zhang et al., 2021a), or cybersecurity (Pawlicki et al., 2022; Vigneswaran et al., 2018). As discussed in the previous section, data uncertainty induced by DQ defects translates to uncertainty in the predictions of the model (Gal, 2016; Gast & Roth, 2018; Gawlikowski et al., 2023). Importantly, this predictive uncertainty significantly shapes the information conveyed by model predictions, particularly with respect to their confidence and the probabilities of each possible outcome (Abdelaziz et al., 2015; Gawlikowski et al., 2023; Hüllermeier & Waegeman, 2021). However, due to the complex and opaque nature of neural networks, it is very challenging to determine exactly how data uncertainty propagates through a neural network model and thus, to correctly capture and effectively address it. Therefore, methods that address data uncertainty by exactly examining its propagation through complex ML models such as neural networks are of high interest, as they make it possible to obtain a well-founded decision basis that is required for valid reasoning and decision-making under uncertainty (Berger, 1985; Kochenderfer, 2015).

In addition to the model employed, effectively addressing data uncertainty also depends strongly on the type of data in which it occurs. Notably, data uncertainty and DQ defects are not confined to structured, tabular datasets. Likewise, they frequently arise in unstructured data, which has become a topic of significantly increasing importance in recent years (Hiniduma et al., 2025; Taleb et al., 2018). On the one hand, this is due to the explosive growth of data such as images, videos, audio clips, or textual content driven by, e.g., the proliferation of social media and user-generated content, rich multimedia and sensor data from IoT and autonomous systems, or the digitization of enterprise documentation (Taha, 2025). In fact, the total volume of global data in 2025 is estimated to amount 181 zettabytes of which 80-90% are unstructured (Bartley, 2025; Buller, 2025). On the other hand, the importance of DQ in unstructured data has also been amplified by the rapid development and widespread deployment of AI techniques for analyzing such data, including automated image and video analysis as used in autonomous driving, and in particular the recent breakthroughs in large language models and generative AI. Indeed, especially within organizations, large amounts of information are increasingly maintained and managed as natural language text which, e.g., is increasingly used as the foundation for AI-driven

retrieval-augmented generation (RAG) systems supporting employees and customers alike (Karakurt & Akbulut, 2026; Lewis et al., 2020; Liu & Li, 2026). However, the effectiveness of such systems fundamentally again decisively depends on the quality of the underlying data and—particularly in fast-changing organizational settings—its currency plays a critical role in ensuring valid and up-to-date responses (Müller et al., 2025). Given that (especially organizational) text corpora are often too large for manual maintenance, automated methods to assess and improve the quality of textual data are particularly valuable.

1.1.4 The Opacity of AI and the Need for Explanations

As AI achieves ever greater performance across a wide range of tasks, their growing complexity has made many models increasingly opaque and difficult for human users to interpret (Barredo Arrieta et al., 2020; Brasse et al., 2023; Zschech et al., 2025). This lack of transparency is particularly problematic in high-stakes and safety-critical applications, where trustworthiness and responsible decision-making are essential (Guo et al., 2025; Hurlin et al., 2024; Sunyaev et al., 2025). A well-known incident where the opacity of AI severely affected humans is the Dutch childcare benefit scandal, where thousands of parents were wrongly accused of fraud by the Dutch tax authorities due to discriminative algorithms, raising demands for more transparency of AI in tax-related tasks and for the authorities to be able to explain their decisions (University of Amsterdam, 2023). Another striking example involves a non-transparent AI system used to generate investigative leads from open-source data in US criminal cases. Its unexplainable outputs led prosecutors to withdraw evidence, thus undermining legal proceedings and consuming investigative resources (Stelloh, 2024). Such incidents prompted increased governmental efforts to establish regulatory frameworks and governance mechanisms aimed at ensuring transparency, accountability, and thus the responsible and ethical use of AI. In fact, two major legally binding AI governance frameworks have already been established in Europe: the EU AI Act adopted by the European Union, and the Council of Europe Framework Convention on AI and Human Rights, Democracy and the Rule of Law[1] (Maslej et al., 2025). In particular, both frameworks introduce binding obligations to foster the responsible and trustworthy deployment of AI by imposing transparency requirements, especially in high-risk contexts (Council of Europe, 2024; European Parliament, 2025). Further, although not legally binding, initial agreed-upon frameworks for global AI governance have been proposed through the updated UN Governing AI for Humanity report and the OECD AI Principles—which explicitly include "Transparency and Explainability" as a principle (OECD, 2024)—both providing recommendations and guiding principles for the responsible and trustworthy use of AI worldwide (Maslej et al., 2025).

This growing demand for transparency and explainability has given rise to XAI, a rapidly developing field focused on making AI predictions more interpretable for decision-makers and

[1] The Council of Europe Framework Convention on Artificial Intelligence and Human Rights, Democracy and the Rule of Law is open for signatures by both Council of Europe member and non-member states. The treaty becomes legally binding for countries only after ratification, which is currently ongoing for most signatories (Council of Europe, 2026).

enabling them to effectively understand, trust, and act upon AI outputs (Brasse et al., 2023; Senoner et al., 2022; Zschech et al., 2025). As discussed, this is particularly important in high-stakes or high-risk applications. For example, when a physician relies on AI support for a critical medical diagnosis, it is essential not to blindly accept and apply the results from black-box AI systems. Instead, explaining the underlying reasons for the prediction not only enhances the patient's trust, but it also allows the physician to critically evaluate, verify, and appropriately act upon the recommendation (Kim et al., 2023). As another example, when an AI system for credit scoring recommends denying a credit application, providing the applicant with an explanation is crucial because it can clarify the reasons behind the recommendation and highlight actionable steps for improvement (Visani et al., 2022). Moreover, such an analysis can also reveal potential discrimination or fairness violations if the prediction relies on sensitive features such as race or sex (Hurlin et al., 2024).

Consequently, a variety of XAI methods have been proposed to explain individual predictions of black-box ML models, aiming to make them more transparent. Recognized examples for such local XAI methods (i.e., methods that aim to explain individual data instances and the corresponding ML model predictions) are LIME (Ribeiro et al., 2016), SHAP (Lundberg & Lee, 2017), ROLEX (Kim et al., 2023), and Integrated Gradients (Sundararajan et al., 2017). However, they and many other existing local XAI methods methodologically rely on simplifying assumptions and heuristics such as random sampling, perturbations, or surrogate models to explain ML predictions.

As a result, it has been recognized that applying different XAI methods, each with its own assumptions and heuristics, can lead to multiple, different explanations for the same data instance (Laberge et al., 2023; Marx et al., 2023; Pirie et al., 2023; Slack et al., 2021). This raises not only uncertainty but fundamental concerns about the correctness of XAI methods, which again is critical in high-stakes applications (Huang & Marques-Silva, 2024; Zhou et al., 2022). Indeed, at most one explanation can correctly reflect the actual, true explanation of an instance. Meanwhile, alternative explanations arise from different sources of uncertainty such as explanation uncertainty introduced by the XAI method's inherent explanation generation process, or model uncertainty stemming from the training and selection of the ML model being explained. Existing work addresses such uncertainty either by aggregating multiple explanations (Neuhof & Benjamini, 2024; Pirie et al., 2023; Schulz et al., 2022) or by extracting consistent patterns from them (e.g., Laberge et al., 2023). Therefore, the (in)consistency of explanations rather than their actual correctness is primarily focused. In particular, it remains unclear whether consistent explanations are truly correct or merely converge on a stable yet incorrect explanation. However, this is critical since in many applications explanations are often deemed trustworthy solely because they appear consistent. It is therefore highly relevant to rigorously assess and understand the sources of uncertainty affecting XAI explanations and, on this basis, to examine whether or not explanation consistency can serve as a reliable indicator for explanatory correctness.

Due to their reliance on simplifying assumptions and heuristics, existing XAI methods are inherently subject to explanation uncertainty as they provide only approximations and often cannot

correctly reflect the true behavior of the underlying ML model (Fernández-Loría et al., 2022; Huang & Marques-Silva, 2024; Petch et al., 2022; Salih et al., 2025), thus exhibiting limited fidelity and faithfulness[2] (Agarwal et al., 2022; Nauta et al., 2023; Pawlicki et al., 2024). More precisely, the actual decision boundaries of the ML model are typically not captured by these methods, even though they ultimately determine the model's behavior around a given instance and thus, should decisively shape its explanation (Jia et al., 2019; Ross et al., 2017; Wang et al., 2022). On the one hand, the reliance on heuristics and approximations leads to incorrect explanations (i.e., with limited fidelity or faithfulness) by the XAI methods (Kim et al., 2023; Laugel et al., 2018). On the other hand, the dependence on stochastic elements, such as random samples and perturbations, also compromises their robustness, leading to the non-reproducibility of outcomes for very similar or even the same instances (Alvarez-Melis & Jaakkola, 2018; Saarela & Geogieva, 2022; Visani et al., 2022). Crucially, explanations that suffer from limited fidelity, faithfulness, or robustness not only introduce uncertainty but undermine the main objective of making ML models transparent: They impede a meaningful assessment of model predictions and behavior, especially the correct identification of undesired or detrimental effects, which in turn hinders effective user intervention and ultimately erodes trust in ML-assisted decision-making (Guo et al., 2025; Petch et al., 2022). Therefore, a method that addresses these shortcomings and explicitly incorporates the model's true decision boundaries would substantially advance the field. In particular, such a method would completely eliminate the explanation uncertainty introduced by the XAI method itself, as it would not rely on simplifying assumptions or heuristics but directly on the true behavior of the model.

1.1.5 Aims of the Dissertation of Synthesis of Contributions

The rapid and widespread adoption of AI gives rise to significant challenges, especially regarding uncertainty which impairs the validity and transparency of AI. Against this backdrop, this dissertation aims to advance its two focal topics by modeling and addressing uncertainty in the context of AI. It contributes concepts and methods that address research gaps in existing literature, thereby supporting the valid and transparent use of AI in organizational and societal decision-making contexts. Regarding the first focal topic focusing on addressing data uncertainty, this dissertation initially develops a taxonomy that systematizes and classifies existing methods that address DQ defects in the context of ML. Subsequently, theoretical foundations for DQ are provided to establish a well-grounded conceptual basis for a rigorous discussion of DQ defects and data uncertainty. Finally, a method for the exact propagation of data uncertainty through neural networks, as well as a language model-based method for assessing and improving currency of textual data are proposed, contributing to this research field. With respect to the second focal topic on explaining ML model predictions, the sources of uncertainty affecting explanations

[2] In literature, both fidelity and faithfulness are commonly used criteria for the correctness of explanations. In this dissertation, fidelity is used for the correctness of surrogate models provided by XAI methods and expresses the extent to which the predictions of the surrogate model agree with the predictions of the ML model being explained. Faithfulness is used for the correctness of feature attributions and counterfactual examples and aims at the correct determination of the relevant features and their respective attributions/changes.

are identified, conceptually distinguished, and decomposed. In particular, this allows for a more detailed understanding and assessment of the correctness of explanations. Thereafter, a method for the exact reconstruction of the decision boundaries of ML models is introduced, which enables faithful and robust explanations while eliminating uncertainty in the explanation generation process. All contributions and the corresponding research gaps addressed are discussed in more detail in the following Section 1.2. Figure 1-5 summarizes and illustrates the research endeavor in this dissertation.

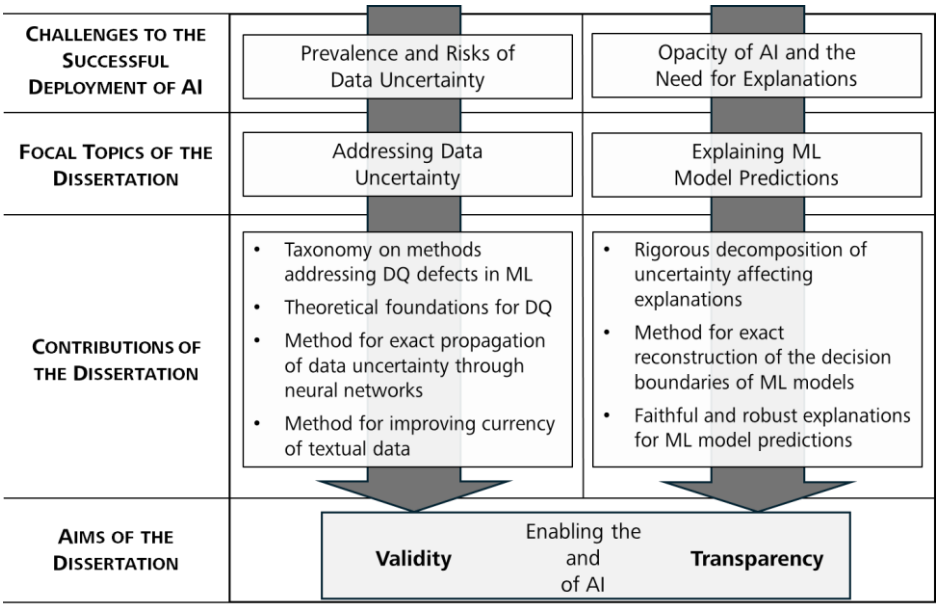


Figure 1-5. Overview of Motivation and Research Endeavor of the Dissertation

1.2 Focal Topics and Research Questions

In this section, the two focal topics of this dissertation are outlined and the research questions addressed by the individual papers are presented.

1.2.1 Focal Topic 1: Addressing Data Uncertainty

The importance of data uncertainty and DQ defects has been recognized in literature and various methods have been proposed to account for them and mitigate their negative effect on ML-based data analysis and decision-making (Côté et al., 2024; Gudivada et al., 2017; L'Heureux et al., 2017; Paleyes et al., 2022). A key consideration for such methods is that different forms of data uncertainty (typically induced by different DQ defects) often need to be treated accordingly. For example, methods addressing data uncertainty resulting from missing values in tabular datasets usually differ substantially from methods that aim to improve the currency of textual data. Moreover, data uncertainty and DQ defects can be addressed at multiple different stages of the ML-based data analysis pipeline. While this is often performed at the input stage through data preprocessing and cleaning techniques (Côté et al., 2024), such as the imputation of missing values (e.g., Little & Rubin, 2020; Peng et al., 2023), it can also be conducted at the model stage, for example through adapted training procedures or uncertainty-aware inference (Abdelaziz et al.,

2015; Kendall & Gal, 2017). Furthermore, data uncertainty may also be addressed at the explanation stage by examining how explanations vary under different realizations of the uncertain input data instance (explanations $e(f, x_1), \ldots, e(f, x_{N_x})$ in Figure 1-2), allowing decision-makers to explicitly consider this variability in subsequent decisions. This dissertation aims to make several contributions to this research field of addressing data uncertainty. The corresponding research questions investigated in this dissertation that contribute to this focal topic are presented in the following.

As already established, mitigating the negative impact of data uncertainty and poor DQ in the context of ML-based data analysis is of utmost importance and a plethora of methods have thus been proposed in literature address them (Côté et al., 2024; Gudivada et al., 2017; L'Heureux et al., 2017; Li et al., 2024b). As the range of ML models, data types, possible DQ defects, and strategies to address them (e.g., at different stages of the ML-based data analysis pipeline) is very broad, research on ML on data with DQ defects has developed into a fragmented landscape. It comprises methods that follow diverse methodological strategies and often even originate from different research strands, each with its own concepts and terminologies. Since accounting for data uncertainty and DQ defects, as well as selecting appropriate methods for addressing them is vital in the context of ML (Gupta et al., 2021; Priestley et al., 2023), this fragmentation poses a significant challenge not only for researchers but also for practitioners seeking to apply ML models to data affected by uncertainty and DQ defects in organizational or other high-stakes contexts. While researchers may struggle to identify research gaps or to clearly position their own methodical contributions to advance this important field, practitioners may face difficulties in searching and identifying suitable existing methods that fit their specific situation and needs. Importantly however, this fragmentation is not limited to methodological differences but also extends to other aspects of the works. For example, it is also difficult to systematically assess and compare the performance of different proposed methods, e.g., on common benchmark datasets. These challenges highlight the need for a taxonomy that systematically classifies and organizes existing research and methods on ML on data with DQ defects according to relevant aspects (i.e., taxonomy dimensions). Therefore, the first research question in this focal topic is:

RQ1: How can existing works dealing with ML on data with DQ defects be systematized and classified with respect to relevant dimensions of a taxonomy?

In this context, well-founded definitions of DQ and DQ defects are essential, as they provide the basis for meaningful modeling of data uncertainty and for the development of methods that address specific DQ defects, which might otherwise remain vaguely defined. Over the last decades, a large body of research on DQ has emerged, including efforts to establish theoretical foundations for DQ (Aebi & Perrochon, 1993; Kon et al., 1993; Price & Shanks, 2005; Wand & Wang, 1996). However, research still points out that fundamental concepts of DQ are still defined in an incoherent way (Haug, 2021; Shamala et al., 2017; Zhang et al., 2019a). Crucially, this also concerns central DQ dimensions such as accuracy and currency. The latter is particularly notable because, as discussed in Section 1.1.3, several incompatible definitions of currency coexist in literature. A major source of ambiguity in this regard is the absence of formal definitions that

precisely capture what is meant by the dimensions themselves (in contrast to their measurement, which may be done by a DQ metric is a subsequent step). Importantly, this issue does not only affect individual DQ dimensions in isolation; accuracy and currency are often recognized as entangled and are similarly perceived (Scannapieco et al., 2005; Zak & Even, 2017). Thus, precise formal specifications of these dimensions would also enable a rigorous examination of this frequently discussed relationship. Overall, such theoretical foundations and specifications of DQ dimensions are necessary to understand how specific DQ defects and forms of data uncertainty arise and to examine their impact, particularly in ML-based decision-making. This lays the basis to assess, improve, and thus address them systematically, especially given that different forms of DQ defects and data uncertainty often require tailored approaches. For the theoretical foundations proposed in this dissertation, an intrinsic perspective, i.e., an "internal view" (Haug, 2025, p. 240, cf. also Lee et al., 2002) on DQ is adopted, that is independent of subjective interpretations of data and the particular context or task at hand (which is focused by extrinsic or contextual DQ, cf., e.g., Strong et al., 1997). Hence, the second research question examined in this focal topic is as follows:

RQ2: How can the central DQ dimensions accuracy and currency be specified in a precise manner based on formal and coherent theoretical foundations for intrinsic DQ?

Due to their remarkable predictive performance and flexibility, neural networks have become a particularly powerful and central model type in ML-based data analysis across a wide range of applications and domains (Tian et al., 2025; Verma et al., 2024a). As discussed, data uncertainty modeled by a probability distribution P_X propagates through a neural network f and induces predictive uncertainty $f(P_X)$ (illustrated by a set of possible realizations $f(x_1), \dots, f(x_{N_x})$ in Figure 1-2). In classification tasks, $f(P_X)$ contains information about the probability of each class and thus provides insight into the prediction confidence under data uncertainty. Consequently, it is essential to determine this predictive uncertainty $f(P_X)$ with high exactness because only then it constitutes a valid basis for subsequent decision-making (Berger, 1985; Kochenderfer, 2015). However, this is very challenging due to the complex nature of neural networks. Existing methods for propagating data uncertainty through neural networks generally follow either a sampling-based approach or make simplifying assumptions about the probability distribution or the layer-wise propagation. Sampling-based methods (e.g., Abdelaziz et al., 2015; Truong, 2021) provide a conceptually straightforward way to propagate arbitrary forms of data uncertainty by pointwise inference but are associated with very high computational cost. In contrast, other methods require the data uncertainty to follow a specific type of probability distribution (often a Gaussian) and use approximations during propagation to reduce computational effort, which however comes at the expense of exactness (Gast & Roth, 2018; Jin et al., 2015; Titensky et al., 2018; Zhang & Shin, 2021). These trade-offs highlight the challenge of propagating uncertainty with high exactness while maintaining computational feasibility and flexibility with respect to the form of data uncertainty. Flexibility is particularly important as it allows the method to address diverse forms of data uncertainty induced by different DQ defects. Against this background, a method is needed that enables the exact and feasible propagation of arbitrary

probability distributions through neural networks, which is the focus of the third research question in this focal topic.

RQ3: How can probability distributions representing uncertainty in input data be propagated through neural networks regarding exactness?

As discussed, currency constitutes a central and widely studied dimension of DQ. Importantly, it is also highly relevant for unstructured data such as natural language text, which is increasingly used in ML-driven applications such as RAG systems, serving as a basis from which large language models can flexibly retrieve relevant information (Karakurt & Akbulut, 2026; Lewis et al., 2020; Saha Roy et al., 2025). However, natural language text data often occur in vast amounts, which makes maintaining currency and high DQ in general particularly difficult and costly, especially when relying on manual curation. This is critical, as data and information quality has been shown to significantly influence user satisfaction and the continued use of information systems (Bhatti et al., 2018; Laumer et al., 2017). A particularly relevant example for such large and dynamic textual corpora are wikis. Wikis provide collaborative, web-based platforms where multiple users can create, edit, and organize information in a shared space, enabling efficient information sharing, version tracking, and collective content development (Beck et al., 2015; Roxburgh et al., 2022). Their flexibility and ease of use have led to widespread adoption across various domains (Bhatti et al., 2018; He & Yang, 2016; Karakurt & Akbulut, 2026). Critically however, wiki data regularly become outdated especially in dynamic organizational settings or rapidly evolving topics in general (Klier et al., 2021). As a result, even large and well-established wikis with many voluntary contributors such as Wikipedia often struggle to keep their information up to date, which highlights the need for automated methods to monitor and maintain the currency of their textual data (Broughton, 2008; Klier et al., 2021; Krempl, 2025). Existing methods addressing the currency of wiki articles are often limited to structured elements of the wiki articles, such as information boxes, and therefore neglect the textual information constituting the majority of wiki data (Lewoniewski, 2017, 2019; Tran & Cao, 2013). Other methods assess the currency of wiki articles as a whole but do not identify which specific (textual) parts of the article have become outdated (Klier et al., 2021; Moestue, 2024). Moreover, these methods do not provide concrete suggestions for improving the currency of the affected data. Therefore, the fourth research question discussed in this focal topic is as follows:

RQ4: How can the assessment and improvement of the currency of textual wiki data be supported in an automated manner?

1.2.2 Focal Topic 2: Explaining Machine Learning Model Predictions

The complexity and opacity of many ML models necessitate explanations to make their predictions transparent and interpretable, especially in high-stakes applications. To this end, existing local XAI methods pursue different strategies, not only in terms of their methodology but also regarding the type of explanations they provide. Literature commonly distinguishes several main types of explanations, including counterfactual explanations, feature attributions, and surrogate models (e.g., Barredo Arrieta et al., 2020; Fernández-Loría et al., 2022). Counterfactual

explanation methods aim to explain model predictions by showing how changes in specific feature values of an instance would alter the prediction outcome of the model. Feature attribution methods assign an importance value to each feature, indicating its respective contribution to the model's prediction. Finally, surrogate model-based methods aim to explain predictions by approximating the complex ML model locally around the considered instance with a simpler, interpretable surrogate model (e.g., a linear regression). Overall, the existing body of XAI research provides a rich set of methods for explaining model predictions from different perspectives.

However, it has been recognized that explanations obtained by XAI methods are inherently subject to uncertainty stemming from different sources (Chiaburu et al., 2024; Förster et al., 2025). Following the theoretical framework introduced in Section 1.1.2, uncertainty can arise at each stage of the ML-based data analysis pipeline, namely data uncertainty, model uncertainty, and explanation uncertainty which also can occur simultaneously and interact. While a few studies in literature adopt a simultaneous perspective on multiple sources of uncertainty (e.g., Förster et al., 2025; Förster et al., 2026; Gosiewska & Biecek, 2019), the majority of works considers one source in isolation, examining its respective effect on explanations (e.g., Alvarez-Melis & Jaakkola, 2018; Fisher et al., 2019; Laberge et al., 2023; Pirie et al., 2023; Schulz et al., 2022). In its second focal topic, this dissertation contributes to explaining ML model predictions, with each contribution focusing on one or multiple sources of uncertainty established in the theoretical framework. While no single contribution addresses all sources simultaneously, this dissertation discusses them within this broader context to provide a more comprehensive perspective on uncertainty in ML-based data analyses (cf. Section 4).

A central question in XAI concerns the identification of the correct explanation for a given data instance (Agarwal et al., 2022; Nauta et al., 2023; Zhou et al., 2022). The notion of a correct explanation is often defined with respect to the given ML model being explained, as XAI methods often are designed to reflect the model's behavior (Jia et al., 2019; Nauta et al., 2023; Ross et al., 2017). In this setting, with a fixed (i.e., certain) data instance x and ML model f, the question of explanation correctness depends entirely on the explanation uncertainty introduced by the applied XAI method(s) (illustrated by the explanations $e_i(f, x)$ in Figure 1-4). However, due to model uncertainty, ML models are typically only approximations of the true underlying data-generating process, i.e., the ground truth, which is thus imperfectly captured. It therefore remains unclear if an explanation $e(f_i, x)$ for a particular model f_i correctly reflects the underlying ground truth, regardless of the applied XAI method e (cf. Figure 1-3). Overall, both model uncertainty and explanation uncertainty contribute to the emergence of multiple, potentially differing explanations. For the important and widely employed explanation type of feature attributions, existing work addresses explanation uncertainty by aggregating multiple explanations for a particular data instance and model (Decker et al., 2024; Neuhof & Benjamini, 2024; Pirie et al., 2023; Schulz et al., 2022). Other works specifically address model uncertainty in feature attribution explanations by extracting consistent patterns (Laberge et al., 2023) or by constructing confidence intervals (Fisher et al., 2019; Marx et al., 2023) from the explanations obtained from different possible ML models. Therefore, prior work primarily focuses on the

(in)consistency of feature attribution explanations rather than their actual correctness with respect to the ground truth. Further, it remains unclear whether consistent explanations are truly correct or merely converge on a structurally biased yet incorrect explanation. In particular, this leaves the underlying reasons (attributable to either model or explanation uncertainty) and their effects on the explanation error largely unexamined. Hence, the first two research questions addressed in this focal topic are as follows:

RQ5: Does consistency across feature attribution explanations serve as a reliable indicator for correctness, or does it mask structural bias regarding the ground truth?

RQ6: How can the error of feature attributions explanations be formally decomposed to isolate the specific effects of explanation and model uncertainty?

Based on this discussion, a key challenge in XAI is to reduce or ideally even eliminate explanation uncertainty (Kim et al., 2023; Zhang et al., 2019b; Zhou et al., 2021). This would be highly valuable, as it would not only reduce explanation errors but also make it possible to analyze the true behavior of opaque black-box ML models, thus making them transparent. However, as argued above, existing XAI methods (e.g., LIME or ROLEX) are inherently subject to explanation uncertainty as they rely on simplifying assumptions, heuristics, and stochastic elements (Jia et al., 2019; Kim et al., 2023; Slack et al., 2020; Visani et al., 2022). In particular, these methods only approximate but do not fully capture the true behavior of the model as defined by its decision boundaries and class subspaces. Given these limitations, explicitly reconstructing the decision boundaries and class subspaces of a ML model in a functional and definite manner[3] paves the way for a variety of rigorous model-related analyses. For instance, one central challenge regarding complex ML models is assessing the sensitivity of their predictions to small input perturbations[4] (König et al., 2024; van Stein et al., 2022; Wang et al., 2021). The goal of sensitivity analyses in this context is to determine whether a model's class prediction for a given instance changes under specific perturbations, thereby revealing whether the instance is vulnerable to malicious changes or data uncertainty, and should thus be treated with caution. Importantly, if all decision boundaries of a model near a given instance are functionally known, the sensitivity of that instance can be rigorously assessed.

Moreover, the transparency obtained by a functional and definite representation of the decision boundaries and class subspaces also enables the functional generation of faithful feature attribution explanations or counterfactual examples which are essentially determined by the decision boundaries of a model (Asrzad et al., 2025; Ross et al., 2017; Verma et al., 2024b; Wang et al., 2022). These explanations do not rely on simplifying assumptions, heuristics, or stochastic elements, and are therefore not subject to the shortcomings of existing XAI methods. Further, they

[3] In this context, the term "functional" is used in the sense that the decision boundaries and class subspaces can be represented based on mathematical formulas which are made explicit and can be directly computed. The term "definite" denotes that this representation is unique, meaning it is the one correct and complete characterization of the decision boundaries and class subspaces for the given ML model.

[4] In the context of neural networks, this task is also referred to as neural network verification in literature.

are not affected by explanation uncertainty, as they definitely and completely reflect the true behavior of the underlying ML model. Taken together, this leads to the following research questions:

RQ7: How can the decision boundaries and class subspaces of a ML model be locally reconstructed in a functional and definite manner, thus making them fully transparent?

RQ8: Based on RQ7, how can the sensitivity of a classification decision be rigorously analyzed?

RQ9: Based on RQ7, how can feature attributions and counterfactual examples be functionally determined to provide local explanations for individual ML model predictions?

1.3 Structure of the Dissertation

This dissertation contains seven research papers which address the research questions raised in Section 1.2. Table 1-1 presents an overview of these papers and groups them regarding the focal topic (FT in Table 1-1) they belong to. Moreover, for each paper, the research questions it addresses, its title, its authors, its outlet, and its current status regarding submission or acceptance are provided.

The remainder of this dissertation is structured as follows. Section 2 presents the first focal topic, *Addressing Data Uncertainty*, comprising four papers. Section 3 focuses on the second focal topic, *Explaining Machine Learning Model Predictions*, and includes three papers. Together, these sections constitute the main contributions of the dissertation. Finally, Section 4 concludes this work by summarizing the main findings and outlining directions for future research.

FT	RQ	Chapter and Title	Authors	Outlet	Current Status
FT1: Addressing Data Uncertainty	RQ1	Chapter 2.1: *Handling Imperfection: A Taxonomy for Machine Learning on Data with Data Quality Defects*	Michael Hagn Bernd Heinrich Thomas Krapf Alexander Schiller	Decision Support Systems (DSS)	This paper is **accepted** and **published** in Volume 196 in the journal *Decision Support Systems* (2025).
	RQ2	Chapter 2.2: *Theoretical Foundations for Data Quality – Disentangling Accuracy and Currency*	Bernd Heinrich Mathias Klier Thomas Krapf Alexander Schiller	ACM Transactions on Management Information Systems (ACM TMIS)	This paper is **under review** for publication in the journal *ACM Transactions on Management Information Systems*.
	RQ3	Chapter 2.3: *Piecewise Linear Transformation – Propagating Aleatoric Uncertainty in Neural Networks*	Thomas Krapf Michael Hagn Paul Miethaner Alexander Schiller Lucas Luttner Bernd Heinrich	Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)	This paper is **accepted** and **published** in the *Proceedings of the 38th AAAI Conference on Artificial Intelligence* (2024).
	RQ4	Chapter 2.4: *The Currency of Wiki Articles – A Language Model-based Approach*	Bernd Heinrich Maximilian Huber Thomas Krapf Alexander Schiller	Proceedings of the International Conference on Information Systems (ICIS)	This paper is **accepted** and **published** in the *Proceedings of the 44th International Conference on Information Systems* (2023).
FT2: Explaining ML Model Predictions	RQ5 RQ6	Chapter 3.1: *Are Consistent Explanations Also Correct? The Impact of Explanation and Model Uncertainty*	Jens Dünisch Bernd Heinrich Thomas Krapf Paul Miethaner	Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)	This paper is **under review** for publication in the *Proceedings of the 2026 ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.
	RQ7 RQ8	Chapter 3.2: *EXPLORE: A Novel Method for Local Explanations*	Bernd Heinrich Thomas Krapf Paul Miethaner	Proceedings of the International Conference on Information Systems (ICIS)	This paper is **accepted** and **published** in the *Proceedings of the 45th International Conference on Information Systems* (2024); winner of the Best Paper in Track Award and Best Design Science Award 1st Runner-Up.
	RQ7 RQ9	Chapter 3.3: *Overcoming the Black Box: Explaining Machine Learning Models with EXPLORE*	Bernd Heinrich Thomas Krapf Paul Miethaner	MIS Quarterly (MISQ)	This paper is **under review** for publication in the journal *MIS Quarterly*.

Table 1-1. Overview of the Papers contained in this Dissertation.

2 Addressing Data Uncertainty

This section contains four research papers concerning the first focal topic on addressing data uncertainty. Section 2.1 presents a taxonomy that systematizes and classifies existing methods that address DQ defects and data uncertainty in the context of ML (RQ1). In Section 2.2, formal and coherent theoretical foundations of DQ are provided that enable a well-grounded and rigorous discussion of DQ defects and data uncertainty related to the DQ dimension accuracy and currency (RQ2). Section 2.3 introduces a novel method that propagates data uncertainty through neural networks regarding exactness (RQ3). Finally, in Section 2.4, a language model-based method for addressing currency-related DQ defects in textual data in wiki articles is proposed (RQ4).

2.1 Paper 1: Handling Imperfection: A Taxonomy for Machine Learning on Data with Data Quality Defects

Current Status	**Full Citation**
This paper is **accepted** and **published** in Volume 196 in the journal *Decision Support Systems* (09/2025)	Hagn, M., Heinrich, B., Krapf, T., & Schiller, A. (2025). Handling imperfection: A taxonomy for machine learning on data with data quality defects. *Decision Support Systems*, *196*, 114493. https://doi.org/10.1016/j.dss.2025.114493

Abstract:

In recent years, machine learning (ML) has become ubiquitous in sectors including transportation, security, health, and finance to analyze large amounts of data and support decision-making. However, real-world datasets used in ML often exhibit various data quality (DQ) defects that can significantly impair the performance and validity of ML models and thus also the decisions derived from them. Therefore, a plethora of methods across various research strands have been proposed to address DQ defects and mitigate their negative impact on ML-based data analysis and decision support. This has resulted in a fragmented research landscape, where comparisons and classifications of methods dealing with ML on data with DQ defects are very challenging for both researchers and practitioners. Thus, based on a structured design process, we develop and present a taxonomy for this research field. The taxonomy serves as a systematic framework to classify and organize existing research and methods according to relevant dimensions and facilitates future work in this area. Its reliability, understandability, completeness, and usefulness are supported by an evaluation with external researchers and practitioners. Finally, we identify current trends and research gaps and derive challenges and directions for future research.

Keywords: Taxonomy, machine learning, data quality, data uncertainty

Please note that the overall formatting of the paper has been adjusted to the remainder of the dissertation.

1 Introduction

In recent years, machine learning (ML) has been adopted in various sectors such as transportation, security, health, and finance to analyze large amounts of data and support decision-making. As an important example, ML is used to assist physicians in critical tasks such as the diagnostic classification of medical images for cancer screenings [1]. In finance, ML facilitates tasks such as algorithmic trading, credit scoring, and fraud detection [2]. In addition, transportation companies such as Uber use ML algorithms to allocate rides, set fares, and recruit drivers [3].

However, the performance and validity of ML models heavily depend on the quality of the data that is used for both training and use [4,5]. In many studies and applications, data quality (DQ) is assumed to be near-perfect; however, real-world datasets often exhibit various DQ defects [6] which can significantly affect the performance and validity of ML models, and thus the decisions derived from them when applied in practice [7]. Several incidents have been traced back to this issue in recent years, some with severe consequences such as the collision between a self-driving car and a truck in Florida in 2017. As a result of inaccurate data labeling during its training process, the ML model failed to recognize the truck as an obstacle, resulting in a fatal accident [8].

DQ defects can arise from various causes: For example, measurement errors lead to noisy or inaccurate data [9], defect sensors [10] or incomplete survey responses [11] can cause missing data, previously accurate data values can become outdated over time [12,13], or privacy concerns may result in distributed, incomplete data [14]. Furthermore, the same data may be affected by multiple causes simultaneously. Importantly, such data is not only defect in the sense that, e.g., certain data values are inaccurate or missing, but it is also uncertain since the true underlying values are unknown and can, e.g., be modeled using probability distributions. As a result of such uncertain data values or instances, the predictions of ML models trained on or applied to these data instances with DQ defects are also subject to predictive uncertainty. Predictive uncertainty describes the uncertainty of the model prediction itself. For example, when a classification model is used to classify an uncertain data instance, the resulting class prediction is not deterministic, but, e.g., a probability distribution over the set of possible classes [15]. Therefore, when ML models are trained on or applied to defect data, the impact of the defect data must be considered when supporting decisions based on their uncertain predictions.

Such issues have been recognized in literature and various methods have been proposed to account for DQ defects and mitigate their negative impact on ML-based data analysis and decision support. The range of possible strategies for such methods is very broad as, e.g., DQ defects could already be targeted before the ML model is trained or applied (e.g., by an imputation method in the case of missing data values), or the training procedure of an ML model could be adjusted to handle a specific DQ defect (e.g., by data label correction during model training). Due to this abundance of proposed methods across various research strands, a fragmented research landscape has emerged in which comparisons and classifications of the methods tackling ML on data with DQ defects are very challenging. Furthermore, this is not only the case from a

methodological point of view; a valid assessment and comparison of the performance of the different proposed methods is also hardly possible, since the methods are often evaluated on (very) customized datasets (due to a lack of benchmark datasets with labeled DQ defects) [16]. A further complicating factor is the terminology and concepts used across papers, which—although describing the same or similar ideas—vary depending on the research strand. These challenges underscore the need for a framework to classify and organize existing research and methods according to relevant dimensions describing the field of ML on data with DQ defects.

Related studies [17,18] have already reviewed specific subsets of DQ defect types and associated DQ handling methods (see Section 2.2 on Related Work) and thus address selected parts and aspects of DQ and its role in ML applications. However, to the best of our knowledge, there is no comprehensive taxonomy comprising relevant dimensions of ML on data with DQ defects. Therefore, we aim to close this gap by addressing the research question: *How can existing works dealing with ML on data with DQ defects be systematized and classified with respect to relevant dimensions of a taxonomy?*

We present a comprehensive taxonomy for the research field of ML on data with DQ defects. Our aim is not to provide an exhaustive, fully comprehensive overview or survey on all existing literature in the field or to detail specific methods, but rather to provide a structured framework for this important area of research. Our main contributions can be summarized as follows:

- Following an established taxonomy design methodology, we develop a comprehensive taxonomy for the field of ML on data with DQ defects, enabling the classification of DQ defects and strategies for addressing these DQ defects in ML-based data analysis with respect to relevant dimensions.
- Based on this taxonomy and the literature we analyzed during the taxonomy building process, we identify current trends and research gaps in this ever-evolving research field and deduce challenges and directions for future research.

The remainder of the paper is structured as follows: Section 2 provides background on DQ and its manifold influence on ML before reviewing related studies on the role of DQ defects in ML in detail. Section 3 outlines the general taxonomy design process proposed by Kundisch et al. [19] which we follow to develop our taxonomy. In Section 4, the final taxonomy for the field of ML on data with DQ defects is presented in detail. In Section 5, we evaluate our taxonomy and present the results. We conclude with a discussion of our contributions to research and practice, and potential starting points for future research.

2 Background and Related Work

2.1 Background

DQ has been the subject of extensive research, yielding important theoretical and practical insights. Various works have established theoretical foundations for DQ and DQ dimensions [20,21]. Building on these works, general frameworks and methodologies for DQ have been

introduced [22,23]. It has been established that DQ can be characterized and analyzed with respect to multiple DQ dimensions [21] such as completeness, accuracy, consistency and currency. Based on this, research has proposed methods to identify and measure DQ defects of various DQ dimensions with the aim of managing DQ and supporting decision-making, e.g., by designing metrics for the consistency [24] and currency [12,13] of data in order to identify data of poor quality before their further use. Going beyond both the identification of DQ defects and the measurement of DQ, multiple works have introduced data cleaning methods and frameworks to address and resolve DQ defects [25]. Others have examined the causes of DQ defects in the data collection process and have proposed ideas for curating DQ by improving this process, e.g., by adding incentives for higher DQ [26,27]. In addition to DQ defects in data values, research has also addressed DQ defects in data labels by providing methods for handling inaccurate or missing labels, for example by a confident learning methodology [28].

Literature has also discussed the variety of ways in which DQ influences ML models trained on or applied to data with DQ defects. It has been established that DQ defects usually negatively affect model performance and validity [4,5] and induce predictive uncertainty [15]. In contrast, addressing DQ defects may have negative effects as well, e.g., if the imputation of missing values introduces bias into the data which then affects ML model training or use. In addition, model fairness, i.e., the absence of bias regarding, e.g., discrimination or social inequalities [29] can also be influenced by DQ [30]. Indeed, research has shown that, depending on how DQ defects are addressed, improving DQ can actually be detrimental to fairness. For example, Guha et al. [31] demonstrate that automated data cleaning techniques may negatively impact fairness in ML models and are even more likely to worsen fairness than to improve it. Another aspect related to DQ is the explainability of ML models, with DQ defects potentially influencing not only the ML model itself, but also the validity of explanations generated for it [32]. Moreover, DQ also is relevant for privacy-enhancing techniques in ML such as federated learning, where data with poor DQ needs to be identified to improve performance [33]. Given these various ways in which DQ influences ML models, we focus on the training and use of ML models on data with DQ defects in our work. In this setting, we analyzed research works proposing methods which aim to address, and thus resolve or mitigate problems caused by DQ defects in ML-based data analysis in an application context such as finance or health (see Section 3.2 for further details on our focus). In particular, we do not consider stages before data preprocessing (e.g., the data collection) or stages beyond the output of the ML model (e.g., the interpretation and adoption of the model predictions by humans). We also do not consider cases in which new DQ defects may influence ML models after the training process of the model is completed, e.g., when previously used data is reclaimed due to privacy concerns, resulting in the need for machine unlearning [14].

2.2 Related Work

In this section, we present related studies aiming to organize the field of ML on data with DQ defects. Importantly, we do not focus on works that just provide an overview of DQ without ML context.

2.1 Paper 1: Handling Imperfection: A Taxonomy for Machine Learning on Data with Data Quality Defects

A first category of works discusses DQ and its role in ML-based data analysis at a more general level as one of several important challenges in the field of ML. L'Heureux et al. [34] outline ML challenges arising from big data properties such as the volume, velocity, variety, and veracity of big data, which inherently include a number of DQ defects resulting from these properties. Paleyes et al. [35] examine DQ issues appearing during the process of data collection and data (pre)processing for ML- based analysis. Other works [36,37] provide an overview of ML methods for learning from class-imbalanced data, which is partly the result of DQ defects such as inaccurate or missing data. None of these works, however, aim to provide a comprehensive overview of ML on data with DQ defects. Moreover, they do not focus on addressing these DQ defects in ML-based data analysis in the application context such as health, especially in the training, use, and output of the ML model.

A second category of works specifically focuses on overviews of DQ defects and their influence on ML training and use. Gong et al. [4] discuss DQ and its impact on ML models and give an overview of DQ dimensions and DQ metrics. Mohammed et al. [7] empirically examine how a variety of DQ defects affect ML model performance. Priestley et al. [38] also analyze DQ in different stages of an ML-based data analysis process and list relevant DQ issues in each stage of the process. Gudivada et al. [39] discuss a selection of DQ defects such as missing, heterogeneous, and duplicate data in the context of ML and present a selection of available DQ management tools. Similarly, Li et al. [18] review common DQ issues before focusing on the potential of Active Learning-based approaches for handling DQ defects. However, all of these works only discuss specific aspects of DQ defects and their handling in ML applications. They in particular ignore the possibility of addressing DQ defects by modifications of the ML model itself. Furthermore, they do not aim to organize and systematize the relevant dimensions of the research field of ML on data with DQ defects.

A third category of works includes studies that incorporate the possibility of addressing DQ defects by modifying the ML model. Das et al. [17] present an overview of various distribution-based and feature- based data irregularities and discuss selected approaches to improve classifier robustness against such irregularities. However, they limit their scope to four types of data irregularities, excluding DQ defects such as inaccurate or outdated data. Gawlikowski et al. [40] review methods to handle DQ defects by incorporating DQ-related (data) uncertainty into different types of neural network (NN) models. Similarly, Wang et al. [41] discuss techniques for dealing with various forms of uncertainty, including data uncertainty caused by DQ defects in graph NNs. Yet, none of them examine DQ defects in detail or cover ML models beyond NNs and graph NNs, respectively. Therefore, these works also do not provide a comprehensive systematization of the field of ML on data with DQ defects.

In summary, while some studies and overviews address selected aspects of DQ defects in ML-based data analysis, a thorough systemization of this research field is still missing. In particular, there is no comprehensive taxonomy of ML on data with DQ defects that allows researchers and practitioners to organize and systematize approaches and methods in the field. As a result,

previous discussions of future research agendas are also limited to the selected DQ aspects covered in existing works. We aim to close these gaps by addressing our research question.

3 Methodology

3.1 Research Method

Step	Description	
1	The observed phenomenon is defined.	
2	The target user group(s) are specified.	
3	The intended purposes of the taxonomy are specified.	
4	The meta-characteristic of the taxonomy is formulated. It determines the perspective on the observed phenomenon (delimiting non-considered phenomena). All dimensions and characteristics of the taxonomy must relate to the meta-characteristic.	
5	Ending conditions for the (iterative) taxonomy building process and evaluation goals are defined.	
	While ending conditions are not satisfied	
6	The approach for the development step of the taxonomy is chosen: Empirical-to-conceptual	Conceptual-to-empirical
7	Real-life objects (e.g., papers addressing methods for ML on data with DQ defects) are identified.	Taxonomy dimensions and characteristics are conceptualized by the authors.
8	Taxonomy dimensions and characteristics are recognized based on the identified real-life objects.	Real-life objects are examined regarding the conceptualized dimensions and characteristics.
9	Taxonomy dimensions and characteristics are grouped by the authors.	[This step is skipped for conceptual-to-empirical iterations.]
10	The state of the taxonomy is created (in the first iteration) or revised (in later iterations) by applying taxonomy operations such as adding, updating, or deleting (sub)dimensions and characteristics.	
11-14	It is tested whether the taxonomy satisfies the defined ending conditions: Yes	No
	The process continues with Step 15.	A new iteration starting at Step 6 is performed.
15	The taxonomy evaluation is conceptualized and operationalized.	
16	The taxonomy evaluation is executed.	
17	Based on the evaluation, it is assessed how well the taxonomy fulfills its intended goals for the identified target user group(s) (cf. Step 2).	
18	The taxonomy building process and the final taxonomy are reported in an appropriate manner.	

Table 1. Extended Taxonomy Design Process (ETDP)

We now describe the extended taxonomy design process (ETDP) by Kundisch et al. [19] which we follow for the development of our taxonomy. A taxonomy is a classification system that helps with the conceptualization of objects (i.e., in our case, papers) [19]. It comprises a number of dimensions and a set of characteristics for each dimension. It is also possible for a dimension to contain multiple subdimensions, each with a set of characteristics, and the dimension can then

be seen as a group of subdimensions. When an object is classified according to the taxonomy, one or multiple characteristics are selected for each dimension or subdimension respectively. The ETDP is based on the well- established taxonomy design process by Nickerson et al. [42] that is widely used in research fields such as computer science, information systems, and finance. Its iterative nature enables taxonomy designers to update, extend and refine taxonomies without the need to repeat the entire taxonomy design process from scratch. We also follow the 26 taxonomy design recommendations of Kundisch et al. [19] (in addition to the ETDP) with operational support and examples of good practice. The 18 steps of the ETDP for designing and evaluating the taxonomy are summarized in Table 1.

Kundisch et al. [19] recommend that at least one empirical-to- conceptual step and at least one conceptual-to-empirical step should be conducted. In the following Section 3.2, we describe the first 14 steps of the ETDP for our taxonomy for the field of ML on data with DQ defects. In Section 4, the final taxonomy is presented and communicated (Step 18). Finally, we discuss the conceptualization and execution of the taxonomy evaluation (Steps 15–17) in Section 5.

3.2 Taxonomy Design Approach

Following the ETDP as described in the previous section, we began by defining the observed phenomenon (Step 1). We focused on the training and use of ML models on data with DQ defects. In this setting, we analyzed research works proposing methods which aim to address, and thus resolve or mitigate problems caused by DQ defects in ML-based data analysis in an application context such as finance or health. To define and clarify which works are relevant to us, we considered a simplified process of ML-based data analysis that comprises three general stages and is shown in Figure 1.

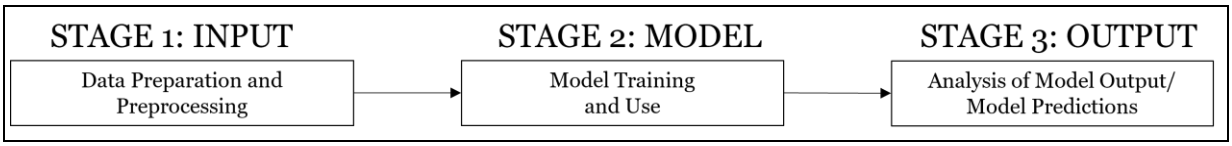


Figure 1. Three-stage Pipeline of ML-based Data Analysis

In Stage 1, the data is assessed and, in most cases, preprocessed and prepared. The DQ defects may already be identified and potentially addressed in this stage, e.g., by imputing missing data values. In Stage 2, the preprocessed data from Stage 1 is analyzed by an ML model in the context of an application, e.g., a diagnostic classification of healthcare data or a decision to grant or refuse a credit based on finance data. Typically, an ML model performs a regression, classification, clustering, or similar task in this second stage. In Stage 3, the ML model and its outputs (predictions) are analyzed and potential information about performance, validity and predictive uncertainty can be examined, e.g., by applying confidence measurements. Papers are relevant to our work if and only if they include an ML-based data analysis (Stage 2) in an application context as described above, i.e., if they comprise at least two consecutive stages of this three-stage pipeline. In particular, we exclude cases in which a method (e.g., an ML method) is only used to correct or address DQ defects (e.g., by ML-based imputation of missing values) without a further analysis of the preprocessed data in an application context. Furthermore, we consider the already

existing raw data that contains DQ defects, but do not explicitly focus on the real-world causes of these DQ defects prior to Stage 1 (e.g., malfunctioning sensors during data collection), as this constitutes a broad own field of research. Note that some existing methods for addressing DQ defects can also change the type of DQ defects in the data. For example, imputing missing values can turn a problem of missing data into a problem of inaccurate data as the imputed values are potentially incorrect. In such cases, we only analyze the DQ defects initially addressed by the method and consider any further changes to the data and its DQ defects to be an inherent part of the method.

In the next steps (2–4) of the ETDP, we defined the target user groups and intended purpose of our taxonomy and formulated its meta characteristic. The target groups of the taxonomy are researchers and practitioners working with ML on data with DQ defects. The purpose of the taxonomy is to provide a comprehensive structure for classifying and systematizing various methods on ML on data with DQ defects in both theoretical contexts and practical applications. We further aim to use our taxonomy to identify different research strands, foci, and research gaps in this area. We define the meta-characteristic of the taxonomy to be *dimensions of ML on data with DQ defects*. Step 5 includes determining the ending criteria for the iterative taxonomy building process and specifying the evaluation goals. For the ending criteria, we chose the criteria presented in [19,42]. They postulate that no new dimensions must be added or removed during the last iteration. After these conditions were met, we conducted a rigorous evaluation with external researchers and practitioners to assess whether the taxonomy is reliable, understandable, complete, and useful (cf. Section 5 and Supplement 2[1]). For this evaluation, we randomly selected 16 of 150 identified relevant papers as 'unseen' test papers and used the remaining 134 papers for the literature analysis in the taxonomy building process.

For our taxonomy, we conducted four iterations (Steps 6-10) as shown in Figure 2. More precisely, we alternately traversed two conceptual-to-empirical and two empirical-to-conceptual iterations, incorporating expertise from the authors and external researchers, as well as additional insights obtained from the identified works on ML on data with DQ defects. In Iteration 4, no further changes were proposed. Therefore, the ending conditions defined in Step 5 were met and the iterative taxonomy building process was completed. Each iteration and the development of the taxonomy throughout the iterations are described in detail in Supplement 1. The subsequent evaluation showed that the evaluation goals were fulfilled, and no further adjustments of the taxonomy dimensions and characteristics were necessary.

[1] The Supplement is available at https://zenodo.org/records/15631630.

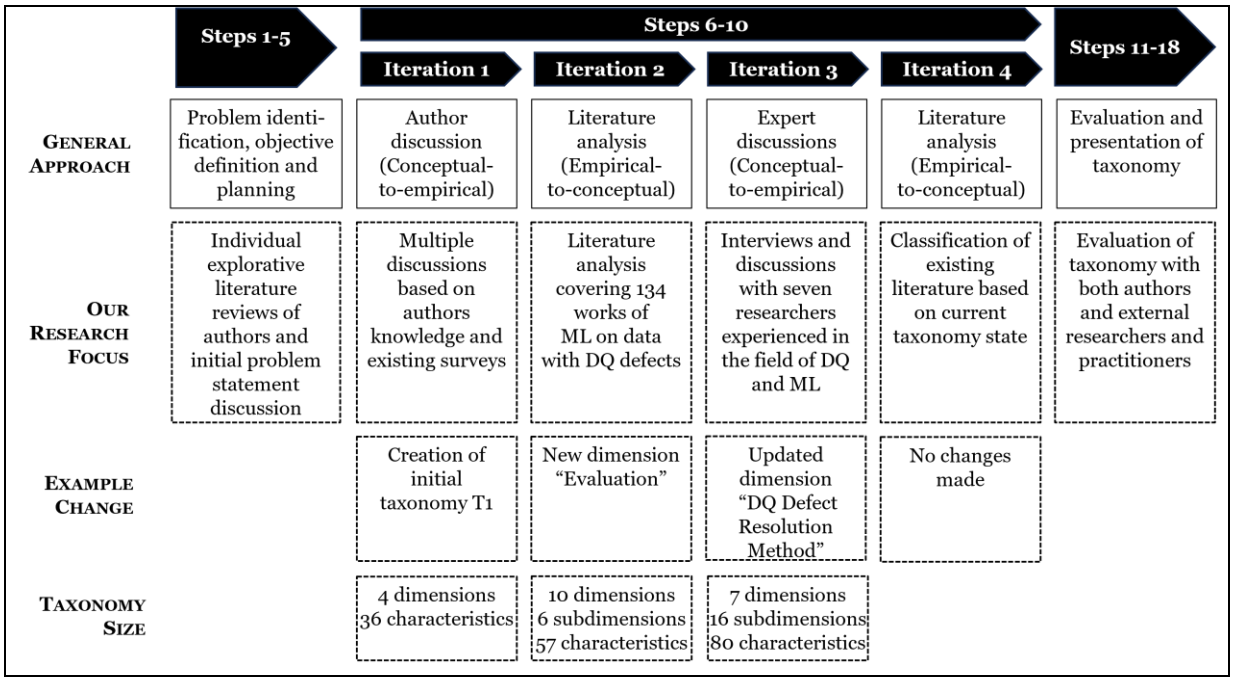


Figure 2. Steps of the Taxonomy Building Process

4 Taxonomy for ML on Data with DQ Defects

In the following, we present the dimensions and characteristics of our proposed taxonomy (cf. Figure 3). In general, the characteristics of each dimension and subdimension are non-exclusive, as papers may address different data with different DQ defects as well as different tasks or models. The only exceptions are characteristics of the form *no use* that are obviously exclusive to the other characteristics of that dimension.

The **first dimension** is the **DQ-related Problem Setting** which explicates how the paper describes the DQ problem it addresses. This dimension encompasses three subdimensions, the first of which is **DQ Dimension**. Its characteristics are the established data value-based DQ dimensions *accuracy*, *currency*, *completeness*, *consistency*, a characteristic *other* (dimension) containing rarely used DQ dimensions (e.g., uniqueness) as well as *not specified*. The last characteristic is only selected if the DQ dimension cannot be extracted or inferred from the paper, i.e., if the paper simply describes the data as defect or deficient (as opposed to using words such as "missing", indicating the characteristic *completeness*).

The second subdimension is the **DQ-related Task**, i.e., the objective with respect to handling DQ defects. This subdimension has eight characteristics. The first characteristic is *outlier correction*, i.e., the detection, removal and/or correction of outliers (outlier instances) in the data. The second characteristic is *noise correction*, i.e., the detection, removal and/or correction of inaccurate data instances, features or feature values. In literature, such inaccurate instances are often referred to as noisy instances, where for example both the feature values and the labels of the instances may be corrupted. The third characteristic is *inconsistency correction,* i.e., the detection, correction and/or removal of inconsistencies, e.g., by violations of consistency rules in the data. Such violations of consistency may comprise, e.g., domain value violations where a

feature value is outside the feasible domain of the feature, violations of rules describing relations between different instances or violations of syntactic or semantic rules in the data. The fourth characteristic is *imputation,* i.e., the imputation or removal of missing data which includes not only, e.g., the completion of missing data values but also missing data handling such as the list-wise deletion of incomplete instances or the deletion of incomplete features from the dataset. While the first four characteristics are associated to DQ-related tasks typically performed during the input stage, the fifth characteristic *defect data in training/use* refers to the handling of defect data during training or use of an ML model, i.e., by modifying an existing or designing a new ML model, DQ defects are directly handled without addressing them during the input stage of the pipeline. Examples for this characteristic include modifications of ML models to handle missing data or methods to make the training and use of ML models more robust against noisy data. The sixth characteristic is *data augmentation*, i.e., the creation of new data instances which are assumed to be free of DQ defects. Note that this differs from imputation as data augmentation typically creates new instances whereas imputation aims to fill in missing values in existing data instances [43]. The seventh characteristic is *duplicate detection,* i.e., the detection and removal of data duplicates. The final characteristic *other* DQ-related task comprises rare DQ-related tasks that do not fit into the other characteristics, such as knowledge transfer between domains or automated error detection.

The third subdimension is **Uncertainty Modeling/Quantification**, that is, if and how the paper formalizes and models the uncertainty resulting from DQ defects. More precisely, since the true data usually is not known (otherwise the DQ defects could be directly resolved), data affected by DQ defects is often referred to as uncertain data. By uncertainty modeling, we refer to quantitative, formal descriptions of data uncertainty based on a theoretical foundation. The characteristics of this dimension are, first, *probabilistic,* i.e., the modeling of uncertainty using probability theory, second, *belief-based,* i.e., the modeling of uncertainty using belief theory, third, *fuzzy theory,* i.e., the modeling of uncertainty using fuzzy (set) theory, and fourth, *other* theories/methods of uncertainty modeling. Moreover, if no such treatment of uncertainty is employed, the characteristic *no uncertainty modeling* is chosen. Since we require theoretical foundations for uncertainty modeling, the use of measures which may be interpreted as quantifications of uncertainty but lack theoretical foundations – such as noise scores or Softmax scores which are normalized on the interval [0,1] but must not be interpreted as probabilities [44] – is not considered as uncertainty modeling and thus belongs to this last characteristic.

The **second dimension** is the **Pipeline Position** in which the DQ defects are addressed. The characteristics of this dimension comprise the three pipeline stages of ML-based data analysis *input, model,* and/or *output* (cf. Figure 1). Note that while we require the paper to consider at least two consecutive stages to be relevant for our taxonomy, addressing the DQ defects does not necessarily take place in all of the stages considered. For example, a paper may include the first two steps (i.e., input and model), but only address DQ in the *input* stage, e.g., by imputing missing values without further consideration of the DQ defects in the subsequent use and evaluation of the model.

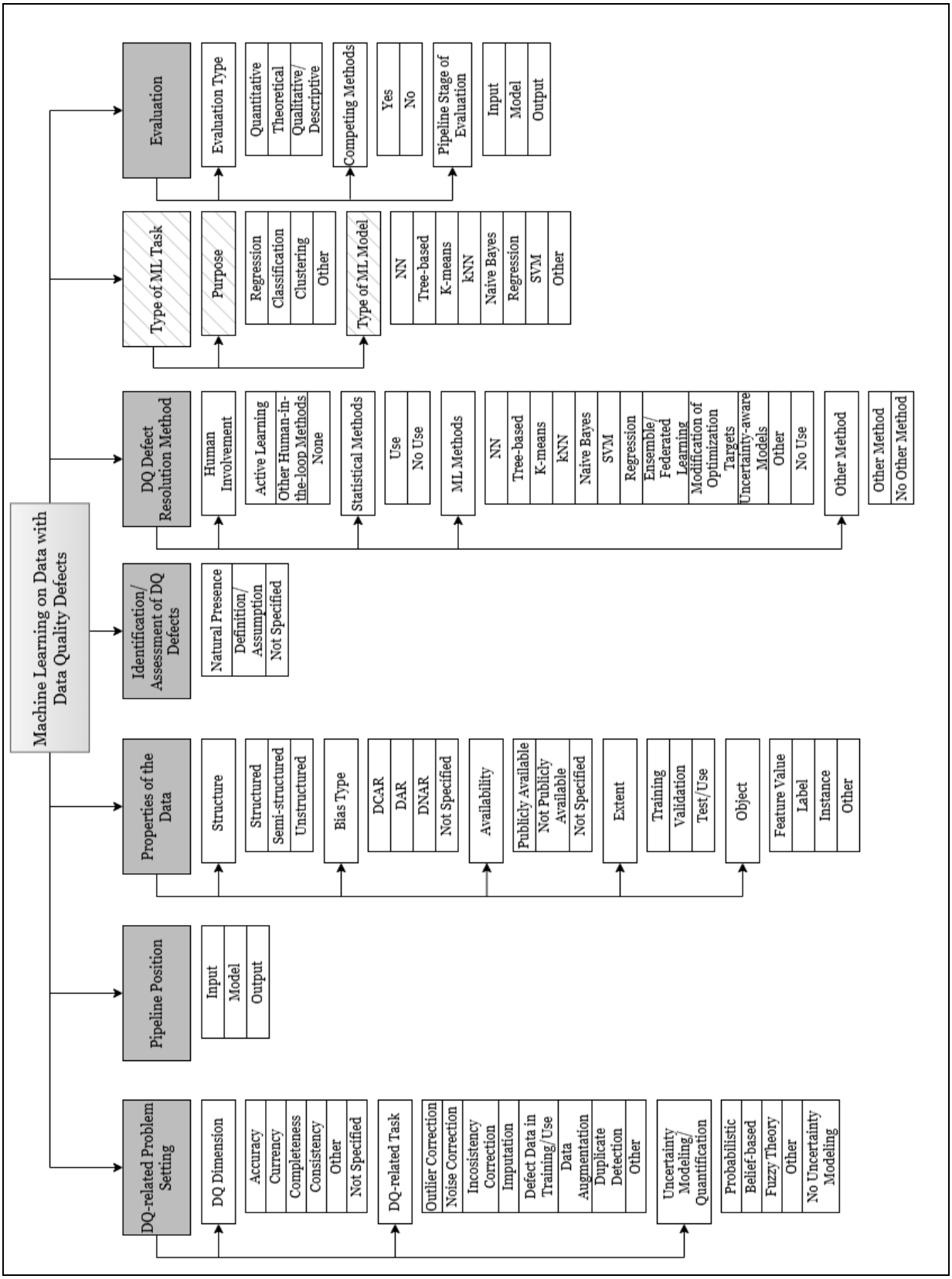


Figure 3. Taxonomy of ML on Data with DQ Defects

The **third dimension** focuses on the data and encompasses multiple subdimensions describing the **Properties of the Data** and the DQ defects considered in the paper. The first subdimension is the **Structure** of the data, i.e., whether the data are *structured, semi-structured,* and/or

unstructured. The second subdimension is the **Bias Type** of the data. In literature, the bias type or missingness mechanism of the data is a common notion when discussing the DQ dimension of completeness [45,46] where data can be missing not at random, missing at random or missing completely at random (cf. Supplement 1). However, we argue that these different bias types may also be defined and analyzed for other DQ dimensions such as accuracy (e.g., systematic versus random measurement errors). Therefore, we propose the notions of data being *defect completely at random (DCAR*), *defect at random (DAR)*, or *defect not at random (DNAR).* In addition, the bias type may also be *not specified* in case it cannot be extracted or inferred from the paper. Another subdimension is the **Availability** of the data, i.e., whether the data/dataset used in the paper is *publicly available* or *not publicly available*. The third characteristic for this subdimension is that the availability of the data is *not specified*. A fourth subdimension is the **Extent** of uncertainty/DQ defects in the data, that is, whether the DQ defects are contained in *training, validation*, and/or *test/use data*. A fifth subdimension is **Object** of uncertainty/DQ defects, that is, whether the DQ defects appear in *feature values, labels, instances*, and/or *other* data objects such as graphs, networks, or unstructured data (e.g., textual data). The characteristic *feature value* is selected if any feature values in the data are defect. The characteristic *instance* is selected in addition to *feature value* if and only if all feature values of instances are affected by DQ defects. Examples of such a case are papers where entire instances are randomly deleted—so that the values of all features are missing—to test the robustness of the ML model to this defect. Thus, *instance* can never be an exclusive characteristic of this subdimension but poses a useful characteristic by indicating whether the entire instance is affected instead of just some of its feature values. Note that the characteristics of all subdimensions of this dimension are generally non-exclusive, as any paper may not only address data with multiple defects, but also different types of data (e.g., structured and unstructured data).

The **fourth dimension** is **Identification/Assessment of DQ Defects**. This dimension has two characteristics based on why the data is known to be defect. On the one hand, DQ defects may be *naturally present* in the data. In this case, information about the presence of DQ defects must be obtained or made available externally. On the other hand, DQ defects can be obtained by *definition or assumption*, e.g., by assuming measurement data as uncertain/noisy (by definition), or by manually deleting feature values or instances to create incomplete data. The third characteristic is that the identification of DQ defects is *not specified* in the paper. Note that while the identification and assessment of DQ defects usually takes place in the input stage of the data analysis pipeline, DQ defects may also be identified in later stages, i.e., during the model use itself or even in the model output.

The **fifth dimension** is the **DQ Defect Resolution Method**, i.e., the method used in the paper to address or resolve the DQ defects. This dimension has a total of four subdimensions. The first subdimension is **Human Involvement** to address DQ defects. Its characteristics are *active learning methods*, *other human-in-the-loop methods,* and *none*. The second subdimension is the use of **Statistical Methods** to address DQ Defects. Its characteristics are *use* of statistical methods and *no use* of statistical methods. The third subdimension is the use of **ML Methods** to address

DQ Defects. The characteristics of this subdimension are various well-known standard ML models that can be used to address DQ defects as well as characteristics describing more specialized models for addressing DQ defects. The characteristics describing the use of standard ML models are use of *neural networks (NNs)*, *tree-based models* such as decision trees (DTs) or random forests (RFs), *support vector machines (SVMs)*, *Naïve Bayes (NB)*, *k-means*, *k-nearest neighbors (kNN)*, and *regression* models. Another characteristic is the use of *ensemble and/or federated learning* models, e.g., training multiple models on different imputed datasets and using ensemble learning to minimize the influence of inaccurately imputed data. The use of *uncertainty-aware models* is a characteristic which not only includes the use of existing, uncertainty-aware model architectures such as Bayesian NNs (BNNs) whose weights are probability distributions to account for uncertainty in the weights trained on potentially defect data, but also methods which modify or extend deterministic ML models to account for uncertainty. An example of the latter would be modifications of ML models to take probability distributions as input data and propagate the uncertainty information given by the distribution to the output. A further characteristic is the *modification of optimization targets* to address DQ defects. For ML models, different types of optimization targets exist and can be modified in various ways to account for DQ defects. Examples of such modifications are NNs with modified loss functions to account for DQ defects in the training data (e.g., by adding regularization) or DTs with modified entropy criteria in the tree building process. The final characteristics of this subdimension are *other* use of ML methods (e.g., Expectation Maximization or genetic programming) and *no use* of ML methods to address DQ defects. The fourth subdimension **Other Method** signals whether another method of DQ defect resolution, e.g., a fuzzy approach not using ML or statistical methods, is employed.

The **sixth dimension** is the **Type of ML Task** for the data analysis in the application context (and not for the handling of DQ defects as considered by the previous dimension). This dimension has two subdimensions, the first of which focuses on the main **Purpose** of the ML model and encompasses the characteristics *classification, clustering, regression*, or *other* type of task (e.g., segmentation). The second subdimension is the more detailed **Type of ML Model** used for the task, that is, whether an *NN* (or one of its variants, such as a CNN), a *tree-based* model (e.g., DT or RF), *SVM*, *k-means, kNN, NB,* a *regression model* (e.g., linear or logistic regression), or *other* ML model is employed. To obtain a reasonable number of characteristics for this subdimension, we propose to distinguish only the types of ML models listed above. Unlike other dimensions, this dimension strictly relates to the ML model but not to the DQ defects (yet it is very informative for users of the taxonomy). We therefore marked it in hatched style in Figure 3.

The **seventh** and final **dimension** is the **Evaluation** of the method that is conducted in the paper. This dimension comprises three subdimensions. The first subdimension is **Evaluation Type**, that is, whether the evaluation of the method is *quantitative* (i.e., by providing measures such as accuracy, precision, F1-measure, mean average error etc.), *theoretical* (i.e., by providing theoretical proofs and/or a deeper theoretical analysis for the method) and/or *qualitative/descriptive* (i.e., the evaluation discusses a wider range of purposes provided by the new method or its ability to handle special types of input data or DQ defects). The second subdimension is whether the

evaluation contains a comparison to **Competing Methods** from existing literature or not. The third subdimension is the **Pipeline Stage of Evaluation**, i.e., the stage (s) of the ML-based data analysis pipeline that the evaluation focuses on. The characteristics of this subdimension are the evaluation of the method in the *input* stage, the *model* stage, and/or the *output* stage. An example for an evaluation at the input stage is a performance assessment of the method addressing DQ defects only in the preprocessing, e.g., by measuring imputation errors, regardless of the resulting performance of the ML model. An example for an evaluation at the model stage would be the assessment of the performance of the ML model in terms of, e.g., accuracy, without versus with addressing the DQ defects. An example of an evaluation at the output stage would be when additional confidence levels for all model outputs are provided and their validity is analyzed.

5 Evaluation

Following the ETDP, we evaluate our taxonomy in terms of reliability, understandability, completeness, and usefulness. More precisely, since the taxonomy is designed for researchers and practitioners working with ML on data with DQ defects, it is a key requirement that the taxonomy is reliable and understandable to both target groups. Reliability constitutes the proportion of joint judgment in which there is an agreement [47]. Understandability describes whether the taxonomy can be understood and employed by its target groups and can be measured by task performance [48]. Furthermore, the taxonomy should be complete und useful [42,49]. To assess these four criteria for our taxonomy, we performed a two-part evaluation with external experts, which is described in the following. A detailed evaluation protocol is provided in Supplement 2.

5.1 Evaluation Methodology

For the evaluation, we used 16 papers which had been randomly selected from the 150 papers identified in Iteration 2 of the taxonomy building process (cf. Section 3). We were able to recruit at panel of eight external experts (called external raters), each with expertise in the fields of DQ and ML and between 3 and 13 years of experience in both academic research and industry. Each external rater received six randomly assigned papers and classified them according to the taxonomy. Thus, in total, each of the 16 papers was classified by three external raters (for details cf. Supplement 2). We first evaluated the reliability and understandability of the taxonomy based on these results. Subsequently, all eight external raters were asked to fill out a questionnaire on the completeness and usefulness of the taxonomy.

To evaluate the reliability of the taxonomy, we measured the agreements between the external raters and a gold standard (in line with [50]) for the classification of the 16 papers for each dimension and over all dimensions on average. To create the gold standard, each test paper was also classified by the four authors of the taxonomy (internal raters). The authors then discussed their classifications to reach a consensus on the classification of all 16 papers (cf. Supplement 2). The agreements between each internal rater and the gold standard were measured as well. In addition, we measured the agreements of the groups of researchers and practitioners respectively (both as parts of external raters). This allowed us to assess whether the description of the

taxonomy together with its dimensions and characteristics is understandable for experts with different backgrounds, as substantial agreement is only possible if the taxonomy is well understood.

A primary concern for any evaluation is that it should be designed in a such a way that high-quality answers can be expected. Well- established research on the appropriate length and complexity of evaluations suggests that survey quality suffers when respondents have difficulty completing complex tasks [51]. Similarly, in evaluations that require high effort, loss of interest among participants may lead to ill- considered and unreliable answers or high non-response to questions, potentially invalidating the entire results of the study [52]. Such concerns also apply for our evaluation as each external rater would have to read, understand, and classify six full research papers describing complex ML methods and applications and, in most cases, containing a number of mathematical formulas. Thus, we concluded that the volume and complexity of the task should be reduced to allow for complete and reliable answers by each rater, which was supported by discussions between the authors and other practitioners prior to the evaluation. Therefore, the raters did not receive the full papers, but automatically generated summaries with a length of no more than two pages per paper. These summaries could not be created by the authors, as the authors' knowledge about the taxonomy could certainly bias the paper summaries in a way that could lead the external raters to classify the papers in the same way as the authors. To avoid this, we used GPT-4o to create a summary of each paper based on a pre-defined prompt which contains a brief description of the taxonomy's objectives, but not its explicit dimensions and characteristics. For a *post-hoc* evaluation of the quality of these paper summaries, we conducted a taxonomy classification using four randomly selected test papers. This classification was independently performed by the authors once based on the full papers and once based on the respective summaries generated by GPT-4o. A comparison of both taxonomy classifications for each paper revealed a near-perfect agreement (cf. Supplement 2.9 for details). Furthermore, these summaries serve as a consistent and comprehensible basis for all internal and external raters, regardless of time constraints or individual knowledge. Thus, both internal and external raters only read the summaries, but not the full papers for their classifications. In particular, the gold standard is also based on the summaries to ensure the comparability between the gold standard and the test persons' classifications (cf. Supplement 2).

To measure and analyze the agreement between groups of raters, we used the concept of inter-rater reliabilities. This approach has proven to be appropriate for taxonomy evaluations as it measures the consensus between multiple classification results of the same papers [49,53]. There are several metrics for measuring inter-rater reliability. In our evaluation, we use two well-established metrics, the "joint probability of agreement" and a Kappa statistic [54,55], namely the Prevalence and Bias adjusted Kappa (PABAK) [55]. Both metrics are discussed in detail in Supplement 2. Completeness and usefulness were evaluated by the average questionnaire results on a 5-point Likert scale.

5.2 Results

We first discuss the results of the evaluation of reliability and understandability based on the classifications by the test persons. To measure inter-rater reliability among the (internal) raters, we calculated inter-rater reliability between rater and gold standard for each internal rater and aggregated the results across the four raters. Across all characteristics, we received an average joint probability of agreement of 0.909 and an average PABAK of 0.840, showing almost perfect agreement between (internal) raters according to established guidelines [56]. Similarly, we received a joint probability of agreement between external raters and the gold standard of 0.821 and a PABAK value of 0.691, indicating substantial agreement [56]. Moreover, we also compared the average results between the group of four researchers and the group of four practitioners among the eight external raters to assess whether the taxonomy could be equally understood and applied by both user groups. The group of researchers achieved an average PABAK of 0.708 with respect to the gold standard whereas the group of practitioners achieved 0.674, showing comparable performance of both groups (cf. Table 2).

	Internal raters	**External raters**		
	Authors	All external raters	Practitioners	Researchers
Joint probability of agreement	0.909	0.821	0.816	0.825
PABAK	0.840	0.691	0.674	0.708

Table 2. Aggregated Inter-rater Reliability between Gold Standard and Raters

The results show that the criterion of reliability is supported and that the descriptions of the taxonomy are understandable not only to the authors, but also for external persons not involved in the taxonomy building process. Naturally, the authors achieved higher agreement with the gold standard (which emerged as the result of the authors' classifications and discussion) in every dimension. For a more detailed analysis of the results, we now discuss the results for each (sub)dimension separately. For this analysis, we refer to the joint probability of agreement between raters and gold standard as these values are very intuitive and easy to interpret. The results for each (sub)dimension are shown in Table 3.

Dimension	**Average joint probability of agreement with external raters**	**Average joint probability of agreement with authors**
DQ Dimension	0.920	0.995
DQ-related Task	0.880	0.938
Uncertainty Modeling/ Quantification	0.867	0.938
Pipeline Position	0.736	0.906
Structure	0.938	0.948
Bias Type	0.854	0.934
Availability	0.847	0.896
Extent	0.667	0.828
Object	0.880	0.910
Identification/Assessment of DQ Defects	0.653	0.724

Human Involvement	0.931	0.969
Statistical Methods	0.708	0.773
ML Methods	0.861	0.908
Other Method	0.604	0.836
Purpose	0.901	0.969
Type of ML Model	0.927	0.953
Evaluation Type	0.965	0.969
Competing Methods	0.917	0.969
Pipeline Stage of Evaluation	0.541	0.901

Table 3. Detailed Inter-rater Reliability between Raters and Gold Standard

Agreement was high for all subdimensions of the dimension *DQ- related Problem Setting*, especially for the subdimension *DQ Dimension* with a strong agreement of 0.920 from the external raters and near- perfect agreement of 0.995 from the authors. For the dimension *Pipeline Position*, the agreement between external raters and the gold standard was relatively low at 0.736, which was due to some differences in the assessment of whether DQ defects were addressed in the output stage. More precisely, only mentioning the addressed DQ defects as the cause for improved model performances was sufficient for some raters to tag the output stage, while for others it was not. These differences did not exist among the authors, which explains their higher agreement. The subdimensions of the dimension *Properties of the Data* generally showed strong agreement across raters, particularly the first subdimension *Structure* and the last subdimension *Object*. For the subdimension *Bias Type*, the agreement was also generally high. Disagreements occurred in a few cases where the bias type of the used dataset was only implicit or the paper discussed all three possible bias types in general, although the data in the evaluation did not exhibit all three types. For the third subdimension *Availability* the average agreement was 0.847, with disagreements arising mostly because some raters knew whether specific datasets are publicly available and classified the paper accordingly, while others classified the availability as *not specified* because they had no further knowledge about the availability of the mentioned datasets. For *Extent*, the lower agreement of 0.667 was due to the fact that the presence of DQ defects in the test data was only implicit in some papers. In the gold standard, the characteristic *Test/Use* was selected in all of these cases, whereas some external raters only selected it when defects of the test data were explicitly mentioned. A dimension with lower agreement was *Identification/Assessment of DQ Defects*, with results of 0.653 among external raters and 0.724 among authors. In many papers, this information was only implicit or mentioned very briefly which may have led to oversights and subsequent classifications as *not specified*. Furthermore, in some cases it was not entirely clear to the raters whether the DQ defects described in a paper were existing defects identified by measurements, or whether the defects were simply assumed. The subdimensions of *DQ Defect Resolution Method* showed varying levels of agreement. The agreement on the subdimension *Human Involvement* was very high while the subdimension *Statistical Methods* showed a lower value of 0.708 as methods such as linear regression or k-means were considered statistical by some raters and ML methods by others. The agreement on *ML Methods* was generally high, especially considering that this subdimension encompassed several non-exclusive characteristics, with many proposed methods belonging to multiple characteristics.

Disagreements occurred in cases where some raters found the distinction between ML methods used for DQ defect improvement and ML methods used for the ML-based data analysis in the application context to be not entirely clear. For *Other Method*, the external raters achieved a lower agreement of 0.604 compared to the authors' agreement of 0.836, because individual external raters sometimes classified uncommon approaches (e.g., graph learning techniques) they were unfamiliar with as *Other Method* whereas other raters with more knowledge of these particular approaches recognized them as statistical or ML methods. Both subdimensions of the dimension *Type of ML Task* showed high agreement between the external raters. Agreement on *Evaluation Type* and *Competing Methods* was also high, with almost all raters agreeing that the majority of papers featured a quantitative evaluation. The lowest agreement between external raters and the gold standard was recorded in the final subdimension *Pipeline Stage of Evaluation*. In the taxonomy, an evaluation of the ML model performance in Stage 2 was defined as an evaluation in the model stage. An evaluation in the output stage was defined as an evaluation that explicitly used DQ or uncertainty/confidence measures for the model output, or an evaluation of an output-based method of addressing the DQ defects. However, this definition partly differs from the also intuitive understanding that any evaluation of model performance is an evaluation in the output stage. Therefore, four external raters classified the evaluation of ML model performance as an output stage evaluation, contrary to the definition used by the authors, leading to a lower agreement. This difference of interpretation did not occur among the authors, resulting in a much higher agreement of 0.901 with the gold standard. In summary, the reliability and understandability of the taxonomy was supported for both internal and external raters across all dimensions. Lower agreement between raters on certain dimensions could be attributed to the raters' differing background knowledge about particular aspects and methods discussed in some of the more specialized papers.

We now discuss the results of the questionnaire-based evaluation of the taxonomy regarding completeness und usefulness. Each external rater received a questionnaire with four questions, where answers on a 5-point Likert scale (1 =strongly agree, 5 =strongly disagree) were possible (cf. Supplement 2). In addition, the external raters were asked to make suggestions for taxonomy updates, especially if they did not agree that the taxonomy fulfilled one of the criteria. The results are shown in Table 4.

Almost all of the external raters agreed that the taxonomy is complete (cf. Questions 1 and 2), with the exception of one of the practitioners. However, this practitioner suggested that the dimensions *Reason for DQ Defect* and *Goal of Real-World ML Application* were missing from the taxonomy. The first suggested dimension is out of scope by the definition of our taxonomy (cf. Section 2.1 and Section 3) as it relates to the data collection process before the input stage. The latter dimension is also out of scope, as it relates to the application of the results beyond the output stage (cf. Section 2.1 and Section 3). Importantly, all external raters consistently agreed that the taxonomy is useful (cf. Question 3). The external raters also agreed that all dimensions and characteristics were necessary (cf. Question 4), with one practitioner being neutral on this question, reasoning that he was only able to assess this for the six reviewed papers but not in

general. Since the practitioner did not suggest removing any dimensions or characteristics and agreed that other papers may indeed discuss and require all dimensions, we decided not to update the taxonomy based on this response. Finally, we note that both researchers and practitioners agree that the taxonomy satisfies the evaluated criteria of completeness and usefulness.

Question	All test persons	Researchers	Practitioners
Q1: Does the taxonomy include **all relevant dimensions and characteristics** to classify the six papers and is the taxonomy in your opinion complete in general?	2.17	2.00	2.25
Q2: Are the **dimensions and characteristics detailed enough** to allow differentiation between similar objects in the six papers and in general?	1.67	1.50	1.75
Q3: Is the **taxonomy useful** for the structuring and classification of the ML methods in the six papers and for ML methods on data with DQ defects in general?	1.17	1.50	1.00
Q4: Are **all dimensions and characteristics necessary** to classify the six papers and to classify ML methods on data with DQ defects in general?	1.50	1.00	1.75

Table 4. Average Questionnaire Results on Likert Scale (Lower is Better)

6 Discussion

In this section, we discuss the results and contributions of our work and provide guidance for future research in the field of ML on data with DQ defects. We proceed as follows: We first discuss our evaluation results. Then, we outline the contributions of our taxonomy to both research and practice by discussing potential applications of the taxonomy in both areas. We also reflect how future research in the field of ML on data with DQ defects could potentially influence our taxonomy. Finally, we identify research gaps and potential for future research based on our taxonomy and its building process.

The evaluation results support the reliability, understandability, completeness, and usefulness of our taxonomy. Both researchers and practitioners were able to apply the taxonomy to various papers and achieve substantial inter-rater agreement. While most taxonomy dimensions showed very strong agreement between raters, a few characteristics of certain dimensions were more difficult to distinguish, especially for external raters, resulting in slightly lower inter-rater agreement for these dimensions. The dimension *Identification/Assessment of DQ Defects* proved to be particularly challenging, as raters were asked to decide whether DQ defects in a paper were artificially introduced or assumed, or whether defects were naturally present. Often, the identification of such (natural) DQ defects poses a challenge in itself, for which DQ metrics [57] (initially mentioned in the description of *Natural Presence*) potentially offer a solution. However, using DQ metrics to indicate potential DQ defects is often part of the method addressing the defects itself. Thus, mentioning DQ metrics in this description could lead raters to incorrectly classify a paper without natural presence of DQ defects into this characteristic simply because DQ metrics are used for addressing the defects. Therefore, we updated the taxonomy by slightly

revising the initial description of this characteristic for our final description of the taxonomy in this work. While the subdimensions *Other Method* and *Pipeline Stage of Evaluation* also proved to be challenging for some external raters, we found that the lower agreement among external raters was due to different knowledge about the methods used in the papers and the subdimensions are thus generally comprehensible. All other dimensions and characteristics, especially the central dimensions *DQ-related Problem Setting*, *DQ Defect Resolution Method* (particularly *ML Methods*), and *Type of ML Task* satisfied the criterion of very high reliability. Overall, the taxonomy has shown to be reliable and understandable to users from both research and practice. In addition, both researchers and practitioners agreed that the taxonomy is complete and useful. We therefore propose to not further adapt its dimensions, characteristics, and their descriptions.

Our taxonomy provides a clear and intuitive framework for ML on data with DQ defects, which is a highly relevant area in both research and practice. Researchers developing new methods in this field can use our taxonomy to position their work within existing literature and can exactly define the research gaps they aim to address. Moreover, the dimensions and characteristics of our taxonomy allow researchers to identify additional relevant information that should be provided in their work, for example, clear and comprehensive descriptions of the analyzed data and its DQ defects. Similarly, the dimensions of our taxonomy support researchers in describing the DQ-related problem that they aim to address with their method (e.g., the extent of DQ defects from training to test data which must be considered in the design of any method), thereby providing guidance to improve their work. Moreover, possible adaptations of methods to related problems can be identified in this manner, thus broadening their contribution to research. For instance, a method for addressing DQ defects by modifying the optimization criterion in NNs may also be suited for other types of ML models in which similar optimization criteria are used. For practitioners, the taxonomy allows them to structure, search, and filter existing methods dealing with ML on data with DQ defects according to their individual needs and preferences. In particular, an overview of available methods for specific use cases can be provided and the most suitable option can be selected, considering the exact DQ defect(s), the data properties, and requirements for the ML-based analysis. The taxonomy dimensions can also help practitioners to identify all relevant dimensions that need to be considered when working with ML on data with DQ defects.

In the future, research in the field of ML on data with DQ defects may result in new ML models and new methods for addressing DQ defects. Our taxonomy will remain implicitly valid without adjustments as any new models and methods can be classified into the characteristics “other”. However, it may be more informative for users of the taxonomy to add new characteristics for models and techniques becoming more relevant in the future, thus making them explicit. Such updates will be possible iteratively without the need for major changes or a new taxonomy building process. Therefore, we argue that our taxonomy can not only support researchers and practitioners in their current work but can also be iteratively adapted, ensuring continued support as the research field advances.

Furthermore, while we do not aim to present an exhaustive, fully comprehensive overview or survey that describes all existing literature in the field of ML on data with DQ defects in detail, the taxonomy and the literature that was researched and classified during the taxonomy building process (cf. Supplement 1) has shed light on current research foci and research gaps. In the following, we will discuss six key findings which we identified as underrepresented and offer potential for future research directions.

First, the taxonomy building process demonstrated that many papers in the field of ML on data with DQ defects lack a conceptional analysis of the DQ defects and provide little detail on them. The DQ defects are often discussed very briefly, and relevant dimensions such as the *Extent* and the *Bias Type* are not mentioned at all and ignored, despite being essential for the functionality of a method. Of the 150 analyzed papers (cf. Iteration 2 and Evaluation), only 36 explicitly include the bias type of the analyzed data. Furthermore, the identification of DQ defects is often not discussed in detail, even though it is highly relevant to practical applicability whether the defects occur naturally or whether they are artificially introduced. Of the 150 analyzed papers, 48 do not specify how the DQ defects were identified. For datasets with natural presence of DQ defects, it is also relevant how they are identified and what criteria (e.g., the value of a DQ metric exceeding a certain threshold) are used to decide whether a DQ defect is present. When DQ defects are introduced artificially, the specific mechanism is critical as it can introduce different biases and patterns, thus making some methods unsuitable. For example, if data uncertainty is introduced by imputation of randomly deleted values [58], biases of the imputation method (e.g., using predictions of an ML model for imputation that is biased towards the majority class in its training data) can change the distribution of feature values (compared to the original, deleted values), thereby adding significant bias to the data. The taxonomy building process also showed that the DQ defects and the resulting data uncertainty are rarely measured or quantified for a more detailed analysis, e.g., by explicitly modeling uncertain values probabilistically and analyzing the properties of the probability distribution.

Second, DQ defects and uncertainty in the training or test data of an ML model may lead to significant uncertainty in its predictions. Not only initial data uncertainty, but also the resulting predictive uncertainty is often not analyzed in detail. Quantifying and analyzing predictive uncertainty are crucial for the responsible use of ML models and valid risk assessment in ML-assisted decision-making. Otherwise, a user may become overconfident in an ML model's decisions and blindly trust them, ignoring any potential risks which could be accurately assessed by quantifying predictive uncertainty. However, most papers studied during the taxonomy building process address DQ defects either at the input or the model stage, but rarely in the output stage where predictive uncertainty is typically analyzed. Of the 150 analyzed papers, only five address DQ defects and the resulting uncertainty in the output stage. Only some papers on BNNs do discuss predictive uncertainty, but they usually focus on model uncertainty—i.e., the uncertainty of the trained model weights—as its primary cause, while mostly ignoring data uncertainty (with the exception of a few selected papers, e.g., [59]).

The taxonomy building process also revealed that research in ML on data with DQ defects often focuses on certain types of data and DQ defects, while others are underrepresented. For DQ defects, the majority of papers addresses the DQ dimensions completeness and accuracy, whereas other important dimensions such as consistency and currency are rarely discussed, even though their relevance is highlighted in a few studies [38]. Addressing DQ defects from these dimensions in the training and use of ML methods should thus be studied carefully in future research. It may also be promising to analyze if methods developed for incomplete or inaccurate data can be adapted for data suffering from defects related to other DQ dimensions. Regarding the type of the analyzed data, the most commonly addressed type is structured data (121 of the 150 papers). With respect to unstructured data, image data is focused on most often, whereas other types are rarely studied in the context of ML analysis with DQ defects. However, it can be expected that the widespread use and rapid progress of large language models will lead to a significantly greater focus on investigating DQ defects in textual data in the future.

Another finding is that existing research focuses on specific types of methods to address DQ defects. While addressing DQ defects using statistical methods or ML methods is very common, methods based on human involvement such as active learning are discussed in only seven of the 150 identified papers. Active learning methods have been recognized in literature as useful approaches to improve DQ by including human decisions on strategically selected data instances (to avoid exhaustive workloads for human users), thereby improving model performance compared to fully automated models [38]. Another type of method which is discussed in 16 of the 150 papers is the modification of ML models for data with explicitly modeled data uncertainty, e.g., by probability distributions. While most research focuses on addressing DQ defects or data uncertainty at the input stage, it has also proved promising to adapt ML models to handle probability distributions representing uncertain data instances (e.g., [15]). Such methods require changes to the ML model design but allow explicit quantification of the predictive uncertainty in the model output by probability distributions. Therefore, they enable mathematically sound confidence estimations, making them a promising focus of future research.

The taxonomy building process also revealed several issues regarding the evaluation of the proposed methods. Firstly, evaluations often do not address the (sub)dimension *Extent* of the data with DQ defects and its potential impact on the evaluation. In particular, when methods are trained on data with DQ defects, we observed many papers in which it is not discussed explicitly whether the test data are actually affected by the same or similar DQ defects. In some cases, test data is even assumed to be free of DQ defects even though the training data suffer from them. The existence of test data without DQ defects in such cases is either an unrealistic setting or the result of non-random test data selection (e.g., only data without missing values is selected as test data), which often introduces biases into the test data. Therefore, the extent of the uncertain data and its potential impact on the evaluation should be determined and analyzed. Secondly, evaluations are often restricted to an evaluation of the ML model performance. However, if DQ defects are addressed in the input stage, the respective results should also be directly evaluated before training and using the ML model, as the performance of the DQ defect improvement method is

only evaluated implicitly by measuring the ML model performance. While 98 of the 150 identified papers address DQ defects in the input stage, only 14 of these 98 papers also perform an evaluation in the input stage. For such an evaluation in the input stage, suitable metrics for measuring whether the DQ defects were successfully addressed are required. For some types of DQ defects such metrics already exist, for example, imputation performance on artificially introduced missing data can be measured by comparing the imputed values with the known original values. However, depending on the aspects of the DQ defects being addressed in the input stage, suitable metrics still need to be developed by future research.

Finally, to perform thorough, representative, and reproducible evaluations of proposed methods for ML on data with DQ defects, it is essential to have benchmark datasets containing specific and known DQ defects. For example, such datasets could contain erroneous feature values together with the information of which feature values are erroneous, or they could represent known data uncertainty by probability distributions. With such benchmark datasets available, authors do not have to resort to assuming or artificially introducing DQ defects into datasets without known defects. Furthermore, benchmark datasets allow different researchers to evaluate their proposed methods on a common basis, thus enabling standardized evaluations and improving comparability between the performances of different methods. In addition, the existence of benchmark datasets could also encourage the study of underrepresented data types or DQ defect types. As a positive example for the task of duplicate detection, Naumann [60] created such benchmark datasets that include ground truth information about the actual duplicates in the data, providing a valuable contribution to duplicate detection research. However, for most types of data and DQ defects, such benchmark datasets including information about DQ defects/data uncertainty are hardly available. Although platforms such as the UCI ML Repository[2] or Kaggle[3] do provide benchmark datasets with DQ defects (e.g., missing or inaccurate values), those datasets neither include a ground truth for DQ defects nor quantify resulting data uncertainty, e.g., through probability distributions for inaccurate values. This leads researchers to use their own methods to impute missing values or to generate probability distributions representing data uncertainty rather than relying on standardized benchmarks. Creating and providing benchmark datasets for ML on data with DQ defects should therefore be a primary goal for future research in this field.

7 Conclusion

DQ defects pose a critical challenge that must be addressed in ML- based data analysis as they can severely affect the validity and performance of ML models and the (human) decisions based on them, as recent examples highlight (cf. Introduction). To structure and systematize this field for both researchers and practitioners, we created a taxonomy for the field of ML on data with DQ defects covering relevant dimensions of both DQ defects and methods that aim to address these defects in both training and use of ML models. Based on this taxonomy, we identified

[2] https://archive.ics.uci.edu/datasets

[3] https://www.kaggle.com/datasets

existing gaps and challenges in this research field. Despite the fact that we developed and evaluated the taxonomy according to the ETDP [19], the resulting taxonomy is not without limitations. First of all, not all characteristics of each dimension can be arbitrarily combined with all characteristics of the other dimensions, i.e., the dimensions are not perfectly orthogonal. For example, the use of uncertainty-aware ML models such as BNNs to address DQ defects already implies that probabilistic uncertainty modeling was employed. Although this represents a typical limitation of taxonomies, we have taken careful measures to ensure that most characteristics across different dimensions are independent and can be combined in papers. A further limitation to acknowledge is that we only considered the process of ML-based data analysis from the input data to the model output stage, and hence deliberately excluded the human user who receives the model output and derives decisions based on it. Similarly, we do not address details of the data collection process that may initially cause the DQ defects in the input data. Examinations and systematizations of these stages may be subjects of future research.

8 References

[1] C. Leibig, M. Brehmer, S. Bunk, D. Byng, K. Pinker, L. Umutlu, Combining the strengths of radiologists and AI for breast cancer screening: a retrospective analysis, Lancet Digit. Health 4 (2022) e507-e519.

[2] M. El Hajj, J. Hammoud, Unveiling the influence of artificial intelligence and machine learning on financial markets: a comprehensive analysis of AI applications in trading, risk management, and financial operations, J. Risk Financ. Manag. 16 (2023) 434.

[3] M. Möhlmann, L. Zalmanson, O. Henfridsson, R.W. Gregory, Algorithmic management of work on online labor platforms: When matching meets control, MISQ 45 (2021) 1999–2022.

[4] Y. Gong, G. Liu, Y. Xue, R. Li, L. Meng, A survey on dataset quality in machine learning, Inf. Softw. Technol. 162 (2023) 107268.

[5] N. Polyzotis, S. Roy, S.E. Whang, M. Zinkevich, Data lifecycle challenges in production machine learning, SIGMOD Rec. 47 (2018) 17–28.

[6] P. Krasikov, C. Legner, Introducing a data perspective to sustainability: how companies develop data sourcing practices for sustainability initiatives, Commun. Assoc. Inf. Syst. 53 (2023) 162–188.

[7] S. Mohammed, L. Budach, M. Feuerpfeil, N. Ihde, A. Nathansen, N. Noack, H. Patzlaff, F. Naumann, H. Harmouch, The effects of data quality on machine learning performance on tabular data, Inf. Syst. (2025) 102549.

[8] Sama, Top 3 examples of how poor data quality impacts ML, 2024. https://www.sama.com/blog/top-3-examples-of-how-poor-data-quality-impacts-ml (accessed 10-30-2024).

[9] P.D. Wentzell, S. Hou, Exploratory data analysis with noisy measurements, J. Chemom. 26 (2012) 264–281.

[10] Y. Li, T. Bao, H. Chen, K. Zhang, X. Shu, Z. Chen, Y. Hu, A large-scale sensor missing data imputation framework for dams using deep learning and transfer learning strategy, Meas. 178 (2021) 109377.

[11] J. Peng, J. Hahn, K.-W. Huang, Handling missing values in information systems research: A review of methods and assumptions, Inf. Syst. Res. 34 (2023) 5–26.

[12] B. Heinrich, M. Klier, Assessing data currency — A probabilistic approach, J. Inf. Sci. 37 (2011) 86–100.

[13] Y. Zak, A. Even, Development and evaluation of a continuous-time Markov chain model for detecting and handling data currency declines, Decis. Support Syst. 103 (2017) 82–93.

[14] H. Zhang, T. Nakamura, T. Isohara, K. Sakurai, A review on machine unlearning, SN Comput. Sci. 4 (2023).

[15] T. Krapf, M. Hagn, P. Miethaner, A. Schiller, L. Luttner, B. Heinrich, Piecewise linear transformation – Propagating aleatoric uncertainty in neural networks, in: Proc. of the AAAI Conf. on Artif. Intell. 38, 2024, pp. 20456–20464.

[16] M. Pattanodom, N. Iam-On, T. Boongoen, Clustering data with the presence of missing values by ensemble approach, in: Second Asian Conf. on Defence Technol, 2016, pp. 151–156.

[17] S. Das, S. Datta, B.B. Chaudhuri, Handling data irregularities in classification: Foundations, trends, and future challenges, Pattern Recogn. 81 (2018) 674–693.

[18] N. Li, Y. Qi, C. Li, Z. Zhao, Active learning for data quality control: A survey, J. Data Inf. Qual. 16 (2024) 1–45.

[19] D. Kundisch, J. Muntermann, A.M. Oberländer, D. Rau, M. Röglinger, T. Schoormann, D. Szopinski, An update for taxonomy designers, Bus. Inf. Syst. Eng. 64 (2022) 421–439.

[20] R. Price, G. Shanks, A semiotic information quality framework: Development and comparative analysis, J. Inf. Technol. 20 (2005) 88–102.

[21] Y. Wand, R.Y. Wang, Anchoring data quality dimensions in ontological foundations, Commun. ACM 39 (1996) 86–95.

[22] C. Batini, C. Cappiello, C. Francalanci, A. Maurino, Methodologies for data quality assessment and improvement, ACM Comput. Surv. 41 (2009) 1–52.

[23] C. Cichy, S. Rass, An overview of data quality frameworks, IEEE Access 7 (2019) 24634–24648.

[24] B. Heinrich, M. Klier, A. Schiller, G. Wagner, Assessing data quality – A probability-based metric for semantic consistency, Decis. Support Syst. 110 (2018) 95–106.

[25] S. Krishnan, J. Wang, E. Wu, M.J. Franklin, K. Goldberg, ActiveClean, Proc. VLDB Endow. 9 (2016) 948–959.

[26] Y.M. Herrera, D. Kapur, Improving data quality: Actors, incentives, and capabilities, Polit. Anal. 15 (2007) 365–386.

[27] D. Peng, F. Wu, G. Chen, Data quality guided incentive mechanism design for crowdsensing, IEEE Trans. on Mobile Comput. 17 (2018) 307–319.

[28] C.G. Northcutt, L. Jiang, I.L. Chuang, Confident learning: Estimating uncertainty in dataset labels, J. Artif. Intell. Res. (2019).

[29] I. Valentim, N. Lourenco, N. Antunes, The impact of data preparation on the fairness of software systems, in: IEEE 30th Int. Symp. on Softw. Reliab. Eng,2019, pp. 391–401.

[30] A. Barry, L. Han, G. Demartini, On the impact of data quality on image classification fairness, in: Proc. of the 46th Int. ACM SIGIR Conf. on Res. and Dev. in Inf. Retr, 2023, pp. 2225–2229.

[31] S. Guha, F.A. Khan, J. Stoyanovich, S. Schelter, Automated data cleaning can hurt fairness in machine learning-based decision making, IEEE Trans. Knowl. Data Eng. 36 (2024) 7368–7379.

[32] B. Heinrich, T. Krapf, P. Miethaner, EXPLORE: A novel method for local explanations, in: Proc. of the Int. Conf. on Inf. Syst. (ICIS), 2024.

[33] Z. Zhang, G. Chen, Y. Xu, L. Huang, C. Zhang, S. Xiao, FedDQA: A novel regularization-based deep learning method for data quality assessment in federated learning, Decis. Support Syst. 180 (2024) 114183.

[34] A. L'Heureux, K. Grolinger, H.F. Elyamany, M.A.M. Capretz, Machine learning with big data: Challenges and approaches, IEEE Access 5 (2017) 7776–7797.

[35] A. Paleyes, R.-G. Urma, N.D. Lawrence, Challenges in deploying machine learning: A survey of case studies, ACM Comput. Surv. 55 (2023) 1–29.

[36] G. Rekha, A.K. Tyagi, V.K. Reddy, A wide scale classification of class imbalance problem and its solutions: A systematic literature review, J. Comput. Sci. 15 (2019) 886–929.

[37] G. Haixiang, L. Yijing, J. Shang, G. Mingyun, H. Yuanyue, G. Bing, Learning from class-imbalanced data: Review of methods and applications, Expert Syst. Appl. 73 (2017) 220–239.

[38] M. Priestley, F. O'donnell, E. Simperl, A survey of data quality requirements that matter in ML development pipelines, J. of Data and Inf. Qual. 15 (2023) 1–39.

[39] V.N. Gudivada, A. Apon, J. Ding, Data quality considerations for big data and machine learning: Going beyond data cleaning and transformations, Int. J. Adv. Softw. (2017) 1–20.

[40] J. Gawlikowski, C.R.N. Tassi, M. Ali, J. Lee, M. Humt, J. Feng, A. Kruspe, R. Triebel, P. Jung, R. Roscher, M. Shahzad, W. Yang, R. Bamler, X.X. Zhu, A survey of uncertainty in deep neural networks, Artif. Intell. Rev. 56 (2023) 1513–1589.

[41] F. Wang, Y. Liu, K. Liu, Y. Wang, S. Medya, P.S. Yu, Uncertainty in graph neural networks: A survey, Preprint arXiv:2403.07185, 2024.

[42] R.C. Nickerson, U. Varshney, J. Muntermann, A method for taxonomy development and its application in information systems, Eur. J. Inf. Syst. 22 (2013) 336–359.

[43] C. Shorten, T.M. Khoshgoftaar, A survey on image data augmentation for deep learning, J. Big Data 6 (2019).

[44] D. Hendrycks, K. Gimpel, A baseline for detecting misclassified and out-of-distribution examples in neural networks, in: Int. Conf. Learning Represent, 2017.

[45] F.Z. Poleto, J.M. Singer, C.D. Paulino, Missing data mechanisms and their implications on the analysis of categorical data, Stat. Comput. 21 (2011) 31–43.

[46] M.C. Tremblay, K. Dutta, D. Vandermeer, Using data mining techniques to discover bias patterns in missing data, J. Data Inf. Qual. 2 (2010) 1–19.

[47] A.Y. Nahm, S.S. Rao, L.E. Solis-Galvan, T.S. Ragu-Nathan, The Q-sort method: Assessing reliability and construct validity of questionnaire items at a pre-testing stage, J. Mod. App. Stat. Methods 1 (2002) 114–125.

[48] J. van der Waa, E. Nieuwburg, A. Cremers, M. Neerincx, Evaluating XAI: A comparison of rule-based and example-based explanations, Artif. Intell. 291 (2021) 103404.

[49] D. Szopinski, T. Schoormann, D. Kundisch, Because your taxonomy is worth it: Towards a framework for taxonomy evaluation, in: Proc. of the 27th Eur. Conf. on Inf. Syst. (ECIS), 2019.

[50] H. Gimpel, D. Rau, M. Röglinger, Understanding FinTech start-ups – A taxonomy of consumer-oriented service offerings, Electron. Mark. 28 (2018) 245–264.

[51] K. Brosnan, B. Grün, S. Dolnicar, Cognitive load reduction strategies in questionnaire design, Int. J. Mark. Res. 63 (2021) 125–133.

[52] H. Sharma, How short or long should be a questionnaire for any research? Researchers dilemma in deciding the appropriate questionnaire length, Saudi J. Anaesth. 16 (2022) 65–68.

[53] K.L. Gwet, Handbook of Inter-rater Reliability: The Definitive Guide to Measuring the Extent of Agreement among Raters, Advanced Analytics, LLC (2014).

[54] J. Cohen, A coefficient of agreement for nominal scales, Educ. Psychol. Meas. 20 (1960) 37–46.

[55] T. Byrt, J. Bishop, J.B. Carlin, Bias, prevalence and kappa, J. Clin. Epidemiol. 46 (1993) 423–429.

[56] J.R. Landis, G.G. Koch, The measurement of observer agreement for categorical data, Biom. 33 (1977) 159–174.

[57] B. Heinrich, D. Hristova, M. Klier, A. Schiller, M. Szubartowicz, Requirements for data quality metrics, J. Data Inf. Qual. 9 (2017) 1–32.

[58] C. Cappiello, F. Cerutti, C. Sancricca, R. Zanelli, About the effects of data imputation techniques on ML uncertainty, in: Joint Workshops at 49th Int. Conf. on Very Large Data, 2025

[59] W.A. Wright, Bayesian approach to neural-network modeling with input uncertainty, IEEE Trans. Neural Netw. 10 (1999) 1261–1270.

[60] U. Draisbach, F. Naumann, DuDe: The duplicate detection toolkit, in: Proc. Int. Workshop on Qual. in Databases, 2010.

9 Supplement

Supplementary data to this article can be found online at https://doi.org/10.1016/j.dss.2025.114493.

2.2 Paper 2: Theoretical Foundations for Data Quality – Disentangling Accuracy and Currency

Current Status	Full Citation
This paper is **under review** (since 02/2026) for publication in the journal *ACM Transactions on Management Information Systems*	Heinrich, B., Klier, M., Krapf, T., & Schiller, A. (2026). Theoretical foundations for data quality – Disentangling accuracy and currency. *Working Paper*, University of Regensburg

Abstract:

Data quality is a field of ever-increasing relevance in research and practice, especially in fields such as AI-driven decision-making. Despite several seminal works, discussions in data quality research are still hampered by a lack of coherent theoretical foundations and data quality fundamentals, e.g., definite definitions for pivotal data quality dimensions such as the currency of data. To address these issues, we contribute to both formal theoretical foundations for data quality and precise specifications of the central data quality dimensions accuracy and currency. On this basis, a rigorous examination of the relation between accuracy and currency becomes possible. We show that our foundations do not only cover data quality defects discussed in literature but can also be used to identify additional relevant data quality defects not yet recognized by previous works. Further, our approach reveals causes and patterns of data quality defects, supporting future work such as the development and evaluation of data quality metrics and the detection of data quality defects. This is particularly important in AI-driven decision-making.

Keywords: Data quality, data quality defects, theoretical foundations, accuracy, currency

Please note that the overall formatting of the paper has been adjusted to the remainder of the dissertation.

1 Introduction

In recent years, the importance of data has increased significantly, both as a research subject for scholars and as an asset for organizations [32, 95]. In companies, for instance, large amounts of data play an ever more crucial role in decision-making, ideally helping them to gain a competitive advantage [23, 47, 73, 86]. However, poor data quality (DQ) still prevents companies from generating the best possible value from their data [7, 39, 42]. This is particularly true for the growing field of AI-driven decision-making [15, 30, 93]. According to a recent study, only a minority of AI projects make it into production, with "data quality and readiness" being cited as the primary obstacle [46]. Consequently, companies spend most of their data analytics efforts for assuring DQ, often about 80% of the working time [36]. Indeed, DQ has long been identified as a crucial success factor for decision support systems and a particularly relevant field for research and practice [1, 28, 42, 44, 63, 64, 69, 74, 78, 91].

Literature has yielded many renowned contributions to the field of DQ, delivering important theoretical and practical insights. For instance, first works aiming to establish theoretical foundations for DQ and DQ dimensions have been proposed [2, 56, 82, 90]. Based on and extending these works, a multitude of general frameworks and methodologies have shaped DQ research [13, 24]. Besides these contributions establishing a broader view on DQ, further works have focused on particular DQ issues: For instance, Parssian et al. [76, 77] analyzed the effect of database operators on DQ, in particular accuracy, Heinrich et al. [42] proposed an approach for duplicate detection based on duplicate-related events, Cappiello et al. [20] shed light on the assessment and management of accuracy, currency, and completeness, and Peng et al. [78] reviewed methods and assumptions regarding the handling of incomplete data. Similarly, Bai et al. [8] and Krishnan et al. [57] focused on assessing and managing DQ in the context of accounting IS and Liu et al. [65] in the context of cooperative IS. Nelson et al. [71] identified accuracy, currency, completeness, and format as key DQ dimensions and analyzed their impact on task-dependent information and system satisfaction, which has frequently been taken up since then (e.g., [3]).

Despite these remarkable works in the field of DQ, research also points out that fundamentals are still defined in an incoherent way [40, 87, 98] and the prominent call by Lee [59] that "data quality researchers need to build a deeper and broader theoretical foundation" may still not be answered sufficiently. Probably, this is because of the fact that DQ is understood as multidimensional [90], with central dimensions such as accuracy and currency being defined and used in different ways [13, 22, 49]. The underlying reasons may even lie deeper, as many theoretical foundations are informal, making precise definitions of DQ dimensions as a central research subject difficult. This problem is particularly evident in the case of currency, one of the most central DQ dimensions [41, 49, 97]. Indeed, existing definitions of currency are even incompatible. For example, while Redman [83] defines currency as the "degree to which a datum in question is up to date", Sicari et al. [88] refer to currency as "the interval from the time when the value was sampled to the time instant at which data are received", and Wand and Wang [90] view

currency as a subdimension of timeliness, defining it as the passed time until an IS is updated after a real-world change. At the same time, several terms such as currency [83], timeliness [81], and data staleness/freshness [22] are used to express a similar intent (cf. also [40]). Hereby, based on existing theoretical foundations, it is difficult to rigorously assess different definitions of time-related DQ dimensions regarding similarities and differences in their meaning. The advantages of formal and coherent DQ fundamentals as envisioned by the paper at hand also become apparent when considering the relation between DQ dimensions such as accuracy and currency. For example, Zak and Even [96] point out that "as data becomes less current it is also likely to become less accurate" and Scannapieco et al. [85] state that "accuracy and currency are very similarly perceived". Precise formal definitions of accuracy and currency would enable a rigorous examination of this frequently discussed relation.

By introducing such precise formal definitions, our work provides the foundation for explanatory and predictive theories of DQ (in the sense of [34]) and thus contributes to cumulative theory-building in IS. Indeed, these definitions are necessary to explain why DQ degrades, and to predict the impact of poor DQ. This is of particular relevance for real-world applications, for example in AI-driven decision-making and machine learning (ML) [37]. In this regard, the quality of analyzed data and improving inaccurate and outdated data by systematically addressing DQ defects are of overwhelming relevance. For example, as discussed below, addressing currency defects based on our theoretical foundations can substantially improve ML performance. This is demonstrated by a small example application involving a well-known challenging and high-stakes healthcare task.

Thus, we aim at formal and coherent theoretical foundations for DQ comprising definitions which enable a precise specification of DQ dimensions. Thereby, we focus on accuracy and currency and take the intrinsic perspective of DQ (i.e., "quality in its own right" ([60], cf. also [92])) to allow a deeper discussion of the provided theoretical foundations. We focus on the following research question:

How can the central DQ dimensions accuracy and currency be specified in a precise manner based on formal and coherent theoretical foundations for intrinsic DQ?

By addressing this research question, we do not aim to propose another framework comprising a list of DQ dimensions to address the multidimensionality of DQ (e.g., [14, 16, 19]). Rather, our work aims to make the following contributions: (1) We propose formal and coherent theoretical foundations for DQ; (2) on this basis, we define precise specifications for accuracy und currency; (3) we present a rigorous examination of the relation between accuracy and currency; and (4) we map DQ defects related to accuracy and currency, which are discussed in existing literature, to our specifications and additionally identify novel DQ defects which have not yet been examined in literature.

The remainder of the paper is structured as follows: First, we position our work in literature and discuss existing contributions on DQ fundamentals ending up in the research gap to be addressed. Next, we introduce our theoretical foundations for DQ and specify the DQ dimensions accuracy

and currency. Then, we conduct both a literature-based analysis and a mathematical analysis regarding our specified DQ dimensions. In the penultimate section, we discuss the results of our work and outline implications for research and practice. Finally, we summarize, reflect on limitations and present an outlook on future research.

2 Background and Related Work

In literature, a basic distinction is made between a subjective perspective on DQ (i.e., interpretations and evaluations of individuals) and an objective perspective on DQ (i.e., reflecting factual states of the real world) [31, 81, 90]. This distinction widely corresponds to general epistemological theories, where (subjective, individual) interpretations and beliefs are contrasted to facts and knowledge, which are seen as "truth—that is, it is certain, congruent with reality and accurate" ([75], cf. also [17]). The latter implies a grounding on the "objectivist approach" [17], which "assumes the existence of an objective reality" [25] and facts [25, 45]. In this regard, data itself can be seen as a) a representation of subjective interpretations and beliefs of individuals which require context and further processing, or b) a representation of objective, reproducible facts [11, 12, 61]. Thereby, data representing both, subjective interpretations or objective facts can be perfect or exhibit defects [18]. The quality of this data corresponds to DQ, and defects in the data are DQ defects.

Following this general distinction, DQ is often differentiated into intrinsic DQ and extrinsic DQ (e.g., contextual DQ), as introduced by Wang and Strong [92] and frequently referenced since then [cf., e.g. 13, 31, 60]. Intrinsic DQ follows the perspective that data has "quality in its own right" [60]. In this respect, DQ focuses on the quality of the data representing objective, reproducible facts. For example, if the last name of a real person is stored in a database, this data can either be accurate or inaccurate (accuracy as intrinsic DQ, cf., e.g., [60]). This is independent from subjective interpretations and evaluations of individuals and the context or task the stored last name is used for. Thereby, the facts in the real or digital world constitute a ground truth, regardless of whether the data representing these facts is known under certainty or uncertainty. In contrast, an extrinsic perspective on DQ, such as contextual DQ, mainly puts the users' interpretations in the center of view [60]. Here, DQ is determined regarding certain tasks, data's preparation and understandability for individuals or its availability for these tasks. Extrinsic DQ is thus task- and context-dependent. For instance, if customers are contacted only via email, the DQ defect that their postal code is stored as inaccurate data in an IS (accuracy as intrinsic DQ) does not seem to be relevant for this specific task (relevance as contextual DQ, cf., e.g., [60]). There are several seminal works focusing on the extrinsic perspective and the impact of perceived DQ, for instance on the success of IS [80, 94] and different aspects of extrinsic DQ of user-generated content [5, 6, 66, 67].

Literature already provides first works aiming to establish theoretical foundations for DQ and DQ dimensions. Apart from works concentrating, for instance, on contextual DQ (e.g., [51]), four major contributions include intrinsic DQ, and are the focus of the following discussion:

Kon et al. [56] aim to lay foundations for analyzing data and DQ via the examination of DQ defects related to data acquisition and production. By analyzing and modeling both data and DQ in an IS, they share a data-oriented view. However, their understanding of DQ does not refer to the representation of entities by data (cf. Definition 1) and thus differs substantially. Indeed, their foundations are based on the process of a structured and quality-oriented data production. More precisely, their specifications of DQ dimensions refer to the agreement of the "actual" process of data production to an "ideal" process but not to the representation of entities by data. Moreover, as this "ideal" process is described as task-dependent, a clear distinction between intrinsic DQ and extrinsic DQ is not emphasized.

Aebi and Perrochon [2] aim to specify DQ in a formal manner to serve as a sound starting point for their research on data improvement. Similar to our perspective, they define foundations for elaborating the agreement between data in an IS and a model of the real world, called "reality-view" [2]. However, as the scope of this "reality-view" is described as task-dependent, a clear distinction between intrinsic DQ and extrinsic DQ is not in focus. Moreover, incorporating the data schema view in their foundations, Aebi and Perrochon [2] do not adopt a detailed data-oriented view. Thus, the data in their model are not structured in more detail. In particular, the structural characteristics of data (comprising entities, features and feature values), which play a crucial role in our data-oriented view, are not focused.

In their seminal work, Wand and Wang [90] establish ontological foundations for DQ, focusing on intrinsic DQ. Their central notion of "proper representation" [90] establishes a relation between possible states of the real world and states of the IS as a blueprint for data production. This notion is independent from the actual data presented by the IS but refers to IS design, thus not taking a detailed data-oriented perspective. Indeed, three of the four DQ dimensions specified by Wand and Wang [90] arise from an analysis of IS "design deficiencies" [90]. As a consequence, these dimensions only cover DQ defects during data production that result from a deficient IS. For example, based on the foundations by Wand and Wang [90], meaningless states of an IS only contribute to poor DQ if they occur during data production.

Price and Shanks [82] use semiotic theory as foundation to derive a framework comprising DQ categories and respective DQ dimensions. Based on the syntactic, semantic and pragmatic levels distinguished in semiotics, they address both the intrinsic (including syntactic and semantic level) and the extrinsic (pragmatic level) perspective on DQ. They aim at an extensive framework where "the set of criteria should be comprehensive" [82]. Given this broad and ambitious approach, they neither strive for nor provide formal and coherent definitions of theoretical foundations, as we do. Moreover, while discussing real-world phenomena and data in IS and acknowledging a data-oriented view, their approach does not deal with structural characteristics of data.

To sum up, literature provides valuable and important theoretical foundations for DQ. Indeed, while these papers are two or more decades old, they are still frequently cited and highly relevant for current DQ research. At the same time, several works call for further fundamentals for DQ [40, 49, 50, 87, 98]. To contribute to this research gap, we aim at theoretical foundations which (a) are coherent and formally defined and are based on an intrinsic perspective on DQ, (b) take

a data-oriented view, and (c) include detailed structural characteristics (i.e., comprising entities, features and feature values) of the considered data. These theoretical foundations for DQ (cf. Contribution (1)) enable to define precise specifications for accuracy und currency (cf. (2)), rigorously examine the relation between accuracy and currency (cf. (3)), map DQ defects related to accuracy and currency, which are discussed in existing literature, to our specifications, and identify novel DQ defects related to accuracy and currency which have not yet been examined in literature (cf. (4)).

3 Theoretical Foundations for Data Quality and Data Quality Dimensions

In this section, we introduce our theoretical foundations based on which we specify the DQ dimensions accuracy and currency. To motivate our theoretical foundations and demonstrate their benefits, we introduce a real-world healthcare case. We use this healthcare example throughout the paper to illustrate the concepts and definitions from our theoretical foundations.

3.1 Motivation and Running Example

In healthcare, a difficult but at the same time high-stakes challenge in hospitals is assessing the severity of illnesses and injuries to properly prioritize, react, and treat (triage). This challenge is of particular importance in intensive care unit (ICU) settings, where the quality of decision-making often means the difference between life and death. Nowadays, this challenge is increasingly supported by the application of ML (e.g., [35, 44, 52]). Specifically, the task of mortality prediction as described by Harutyunyan et al. [38] based on the MIMIC dataset [53] has received great attention. Here, mortality predictions are made based on data collected during the first 48 hours of a patient's ICU stay. Data collected include Glasgow coma scale data, typically used to assess the level of consciousness of patients, as well as the values of clinical features such as temperature and glucose levels. We consider an excerpt of the MIMIC dataset (cf. Table 1) to serve as running example, representing data from a hospital's medical healthcare database. The excerpt shows data referring to ICU patients.[1]

Patient First Name	Patient Last Name	Clinical Diagnosis	Medication	Temperature	Glucose	Weight
Mary	Smith	Diabetes	Insulin	36.5	202	60
John	Stones	RF	Propofol	37.7	145	80
…	…	…	…	…	…	…

Table 1. Data Excerpt Medical Healthcare Database

The collected data in the MIMIC dataset, which forms the basis for ML training, is subject to substantial DQ defects as already discussed by e.g. Cross et al. [26], Harutyunyan et al. [38], and Sauer et al. [84]. For example, the capturing of patient data is usually intermittent rather than

[1] To enhance readability in the paper, we only consider a simplified set of selected features and use aliases as first and last names of patients instead of the anonymous identifiers provided in the MIMIC dataset.

continuous, occurring at specific points in time, such as when a care person enters the room, leading to outdated data values. In particular, if drugs are administered that strongly alter the values of clinical features, the recorded (outdated) values may differ strongly from the feature values in the real world; however, this aspect is usually not considered in existing works (cf., e.g., [38]). As an illustrative application of our insights, we have modeled a few examples of such drug administration effects based on our theoretical foundations and our definition of currency. As our results in the Section Discussion and Implications show, addressing currency defects in this small example has already improved ML performance in the mortality prediction task. This is evidenced by a recall improvement of 5.4 percentage points (~15% relative increase) and a precision improvement of 7.2 percentage points (~12% relative increase). Both indicate not only a positive effect enabled by thorough consideration and systematic addressing of DQ defects but also a relevant effect on decision-making in ICU settings.

3.2 Theoretical Foundations

The paper at hand focuses on intrinsic DQ (objective perspective) and can serve as a basis for extensions to extrinsic DQ in the future. In this regard, we define data as a representation of entities constituting objective, reproducible facts (cf. [11, 12, 61]) and adopt the following definition of DQ [42, 76, 77].

Definition 1 (Data Quality). Both real-world entities (e.g., persons) and digital entities (e.g., virtual concepts) can be represented by data (e.g., by a data record, storing the person's name and address). This data representation can be perfect or imperfect, i.e., exhibit defects [18], which corresponds to DQ [76, 77].[2]

Regarding Definition 1, we are interested in representations of entities by data and, in particular, whether these representations contain specific DQ defects such as a "Data entry error" which can be attributed to a certain DQ dimension like accuracy. We will define these terms and concepts more closely, starting to introduce the entities and data considered in Definition 1.

Definition 2 (Entity, Entity Space, Data Space). A real-world or digital entity e_i (with $i \in \{1, 2, \dots, I\}$) exhibits features f_{j_i} and respective feature values $f_{j_i}(e_i)$ (with $j_i \in \{1, 2, \dots, J_i\}$). The set of all entities constitutes the entity space $E = \{e_1, \dots, e_I\}$. Analogously, the set of all data representations of entities forms the data space $D = \{d_1, \dots, d_K\}$. A data representation d_k also exhibits features and respective feature values, which we denote by $f_{l_k}^D$ and $f_{l_k}^D(d_k)$.

In our running example, the objective fact "The clinical diagnosis for the ICU patient Mary Smith is pneumonia" is constituted by the entity $e_i = \text{"Mary Smith"}$, the feature $f_{j_i} = \text{"Clinical diagnosis"}$ and its feature value $f_{j_i}(e_i) = \text{"Pneumonia"}$. However, this entity is actually represented by data $d_k \in D$ (e.g., a digital patient record) referring to Mary Smith, the clinical diagnosis $f_{j_k}^D$ and the data value $f_{j_k}^D(d_k) = \text{"Diabetes"}$ (cf. Figure 1). This means that at least

[2] Our understanding of the relation between data, entity and fact is grounded on an "objectualist" view of facts where entities and their *features and feature values* (objects) can be "constituents of facts" (Fine 1982, p.65).

one DQ defect is present, as the feature value is imperfectly represented by the respective data. A comprehensive systematization of such (types of) DQ defects related to the DQ dimensions accuracy and currency is the central objective of our specifications of these dimensions. This, in turn, requires the underlying theoretical foundations.

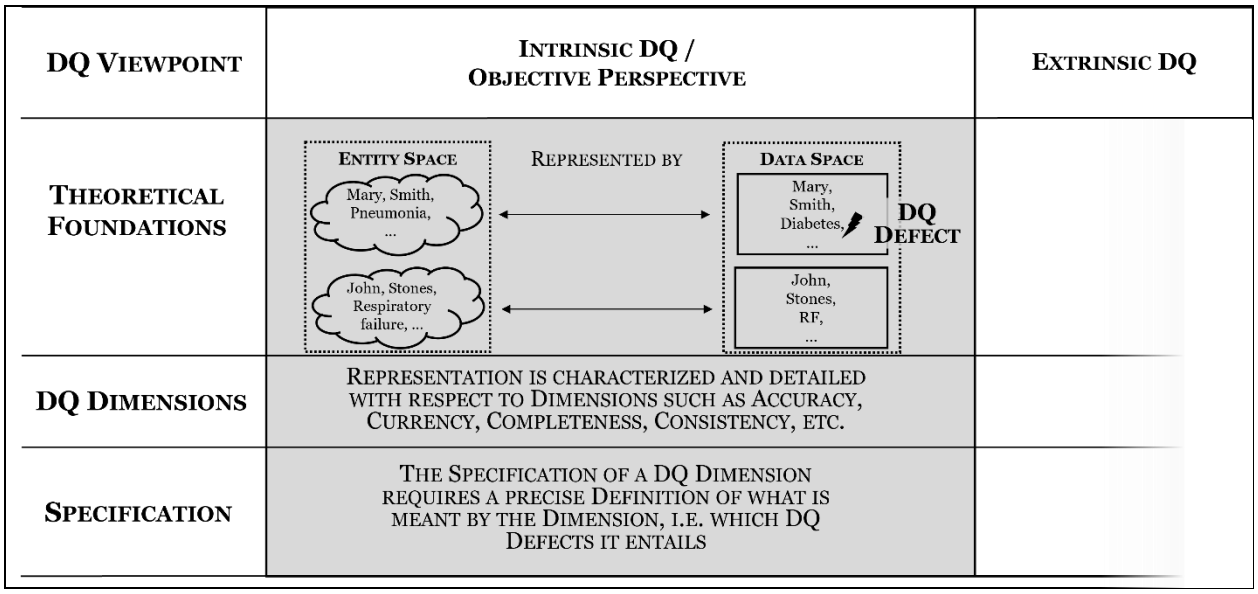


Figure 1. Nexus of Data Quality Fundamentals (Focus of this Paper Shaded in Gray)

Figure 1 further serves to illustrate our understanding of DQ and position our work within the nexus of DQ fundamentals (focus shaded in gray). Our theoretical foundations model the entities and data as well as their structural characteristics (cf. Definition 2) focusing on an intrinsic perspective on DQ. A perfect or imperfect representation of entities by data defines the core of our understanding of DQ (Definition 1), and thus whether DQ defects are present, which in turn can be assigned to specific DQ dimensions. Theoretically, the set of all DQ defects corresponds to all imperfect representations of entities by data. For instance, a data transmission error can cause an inaccurate capturing of a person's clinical diagnosis as data in an IS (cf. "Diabetes" instead of "Pneumonia" regarding the entity "Mary Smith" in Figure 1). Thereby, specifications of the DQ dimensions accuracy and currency make this definition of DQ and DQ defects tangible.

In literature, the definition of DQ and DQ dimensions is often closely related to the measurement of DQ. Yet, there is a vital distinction between the specification of different DQ dimensions and the way these specifications are measured. Specifying a DQ dimension based on Definition 1 requires a precise definition of what is meant by the dimension, i.e., which (types of) DQ defects are contained by the dimension. In contrast, the measurement of a specified DQ dimension demands a precise definition of how it is assessed (e.g., by means of a DQ metric) referring to the underlying specification and in particular, the (types of) DQ defects it aims to measure.

Our understanding of DQ is grounded in the representation of an entity e by data d, e.g., within an IS.[3] In order to rigorously examine such representations, we next introduce a representation function. For this definition, we closely follow Heinrich et al. [42] who focus on duplicate detection and therefore extensively address the relationship between entities and data.

Definition 3 (Representation Function). The function $\psi: D \rightarrow E \cup \{0\}$ describes how each data representation $d \in D$ is mapped to the corresponding entity it represents or to 0 (as an arbitrary placeholder, with $0 \notin E$) if d does not represent any entity in E.[4] Analogously,[5] the function $\psi^F: F^D \rightarrow F \cup \{0\}$ describes how each data representation feature f^D is mapped to the feature it represents or to 0 (as an arbitrary placeholder, with $0 \notin F$) if f^D does not represent any feature in F.

When specifying currency as DQ dimension, we have to reflect that entities and data can change over time.

Definition 4 (Time). We consider a time interval $[0, T]$ with $T > 0$ and a discretization in $\mathcal{H} \in \mathbb{N}$ intervals with time step $h = \frac{T}{\mathcal{H}}$. This leads to a finite number of considered points in time $0 = T_0 < T_1 < \cdots < T_{\mathcal{H}-1} < T_{\mathcal{H}} = T$ with $T_i = i * h$. For reasons of simplicity, we also write $t \in [0, T]$ instead of $t \in \{T_0, T_1, \dots, T_{\mathcal{H}}\}$ and $t \in [h, T]$ instead of $t \in \{T_1, T_2, \dots, T_{\mathcal{H}}\}$ in the following. When explicitly considering a certain point in time $t \in [0, T]$, we add an index t to the notation of an entity e, a data representation d, a feature f, etc. (i.e., we write e_t, d_t, f_t etc.).

Naturally, data can represent the same entity at different points in time t, which is vital for the DQ dimension currency. To clarify what is meant by "represent the same entity", we introduce a definition for identity. In this definition, we expand the understanding of Heinrich et al. [42] by additionally considering time and the identity of features.

Definition 5 (Identity). For a pair of a data representation d_{t_a} at the point in time t_a and an entity e_{t_b} at the point in time t_b (with $t_a, t_b \in [0, T]$, where $t_a = t_b$ is the standard case, but not required), identity holds if and only if for the representation function ψ (cf. Definition 3) it holds: $\psi(d_{t_a}) = e_{t_b}$. Analogously, identity holds for a pair of a feature $f_{t_a}^D \in F^D$ at t_a and a feature $f_{t_b} \in F$ at t_b if and only if it holds $\psi^F(f_{t_a}^D) = f_{t_b}$.

The definition of identity for features enables the identification of which data represents a certain feature, which is important both for the specification of a DQ dimension as well as for its subsequent measurement (e.g., to estimate the probability that a stored data representation corresponds to a particular real-world entity and is accurate with respect to it). In our running example, ψ^F

[3] A further breakdown of entities (e.g., an entity can consist of subentities) is possible but would only introduce additional complexity to the notation. Instead, in our definitions, relational aspects (e.g., specializations or aggregations such as an entity being part of another entity) appear implicitly.

[4] A data representation $d \in D$ does not represent any entity in E and maps to 0 (as an arbitrary placeholder), if a corresponding entity (e.g., a person) in E does not exist (anymore).

[5] Similarly to E and D, we make the following definitions: By $F = \{f_{1_1}, \dots, f_{J_1}, f_{2_1}, \dots, f_{J_I}\}$, we denote the feature space (comprising all features across all entities in E) and by F^D, we denote the data representation feature space.

maps the feature "Temperature" to the body temperature of the patient and not, e.g., to the temperature of the hospital room or the outside temperature. An additional definition of identity for pairs of feature values, however, is not required, as it is covered by the definition of identity for their respective features.

While the **representation function** ψ establishes the fundamental concept of identity by linking data to what it represents (cf. [42]), it does not establish how exactly this representation is (syntactically and semantically) realized, and in particular, whether it is perfect or imperfect. In other words, ψ ensures that given data actually represents a considered entity or feature, but it does not determine the syntax and semantics of that representation.

For this purpose, we extend the **representation function** ψ **and** introduce the relation δ_A between entities, their features and feature values, and the data representing them, enabling a matching on the level of syntax or semantics. Grounded on the objectivist approach, the (testable) relation δ_A formalizes the data representation in a *syntactical* and/or *semantical* manner. This is similar to other research fields such as ontology matching, where data has to be matched regarding syntax and semantics [27]. The relation δ_A links entities, features, and feature values, thus making their matching tangible.

We characterize δ_A as a *syntactical mapping* if δ_A operates on a structural and/or character level. For instance, δ_A as a syntactical mapping can consider data types and assess string-based data representations of the entity state space S[6] in terms of the used characters or other linguistic rules such as grammar [21]. For example, if δ_A refers to capitalization rules used for data (e.g., "A" instead of "a" at the beginning of proper names), δ_A is regarded a syntactical mapping. We characterize δ_A as a *semantical mapping* if δ_A factors in the meaning of the elements of entity state space and data state space (according to [33]), for instance, equivalence in meaning (e.g., synonyms), regardless of the syntax. For example, δ_A as a semantical mapping incorporates the equivalence in meaning of a feature value "Respiratory failure" and data "RF" as a convention/abbreviation, or the measurement of "100 degrees Fahrenheit" as "37.7 degrees Celsius" (cf. running example). Further examples are presented in Table 2.

[6] Analogously to E and F. by $FV = \{f_{1_1}(e_1), \dots, f_{J_I}(e_I)\}$ we denote the feature value space (across all entities in E and their respective features), and by FV^D, the data representation feature value space. On this basis, we introduce the entity state space $S = E \cup F \cup FV$ as union of the entity space, feature space, and feature value space. Analogously, we define the data state space as $S^D = D \cup F^D \cup FV^D$.

Representation Type	Representation Function	Relation	Example
Entity e by data d	$\psi(d) = e$	$\delta_A^e(e,d) = true$	A patient (the person) is perfectly represented in a syntactical and semantical manner
Entity e by data d	$\psi(d) = e$	$\delta_A^e(e,d) = false$	An inpatient is semantically imperfectly represented as already discharged by being stored in the wrong database relation, since the DQ defect "Coded wrongly or not conform to real entity" occurs
Feature f by data f^D	$\psi(d) = e,$ $\psi^F(f^D) = f$	$\delta_A^f(f, f^D) = true$	The feature "ZIP code" in the address data of the patients captures their postal code, thus perfectly representing the feature in a syntactical and semantical manner
Feature f by data f^D	$\psi(d) = e,$ $\psi^F(f^D) = f$	$\delta_A^f(f, f^D) = false$	The feature "Name" lacks semantically clear information that it refers to the last name of the patient, thus the DQ defect "Ambiguous data" occurs
Feature value $f(e)$ by data $f^D(d)$	$\psi(d) = e,$ $\psi^F(f^D) = f$	$\delta_A^{fv}(f(e), f^D(d)) = true$	The respiratory failure of a patient is syntactically and semantically accurately stored by the value "RF" as conventional abbreviation
Feature value $f(e)$ by data $f^D(d)$	$\psi(d) = e,$ $\psi^F(f^D) = f$	$\delta_A^{fv}(f(e), f^D(d)) = false$	The actual temperature of a patient is 39 degrees, but is semantically imperfectly stored as 38 degrees, since the DQ defect "Data entry error" occurs

Table 2. Interplay Between Representation Function and Syntactical/Semantical Mapping

Figure 2 illustrates entities and data representations that hold identity (cf. arrows drawn between entities $e_{i,t}$ and data representations $d_{i,t}$ with $\psi(d_{i,t}) = e_{i,t}$) for two points in time $t_0 < t_1$ and $t \in \{t_0, t_1\}$. Note that the data at t_0 illustrate the records in Table 1. The data at t_1 is to be understood as a later version of the hospital's database.

For the point in time t_0, the actual value of the feature "Glucose" for Mary Smith in the entity state space is 152. However, because of a measurement error, an erroneous value of 202 was captured and stored (cf. data state space). This violates δ_A^{fv} to provide a semantically accurate mapping, meaning that these values constitute an accuracy defect. As a result, Mary Smith is erroneously diagnosed as suffering from Diabetes, leading to the feature value of the feature "Clinical diagnosis" for "Mary Smith" being "Diabetes" in the data state space (DQ defect: erroneous value). This does not accurately represent the entity's feature value, as Mary is in fact ill with a pneumonia at t_0, again leading to a violation of δ_A^{fv} to provide a semantically accurate mapping. Consequently, this constitutes a DQ defect related to accuracy for the entity "Mary Smith" (at t_0). Between the points in time t_0 and t_1, an event occurs: The pneumonia of "Mary

Smith"—which was not accurately captured at t_0—progresses severely, resulting in a critical increase of her temperature from 36.5 to 39.2 in the entity state space. The other feature values of "Mary Smith" remain unchanged. Thus, at the later point in time t_1, the representation of the temperature of "Mary Smith" has become outdated: Although the value for her temperature was captured accurately at t_0, this no longer holds at t_1 (violation of δ_A^{fv} to provide a semantically accurate mapping). Indeed, the event of the pneumonia progression caused a change in the entity state space and specifically the temperature's feature value, but the feature value in the data state space was not updated accordingly, rendering it outdated.

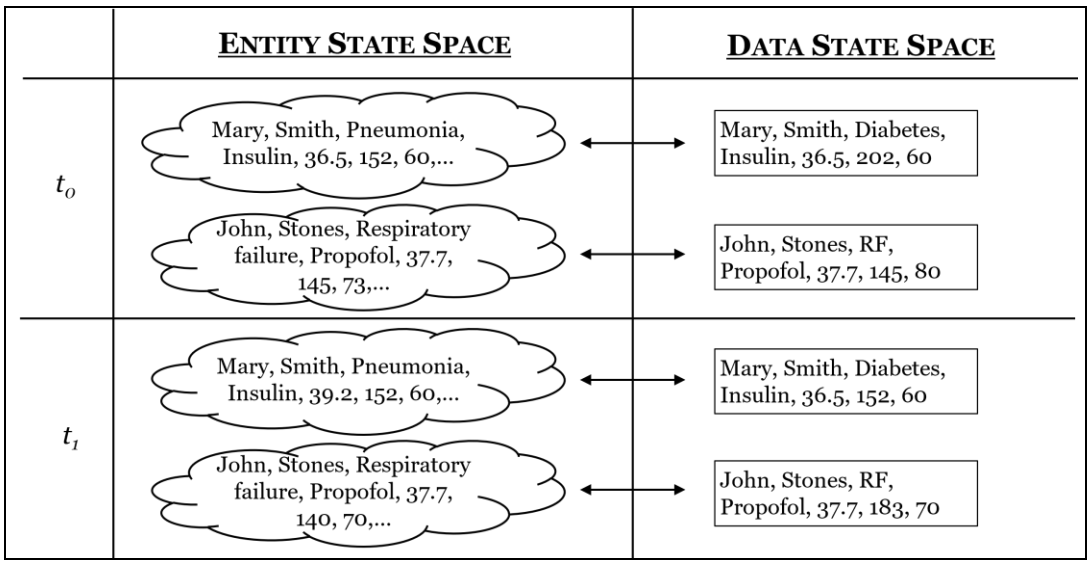


Figure 2. Illustration of Entities and Data Representations (Example)[7]

To identify and systematize such DQ defects, we introduce DQ-related events in our theoretical foundations (e.g., the severe progression of the diagnosed pneumonia) that induce changes in the entity state space and/or the data state space, potentially leading to DQ defects.

Definition 6 (Event Space, Event). The event space for the time interval $[0, T]$ is denoted by Ω, describing the set of all DQ-related events $\omega \in \Omega$ in $[0, T]$.

Definition 7 focuses on event-based state transitions pivotal for studying time-related DQ defects.

Definition 7 (Event-based State Transition). At each point in time $t \in [h, T]$ a DQ-related event $\omega \in \Omega$ can lead to a change of the entity state space S_t and/or the data state space S_t^D (e.g., it leads to $S_{t-h} \neq S_t$ and/or $S_{t-h}^D \neq S_t^D$). This change is called an event-based state transition. Each event $\omega \in \Omega$ induces the (possibly empty) sets $s_t^- \subset S_{t-h}$, $s_t^+ \subset S_t$, $s_t^{d,-} \subset S_{t-h}^D$ and $s_t^{d,+} \subset S_t^D$. An element of S_{t-h} or S_{t-h}^D at the point in time $t - h$ is element of the subsets s_t^- or $s_t^{d,-}$ if it does not exist anymore at the point in time t. An element of S_t or S_t^D at the point in time t is element of the subsets s_t^+ or $s_t^{d,+}$ if it did not yet exist at the point in time $t - h$. Each event-based state transition can be described using three basic transition types:

[7] For reasons of readability, feature values are listed without denoting the associated features.

i. Add new states: $S_t = (S_{t-h} \uplus s_t^+ | \exists \omega)$ and/or $S_t^D = (S_{t-h}^D \uplus s_t^{d,+} | \exists \omega)$

ii. Delete existing states: $S_t = (S_{t-h} \setminus s_t^- | \exists \omega)$ and/or $S_t^D = (S_{t-h}^D \setminus s_t^{d,-} | \exists \omega)$

iii. Change existing states: $S_t = (((S_{t-h} \setminus s_t^-) \uplus s_t^+) \mid \exists \omega)$ and/or $S_t^D = (((S_{t-h}^D \setminus s_t^{d,-}) \uplus s_t^{d,+}) \mid \exists \omega)$

For simplicity of notation, we abstain from introducing a time index when denoting events. It is sufficient to consider the points in time of the sets s_t^+ and s_t^- as well as $s_t^{d,+}$ and $s_t^{d,-}$ induced by the event. Moreover, this definition may easily be generalized to include state transitions caused by multiple events. Since state transitions are always induced by events, if no event occurs in the time period $[t-h, t]$, the state spaces at $t-h$ and t coincide.

To illustrate Definition 7, we use our running example. The event $\omega_1 \in \Omega$ of the severe progression of the pneumonia of "Mary Smith" could lead to another event $\omega_2 \in \Omega$, the change of her medication from "Insulin" to "Levofloxacin", which changes the feature value of the feature "Medication", where $s_t^- = \{\text{"Insulin"}\}$ and $s_t^+ = \{\text{"Levofloxacin"}\}$ (cf. Definition 7 iii.). If the event $\omega_2 \in \Omega$ is not captured by the hospital staff and the database is not updated as a result (i.e., $s_t^{d,+} = s_t^{d,-} = \emptyset$), this could dramatically impede further treatment of Mary Smith, since the current medication is not stored in the database. Critically, this outdated state of the data could lead to a delayed or even forgotten administration of Levofloxacin to "Mary Smith", potentially with severe health-related consequences for the patient. Moreover, many other (types of) events and state transitions can be modeled as elements of the event space Ω and the entity and/or data state spaces S_t and S_t^D respectively, such as, e.g., the change of the clinical diagnosis of "Mary Smith" from Diabetes, which is erroneously updated as Influenza (instead of Pneumonia) in the IS, i.e., $s_t^{d,-} = \{\text{"Diabetes"}\}$ and $s_t^{d,+} = \{\text{"Influenza"}\}$ holds. Again, this inaccurate state of the data can have serious consequences, ranging from an ineffective distribution of medications in the ICU to no recuperation of "Mary Smith" from the pneumonia because of inadequate treatment.

3.3 Specification of the Data Quality Dimensions Accuracy and Currency

In DQ literature and our understanding, accuracy is not regarded as a time-related DQ dimension [13, 97]. Thus, when discussing accuracy, we do not explicitly deal with time (cf. Definition 4) and temporal elements such as event-based state transitions (cf. Definition 7) which will play a vital role when specifying the DQ dimension currency afterwards. Consequently, and following Definition 1, accuracy refers to the representation by data *at one single point in time.* On the basis of the relation δ_A discussed above, the DQ dimension accuracy is defined as follows:

Definition 8 (Accuracy). A **data representation of an entity/feature/feature value is accurate if** identity holds (cf. Definition 5) and the relation δ_A is obtained to be true.

In the following, we define the corresponding *accuracy specifications* for data referring to entities, features, and feature values in a formal and precise manner. These apply to any but fixed point in time $t \in [0, T]$.

Regarding entities, the specification of accuracy is defined as

$$\mathcal{S}_A^e: E_t \times D_t \to \{true, false\},$$
$$\mathcal{S}_A^e(e_t, d_t) = \begin{cases} true & , \text{if } \psi(d_t) = e_t \wedge \delta_A^e(e_t, d_t) = true \\ false & , else \end{cases} \quad \text{(I)}$$

where $\delta_A^e: E_t \times D_t \to \{true, false\}$ is a mapping as discussed above.[8]

To specify accuracy for data representation features, the accuracy of the corresponding data representations exhibiting the feature in the sense of Specification (I) is an obviously necessary precondition. Hence, the set $(F_t \times F_t^D)_A := \{(f_t, f_t^D) \in F_t \times F_t^D \mid \mathcal{S}_A^e(e_t, d_t) = true\}$ serves as a basis to extend the accuracy specification to features as follows:

$$\mathcal{S}_A^f: (F_t \times F_t^D)_A \to \{true, false\},$$
$$\mathcal{S}_A^f(f_t, f_t^D) = \begin{cases} true & , \text{if } \psi^F(f_t^D) = f_t \wedge \delta_A^f(f_t, f_t^D) = true \\ false & , else \end{cases} \quad \text{(II)}$$

Finally, to specify accuracy for data representation feature values, the accuracy of the underlying data representation features in the sense of Specification (II) is again an obviously necessary precondition. Hence, the set $(FV_t \times FV_t^D)_A := \{(f_t(e_t), f_t^D(d_t)) \in FV_t \times FV_t^D \mid \mathcal{S}_A^f(f_t, f_t^D) = true\}$ serves as a basis to extend the accuracy specification to feature values as follows:

$$\mathcal{S}_A^{fv}: (FV_t \times FV_t^D)_A \to \{true, false\},$$
$$\mathcal{S}_A^{fv}\left(f_t(e_t), f_t^D(d_t)\right) = \delta_A^{fv}(f_t(e_t), f_t^D(d_t)) \quad \text{(III)}$$

To highlight the diversity of accuracy defects, we continue with the example of the "Data entry error" from Table 2. Here, the actual temperature of 39 degrees of a patient was semantically imperfectly stored as 38 degrees. This is an accuracy defect according to Specification (III). If a nurse recognizes this accuracy defect and attempts to correct it but accidentally stores an additional patient record with the corrected temperature, a duplicate is created. This constitutes an additional DQ issue. In addition, there is still an accuracy defect, now in the form of contradicting values of the two records. More precisely, for both records, the data representing the temperature value can be assessed with Specification (III), but only one value is true with respect to δ_A^{fv} and thus does not exhibit an accuracy defect.

In contrast to accuracy, the DQ dimension currency is understood as time-related in DQ literature [13, 49, 90]. Hence, for the respective specification, the definitions for time and time-related

[8] Note that Specification (I) (and the following) can serve as a basis for both measuring accuracy in a world under certainty and a world under uncertainty. Grounded on Specification (I) different DQ metrics can be defined, where entities and data representations are known by certainty (i.e., e, d are known) or parts of e and d are unknown (e.g., e has to be indicated by probabilities). In this way, Specification (I) defines what is meant by accuracy, and not how it can be measured.

aspects (cf. Definitions 4, 6 and 7) have to be taken into account. The DQ dimension *currency* is defined as follows:

Definition 9 (Currency). For two points in time $t_0 < t_1$, a data representation of an entity/feature/feature value is current at t_1 if the data representation —which was indeed accurate at t_0— is still accurate at t_1.

In this sense, the *currency specifications* for entities, features, and feature values defined in the following are generally based on the respective accuracy specifications (cf. Specifications (I), (II) and (III)).

The set of entities and data representations for which currency is considered is given by $(E_{t_1} \times D_{t_1})_C^{t_0} \coloneqq \{(e_{t_1}, d_{t_1}) \in E_{t_1} \times D_{t_1} \mid \mathcal{S}_A^e(e_{t_0}, d_{t_0}) = true\}$ (cf. Definition 9). On this basis, the currency specification regarding entities is defined as follows:

$$\mathcal{S}_{C,t_0}^e : (E_{t_1} \times D_{t_1})_C^{t_0} \to \{true, false\},\ \mathcal{S}_{C,t_0}^e(e_{t_1}, d_{t_1}) = \mathcal{S}_A^e(e_{t_1}, d_{t_1}) \tag{IV}$$

To specify currency for data representation features, the currency of the corresponding data representations in the sense of Specification (IV) is an obviously necessary precondition. Hence, the set of pairs of features for which currency is considered is given by $(F_{t_1} \times F_{t_1}^D)_C^{t_0} \coloneqq \{(f_{t_1}, f_{t_1}^D) \in F_{t_1} \times F_{t_1}^D \mid \mathcal{S}_{C,t_0}^e(e_{t_1}, d_{t_1}) = true \wedge \mathcal{S}_A^f(f_{t_0}, f_{t_0}^D) = true\}$ (cf. Definition 9). On this basis, the currency specification for features is defined as follows:

$$\mathcal{S}_{C,t_0}^f : (F_{t_1} \times F_{t_1}^D)_C^{t_0} \to \{true, false\}, \mathcal{S}_{C,t_0}^f(f_{t_1}, f_{t_1}^D) = \mathcal{S}_A^f(f_{t_1}, f_{t_1}^D) \tag{V}$$

Finally, to specify currency for data representation feature values, the currency of the underlying data representation features in the sense of Specification (V) is an obviously necessary precondition. Hence, the set of pairs of feature values for which currency is considered is given by $(FV_{t_1} \times FV_{t_1}^D)_C^{t_0} \coloneqq \{f_{t_1}(e_{t_1}), f_{t_1}^D(d_{t_1})) \in FV_{t_1} \times FV_{t_1}^D \mid \mathcal{S}_{C,t_0}^f(f_{t_1}, f_{t_1}^D) = true \wedge \mathcal{S}_A^{fv}\left(f_{t_0}(e_{t_0}), f_{t_0}^D(d_{t_0})\right) = true\}$ (cf. Definition 9). On this basis, the currency specification for feature values is defined as follows:

$$\begin{aligned} &\mathcal{S}_{C,t_0}^{fv} : (FV_{t_1} \times FV_{t_1}^D)_C^{t_0} \to \{true, false\}, \\ &\mathcal{S}_{C,t_0}^{fv}(f_{t_1}(e_{t_1}), f_{t_1}^D(d_{t_1})) = \mathcal{S}_A^{fv}(f_{t_1}(e_{t_1}), f_{t_1}^D(d_{t_1})) \end{aligned} \tag{VI}$$

To illustrate the specifications of currency, we again refer to our running example and consider the feature "Temperature" concerning "Mary Smith" (cf. Figure 2). Since the data "36.5" agrees with the actual feature value at t_0 in the sense of accuracy (cf. Specification (III)) and the currency of the underlying data representation feature is given (cf. Specification (V)), the pair of feature values at t_1 is part of the set $(FV_{t_1} \times FV_{t_1}^D)_C^{t_0}$. Thus, we can consider its currency at t_1. Accuracy of the data at t_1 in the sense of Specification (III) is not given, as the actual temperature changed from "36.5" to "39.2" while the data did not change. Thus, the data is regarded as not current with respect to t_0. While previous literature has focused on this case of changes in the entity state space, our specifications highlight that currency defects may also be caused by

changes in the data state space. For instance, currency also does not hold for the value of the feature “Glucose” of “John Stones”, as the value was (wrongly) changed from “145” to “183” in the database (e.g., caused by a measurement error), while the actual feature value changed from “145” to “140”. In both cases, regardless of the underlying type of event-based state transition (i.e., in the entity state space, data state space, or both), noncurrent data would impede an optimal treatment for the ICU patients.

4 Literature-based Analysis

In a first step, we analyze the proposed specifications for the DQ dimensions accuracy and currency in terms of scope. More precisely, we examine whether the provided specifications indeed cover all DQ defects from literature that are related to accuracy and currency. As illustrated in Figure 1, DQ defects disrupt the representation of entities by data. DQ dimensions characterize this representation and thus, disruptions of the representation related to the respective dimension should be covered by the corresponding specification. Hence, the proposed specifications of accuracy and currency should be able to cover a large variety of DQ defects to exhibit a broad scope of our contribution.

DQ defects have been recognized and discussed intensively in literature [54, 55]. Therefore, we systematically searched the literature for relevant works describing DQ defects in a first step. On this basis, we identified DQ defects related to the DQ dimensions accuracy and currency. Then, we proceeded by evaluating which of these DQ defects are indeed covered by the proposed specifications. In the following, these steps are presented in detail.

4.1 Search for Relevant Papers Describing Data Quality Defects

Without posing a temporal restriction, we searched the databases ACM Digital Library, AIS Library, Google Scholar, IEEE Xplore, ScienceDirect, Springer Link and Wiley for search terms such as (“data defects” AND “data quality”) or (“data problems” AND “data quality”). For search terms where the number of results was exceedingly large, we restricted the search to the first 300 results per database. The resulting 2,711 works were manually screened based on title, abstract and, if necessary, full text, to determine whether or not they describe concrete DQ defects. Overall, we identified 98 relevant works. These works discuss DQ defects not only with respect to the domain of IS, but also in various other fields such as computer science (including sensor technology), business, finance, biomedicine, and medicine. A list of all works (referred to as 1-98 in Table 3) is provided in Appendix A.1.

4.2 Identification of Data Quality Defects Related to Accuracy and Currency

We extracted descriptions of DQ defects from the identified 98 papers, summarizing equivalent descriptions from different papers. This led to a list of 94 descriptions of different DQ defects.

Four researchers independently from each other classified these descriptions as being related to either "accuracy", "currency" or "not relevant". To analyze intercoder reliability we consider Krippendorff's Alpha and Fleiss' Kappa as well-known and commonly used measures which factor out agreements by chance [72]. The results of 0.687 for Krippendorff's Alpha and 0.686 for Fleiss' Kappa are evidence of "substantial" agreement [58]. Disagreements were discussed by the raters to reach consent. In total, 52 relevant DQ defects were identified. These are presented in Table 3 together with the related DQ dimension ("A" for accuracy and "C" for currency).

4.3 Assessment of our Specifications of Accuracy and Currency

We further assessed which of the 52 identified DQ defects from literature are indeed covered by our specifications. An in-depth examination of these DQ defects revealed that it is possible to categorize them into seven Groups i-vii) to allow a group-based, more efficient assessment compared to an assessment of each single DQ defect. These are shown in the column "Groups" in Table 3.

DQ defect [Paper number(s) without the prefix "A" (cf. Appendix A.1)]	**DQ Dim.**	**Groups**
Attributes containing attribute values or vice versa [39, 81]	A	i
Circularity in a self-relationship [62, 69-71]	A	i
Coded wrongly or not conform to real entity [25, 36, 42, 64, 89-90, 92]	A	i
Data entry errors [4-5, 7-8, 16, 23, 34, 38, 41, 46-47, 51, 55-58, 64, 67-68, 75-76, 78, 80, 83, 85, 88, 97]	A	i
Different data for the same entity across multiple databases [20, 38, 41, 47, 55, 57, 67, 75-76, 78, 87-89]	A	i
Distillation errors [41]	A	i
Domain violation / Values outside domain range / Set violation / Interval violation [4-5, 8, 14, 16-17, 26-27, 33-34, 36, 50, 53-55, 57, 60-62, 65-66, 69-71, 73, 79-81, 89-90, 95-96, 98]	A	i
Illegal / Implausible values [25, 34, 36, 39, 53-54, 59, 65, 74, 86, 94, 96]	A	i
Inconsistent duplicate tuples / Inconsistent data / Contradicting records or values [5, 9, 16-20, 23, 26-27, 30, 32, 35-36, 38, 44-45, 47, 50, 52-55, 61-62, 64, 66, 69-71, 74, 77-81, 83, 85-90, 95]	A	i
Inconsistent spatial data* [5, 8, 55, 57, 79]	A	i
Incorrect / Erroneous / Manipulated values / attributes [6, 8, 10-13, 15-20, 22-23, 25-29, 34-35, 37-39, 44-45, 49-50, 52-53, 55-56, 58-60, 62, 64-65, 67-70, 72-74, 79-80, 82, 84-88, 90-91, 93, 96]	A	i
Incorrect (temporal) reference [5, 16, 26-27, 36, 39, 44, 48-50, 59-62, 65, 69-70, 72, 74, 83, 87, 90]	A	i
Inference rule violation / Factual error / Denial constraint violation [45-46, 50, 89, 98]	A	i
Key dependency violation [27, 46, 50-51, 62, 89]	A	i
Measurement errors [2, 11-12, 20, 25, 34-35, 38, 41, 43, 46, 59, 63, 67, 79-80, 86, 89]	A	i

* These DQ defects are not precisely defined by the listed works. Thus, the defects can be caused by inaccurate data (defect related to the DQ dimension accuracy) or missing data (defect related to the DQ dimension completeness). As the DQ dimension completeness is out of scope for this analysis, the focus is on the respective part of the DQ defects related to accuracy.

Misfielded values / Entry into wrong fields [3, 5, 8, 27, 30, 36, 46, 51, 54-55, 60, 62, 71, 74, 79-80, 85, 90, 94, 96-97]	A	i
Misspellings / Grammar errors / Invalid characters / Character alignment violation [1, 4-6, 8, 15-16, 21, 27-28, 30, 36, 51, 53, 55, 60, 62, 65-66, 69-71, 74-80, 83, 90, 96-98]	A	i
Referential integrity violation [4-5, 8, 10, 16, 26-27, 36, 47, 53, 55, 60, 62, 64, 66, 69-71, 73-74, 78-81, 89-90]	A	i
Semantic (temporal) integrity violation [48, 50, 54, 61, 88, 96]	A	i
(Temporal) conditional inclusion dependency violation [48, 50]	A	i
(Temporal) inclusion dependency violation* [45, 48, 50]	A	i
Temporal notion constraint [48]	A	i
(Temporal) transition constraint violation [47-48, 50, 54, 65]	A	i
Unique value violation [1, 5, 8, 16, 26-27, 29, 36, 53, 55, 60, 62, 69-71, 74, 78-81, 85, 89-90, 96]	A	i
Violation of business domain constraint [8, 16, 26, 48, 62, 66, 69-71, 96, 98]	A	i
Violation of (temporal) conditional functional dependency [39, 45, 48, 50, 84, 90, 95]	A	i
Violation of (temporal) functional dependency / attribute dependencies [4, 5, 8, 16, 26-29, 45, 48, 50, 53, 60-62, 65-66, 69-71, 74, 78, 80, 84-85, 96-97]	A	i
Wrong data due to lack of concurrency control / proper crash recovery [4, 38, 55, 80, 86]	A	i
Wrong data type [1, 5-6, 8-9, 17, 29, 34, 36, 51, 53, 55-56, 58, 60-62, 65-66, 69, 71, 74, 76, 79-80, 83, 85, 89-90, 95-98]	A	i
Wrong derived-field / Wrong data among related attributes, information quality contamination [7, 16, 18, 33, 36, 47, 55, 62, 64, 73, 80, 87, 96-97]	A	i
Wrong categorical data [5, 8, 55, 69, 80]	A	i/iii[9]
Sequence constraint violation [48, 72]	A	i/iv[10]
Cardinality constraint violation* [27, 48, 50]	A	i/v[11]
Cardinality ratio violation* [48, 50]	A	i/v[11]
Integrity constraint violation* [66, 94]	A	i/v[11]
Invalid tuples [66]	A	i/v[12]
Meaningless / Undocumented / Empty sense data [17, 19, 27, 32, 62, 71-73, 75-76, 91, 98]	A	i/v[13]
Abbreviations / Truncated values [5, 8, 17, 34, 36, 53, 55, 57, 62, 65-66, 73-74, 79-80, 83, 97]	A	ii
Imprecise / Cryptic data [4-5, 11, 13, 27, 33, 36, 50, 52-53, 62-63, 65-67, 70-71, 73-74, 80, 89]	A	ii
Incomplete context [5, 8, 43, 55, 73, 79, 96]	A	ii

[9] If this DQ defect is caused by inaccurate data (e.g., "out of category range data" [55]) it is categorized into Group i); otherwise (i.e., if it is caused by the misuse of a feature, e.g., "wrong abstraction level" of categorical data [55]), it is categorized into Group iii).

[10] This DQ defect occurs if identity seems to hold for two data representations with respect to the same entity and data representing the feature values of this entity is the same but a given rule claims it should be different or vice versa. If this violation occurs because identity with respect to the entity does not actually hold for one of the data representations, the DQ defect is categorized into Group iv); otherwise (i.e., if the violation of the rule is caused by inaccurate feature values), it is categorized into Group i).

[11] These DQ defects refer to the violation of integrity rules. If the violation is caused by inaccurate data, the DQ defect is categorized into Group i); otherwise (i.e., if the data should not exist at all), it is categorized into Group v).

[12] This DQ defect refers to data that "do not represent valid entities" [70]. If this is the case because there is no entity for which identity holds, the DQ defect is categorized into Group v); otherwise (i.e., if there exists an entity for which identity holds, but the data is not "valid"), it is categorized into Group i).

[13] If this DQ defect occurs with respect to features or feature values, it is categorized into Group i); otherwise (i.e., if it occurs with respect to entities), it is categorized into Group v).

Ambiguous data [7, 16, 23, 26, 53, 57, 60, 62, 69, 79, 90-91, 96]	A	ii/iii/iv[14]
Disjoint subdomains [50]	A	iii
Domain schizophrenia [18]	A	iii
Heterogeneity of aggregation / abstraction / granularity / Different scaling [4, 9, 22-23, 27, 31, 33-34, 36, 48-50, 53, 55-56, 60, 62, 66, 71, 74, 78-79, 90, 95]	A	iii
Heterogeneous reference calendar [48]	A	iii
Heterogeneous representation of intervals [48]	A	iii
Approximate duplicate tuples [5, 9, 16, 19, 26-27, 30, 36, 46, 48, 52, 55, 62, 66, 69-71, 74, 77, 80-81, 89, 96]	A	iv
Heterogeneity in time reference [11, 20, 27, 36, 55, 57, 60, 66-67, 74, 78, 80]	C	vi
Outdated (temporal) data [2-5, 8, 10, 15, 23-24, 27-29, 33, 35-36, 40, 43-46, 52, 55-56, 60-64, 71-72, 75-77, 79, 83, 85, 87, 90, 93, 95, 98]	C	vi
Outdated reference [4-5, 27, 35, 44, 71, 87, 90]	C	vi
Coalesced tuples [48]	C	vii
False state tuple [48]	C	vii

Table 3. Data Quality Defects Discussed in Literature

For illustration purposes, we present two frequently mentioned DQ defects in the context of our running example. First, the DQ defect "Data entry errors" has been mentioned in 27 of the 98 analyzed papers and is related to accuracy. For instance, this DQ defect occurs when a patient's body weight is not captured accurately in an IS because of a typo. Second, the DQ defect "Outdated (temporal) data" has been mentioned in 41 of the 98 analyzed papers and is related to currency. When the clinical diagnosis of a patient changes and this change is not accurately gathered in the database, the stored data on the patient's clinical diagnosis becomes outdated.

An analysis of the 52 DQ defects identified in literature and their categorization into the seven Groups i-vii) shows that they are indeed covered by our specifications of accuracy and currency. Detailed, formal proofs for each group are provided in Appendix A.2. For example, all DQ defects in Group i), which comprises DQ defects where data representations of feature and feature values do not agree with the feature/feature value in the entity state space, are proven to yield *false* in Specifications (II) or (III), respectively, and thus are covered by our specifications.

Linking DQ defects to specifications of DQ dimensions in this way provides a deeper understanding of DQ defects in literature that are partly described in an imprecise or ambiguous way (cf. * in Table 3). The specifications of accuracy and currency enable a systematic analysis of DQ defects discussed in existing work and provide a clear delineation of what constitutes a DQ dimension—that is, which types of DQ defects fall within a given dimension and which do not. Moreover, our specifications allow to identify additional DQ defects which have not yet been considered by existing literature (cf. Section "Discussion and Implications").

[14] If this DQ defect occurs with respect to feature values, it is categorized into Group ii), if it occurs with respect to features, it is categorized into Group iii) and if it occurs with respect to entities, it is categorized into Group iv).

5 Mathematical Analysis

In this section, we provide a mathematical analysis of our theoretical foundations and specifications of DQ dimensions. This mathematical analysis establishes a provable foundation that supports and generalizes the empirical systematization of DQ defects outlined above. It is structured as follows: First, for a given time interval we provide a systematical analysis of possible cases of changes in the entity state space and the digital state space and their respective effects on accuracy over time. Clarifying which of these cases are addressed by currency, we delve into the relation of accuracy and currency. Second, by splitting the considered time interval into smaller parts, we examine these cases in greater detail, providing a combinatorial view on sequences of multiple consecutive changes over time. In particular, we analyze the interplay between accuracy, currency and event-based state transitions.

In order to examine the relation between accuracy and currency, we fix two points in time $t_0 < t_1$ and consider data $f_t^D(d_t)$ representing $f_t(e_t)$ such that $(f_t(e_t), f_t^D(d_t)) \in (FV_t \times FV_t^D)_A$ for each $t \in [t_0, t_1]$. To systematically analyze state changes from t_0 to t_1, we consider all possible cases regarding how the feature values $f_{t_0}(e_{t_0}), f_{t_1}(e_{t_1})$ and data $f_{t_0}^D(d_{t_0}), f_{t_1}^D(d_{t_1})$ could differ. Note that focusing on the level of feature values here is without loss of generality, given that identity holds (cf. Definition 4) for respective pairs of entities or features (the results are analogous).

Table 4 shows a list of all cases of state changes from t_0 to t_1. Each of these cases is given a label from $L1$ to $L14$, which indicates whether there is a change in the entity state space (i.e., $f_{t_0}(e_{t_0}) \neq f_{t_1}(e_{t_1})$) and/or a change in the data state space (i.e., $f_{t_0}^D(d_{t_0}) \neq f_{t_1}^D(d_{t_1})$) as well as whether accuracy is given at t_0 (i.e., $\delta_A^{fv}\left(f_{t_0}(e_{t_0}), f_{t_0}^D(d_{t_0})\right) = true$) and/or t_1 (i.e., $\delta_A^{fv}\left(f_{t_1}(e_{t_1}), f_{t_1}^D(d_{t_1})\right) = true$), respectively. The table covers all 16 cases potentially leading to an accuracy and/or currency defect, as every combination of change in the entity state space, change in the data state space, accuracy at t_0 and accuracy at t_1 is considered. However, the two combinations with no change in the entity state space and no change in the data state space but different accuracy at t_1 and at t_0 cannot occur (the mapping $\delta_A^{fv}: FV_t \times FV_t^D \to \{true, false\}$ yields the same result for the same pair of feature values over time). Thus, the 14 cases listed in Table 4 form a complete basis for the systematic identification of all possible types of accuracy and currency defects.

As a first result, for each of these cases, we are able to derive whether currency of the pair of data $(f_{t_1}(e_{t_1}), f_{t_1}^D(d_{t_1}))$ is defined at all and whether it holds (cf. last column of Table 4).

Label	Change in the entity state space	Change in the data state space	Accuracy at t_0	Accuracy at t_1	Currency at t_1 with respect to t_0
L1			Given	Given	Holds
L2	X	X			
L3	X				
L4		X			
L5	X	X		Not given	Does not hold
L6	X				
L7		X			
L8	X	X	Not given	Given	Not defined
L9	X				
L10		X			
L11				Not given	
L12	X	X			
L13	X				
L14		X			

Table 4. Schematical Consideration of State Changes over Time

As a direct consequence from the definition of $(FV_{t_1} \times FV_{t_1}^{D})_{C}^{t_0}$, currency is defined if and only if accuracy at t_0 is given (cf. cases $L1$ to $L7$). With respect to these cases, currency holds if and only if accuracy at t_1 is given (cf. cases $L1$ to $L4$). On this basis, we can examine the relation between the DQ dimensions currency and accuracy in a rigorous way: One immediate insight illustrated in Table 4, which follows from Specifications (III) and (VI), is that *currency implies accuracy*. What distinguishes the two dimensions is the fact that currency is only defined for cases where accuracy at t_0 is given (cf. Specification (VI) and Table 4). This establishes currency as an extension of accuracy subject to a temporal condition.

Table 4 can also serve as a basis to systematically identify types of DQ defects, in particular with respect to the cases $L5$, $L6$ or $L7$. However, if DQ defects result from (patterns of) multiple state changes over time, their identification requires a more elaborate analysis.

Hence, we proceed with a combinatorial view of state changes over time to analyze situations where one state change succeeds another. We consider three points in time $t_a < t_b < t_c$ in $[t_0, t_1]$ and define the intervals $I_1 := [t_a, t_b]$, $I_2 := [t_b, t_c]$ and the combined interval $I_3 := [t_a, t_c]$. Applying the schema of Table 4 to these three intervals yields the labels l_a, l_b and l_c from $L1$ to $L14$ for the respective cases (cf. Table 4). Further, we write $l_c = l_a * l_b$ to denote that l_c is the composition of l_a and l_b. Knowing only l_a and l_b might not be sufficient to fully determine l_c, we are however able to determine the set of all possible values for l_c based on the combinatorics of changes given in Table 5.

l_b \ l_a	**No Change**	**Change**
No Change	No Change	Change
Change	Change	No Change/Change

Table 5. Combinatorics of Changes in the Event State Space or Data State Space

For example, in a first scenario, a change in the event state space in the interval I_1 is reverted by a second change in the interval I_2 such that there is no change in the event state space for the interval I_3. In a different scenario, the second change in the event state space does not revert the first one, which leads to a change in the event state space in I_3 and thus to a different label l_c. These kinds of considerations yield a partial multivalued mapping, which is shown as a linkage table in Table 6. Here, the entry of the row l_a and the column l_b in the array lists all possible values for $l_c = l_a * l_b$. For example, $L8 * L6$ might be $L12$ or $L14$, which means that a state change with label $L8$ followed by a state change with label $L6$ can yield a state change of type $L12$ or $L14$, depending on the two scenarios described above. Note that half of the compositions are not feasible, because they would imply different values for accuracy at t_b.

l_a \ l_b	**L1**	**L2**	**L3**	**L4**	**L5**	**L6**	**L7**	**L8**	**L9**	**L10**	**L11**	**L12**	**L13**	**L14**
L1	L1	L2	L3	L4	L5	L6	L7	-	-	-	-	-	-	-
L2	L2	L1/ L2/ L3/ L4	L2/ L4	L2/ L3	L5/ L6/ L7	L5/ L7	L5/ L6	-	-	-	-	-	-	-
L3	L3	L2/ L4	L1/ L3	L2	L5/ L7	L6	L5	-	-	-	-	-	-	-
L4	L4	L2/ L3	L2	L1/ L4	L5/ L6	L5	L7	-	-	-	-	-	-	-
L8	L8	L8/ L9/ L10	L8/ L10	L8/ L9	L11/ L12/ L13/ L14	L12/ L14	L12/ L13	-	-	-	-	-	-	-
L9	L9	L8/ L10	L9	L8	L12/ L14	L11/ L13	L12	-	-	-	-	-	-	-
L10	L10	L8/ L9	L8	L10	L12/ L13	L12	L11/ L14	-	-	-	-	-	-	-
L5	-	-	-	-	-	-	-	L1/ L2/ L3/ L4	L2/ L4	L2/ L3	L5	L5/ L6/ L7	L5/ L7	L5/ L6
L6	-	-	-	-	-	-	-	L2/ L4	L1/ L3	L2	L6	L5/ L7	L6	L5
L7	-	-	-	-	-	-	-	L2/ L3	L2	L1/ L4	L7	L5/ L6	L5	L7
L11	-	-	-	-	-	-	-	L8	L9	L10	L11	L12	L13	L14
L12	-	-	-	-	-	-	-	L8/ L9/ L10	L8/ L10	L8/ L9	L12	L11/ L12/ L13/ L14	L12/ L14	L12/ L13
L13	-	-	-	-	-	-	-	L8/ L10	L9	L8	L13	L12/ L14	L11/ L13	L12
L14	-	-	-	-	-	-	-	L8/ L9	L8	L10	L14	L12/ L13	L12	L11/ L14

Table 6. Composition of State Changes

Assessing these time intervals also enables to apply the analysis to event-based state transitions (cf. Definition 6). We can apply the mapping (cf. Table 6) iteratively for more than two intervals whereby the composition is associative. In particular, the composition of state changes yields a recipe of how to affiliate a state change between t_0 and t_1 to the event-based state transitions at points in time between t_0 and t_1. More precisely, for every state change l between t_0 and t_1, the event-based state transitions between t_0 and t_1 correspond to state changes with labels $l_a, \dots, l_{\mathcal{H}}$

such that $l = l_a * ... * l_{\mathcal{H}}$. This higher level of granularity allows for a careful identification of causes of DQ defects which involve certain patterns of multiple state changes over time.

The above considerations enable a precise description regarding the interplay of accuracy, currency and event-based state transitions: Let $l \in \{L5, L6, L7\}$ be the label of a state change from t_0 to t_1 and consider the sequence of event-based state transitions with corresponding labels $l_a, \dots, l_{\mathcal{H}}$ such that $l = l_a * ... * l_{\mathcal{H}}$. We call the last index value I (in the ordered index sequence $(a, b, c, \dots, I, \dots, \mathcal{H})$) such that $l_I \in \{L5, L6, L7\}$ the *actuator* of l. In other words, the actuator affiliates a state change from t_0 to t_1 where currency does not hold at t_1 to the event-based state-transition at the last point in time $t \in [t_0 + h, t_1]$, where the accuracy at $t - h$ was given, but the accuracy at t is not. Table 6 can be used to verify that the actuator is well-defined: If $L5$, $L6$ or $L7$ is written as a product of two labels, one of these labels has to be an element of $\{L5, L6, L7\}$ as well. Applying this observation iteratively yields the same result for the sequence $l_a * ... * l_{\mathcal{H}}$.

Note that the label of the actuator does not have to agree with l. For example, Table 6 shows that there could exist a state transition of type $L2$ followed by a state transition of type $L7$ such that the resulting state change is of type $L6$. In this case, we would get $l = L6$, but the actuator would be $l_b = L7$. For example, this case occurs when the temporary transfer of a patient to another ward for an examination (affecting the entity state space) is recorded accurately in the data at first (this yields label $L2$, cf. Table 4), but the ward transfer is reverted in the data before the examination and the stay in the other hospital ward actually end (this yields label $L7$, cf. Table 4). In combination, there is a change in the entity state space but no change in the data state space (this yields label $L6$, cf. Table 4). This example highlights how our mathematical analysis enables the formal systematization of DQ defects, expanding upon the empirical systematization provided by our literature-based analysis.

6 Discussion and Implications

Theoretical foundations for DQ are crucial to cope with DQ dimensions and DQ defects in research and practice. The coherent foundations introduced in this paper to specify the DQ dimensions accuracy and currency allow for a rigorous analysis and discussion of the DQ dimensions themselves, their relation and associated DQ defects. As a result, a systematic understanding of DQ dimensions and DQ defects is provided. In the following, we will elaborate on four aspects in more detail:

First, our theoretical foundations serve as a valuable basis for both the discussion of DQ in research and the precise specification of DQ dimensions.

Our theoretical foundations for DQ are based on recognized literature and provide formal and coherent definitions. They substantially extend existing foundations in three important aspects: (a) The theoretical foundations expose in depth what is meant by both, the intrinsic perspective on DQ grounding on facts and the representation of entities by data characterized by different DQ dimensions; (b) by aiming attention at data representations, they take a data-oriented view being the core of DQ; (c) they include a detailed structure of the considered data (referring to

entities, features and feature values), allowing a detailed specification of DQ dimensions. Thus, the theoretical foundations not only enable the proposed specifications of accuracy and currency but also provide a valuable grounding for further DQ dimensions such as consistency. For instance, if (semantic) consistency is defined as the absence of contradictions between considered data based on a rule set [43], the relation between consistency and accuracy can be analyzed formally (e.g., is consistent data always accurate and vice versa).

Second, we establish precise specifications of the DQ dimensions accuracy and currency also covering the levels of entities and features.

Literature provides many definitions for accuracy. For instance, Wang and Strong [92] define accuracy as "the extent to which data are correct, reliable and certified"; Ballou and Pazer [9] state that data is accurate when the data values stored in a database correspond to real-world values; further, Sicari et al. [88] define accuracy as the degree of the similarity of a measured value to some other value that is considered true. At their core, many existing definitions of accuracy focus on a juxtaposition of a data value and an ideal value that usually is seen as its real-world counterpart. However, it is not precisely specified when this juxtaposition is regarded as an agreement (i.e., under which formal condition the data value is regarded as accurate; cf. also [40]). Yet, a rigorous understanding of this agreement and when it is broken by DQ defects is a crucial foundation for the scientific discourse about accuracy and other DQ dimensions. In addition, existing specifications of accuracy do not reveal the concept of identity. Thus, DQ defects from literature which stem from a lack of identity (e.g., DQ defect "Meaningless / Undocumented / Empty Sense Data" in Table 3) are not covered by existing specifications. Moreover, our theoretical foundations comprise structural characteristics regarding entities, features, and feature values. This allows to precisely specify accuracy not only with respect to feature values (as common in existing definitions), but also with respect to entities and features. This way, all DQ defects from literature related to accuracy are covered (cf. Groups i-v) in the Section "Literature-Based Analysis").

While definitions of accuracy in literature coincide to some extent, many definitions of time-related dimensions agree neither in terms of terminology nor in terms of notion. As Batini et al. [13] recognize, major time-related dimensions are referred to as currency, timeliness and volatility [22, 49]. For instance, Jarke et al. [48] and Bovee et al. [16] define currency as the age of data (i.e., the time since its last update) and volatility as the frequency of changes to the represented feature values. Indeed, information about the time since the last update of data and the frequency of changes to feature values can support the measurement of currency but should not be part of a definition / specification of currency (i.e., what is meant by currency and not how it is measured). The definitions by Wang and Strong [92], Liu and Chi [62], and Sicari et al. [88] are also based on the age of data, and thus, the same argument applies. Moreover, the definition of timeliness by Wand and Wang [90] refers to the delay between the change of feature values and the resulting modification of the corresponding data. However, being able to specify currency is all the more important if there is no such resulting modification. The specifications of currency proposed in this paper avoid such issues and are generally applicable, because they do

not depend on the existence of any change to the data state space or the entity state space. Moreover, they do not only cover currency of feature values but also define currency of features and entities.

Besides, our work also supports researchers by enabling to compare and potentially harmonize existing definitions in the literature. Indeed, based on our foundations and specifications, existing (groups of) DQ definitions can be compared and it can be assessed whether they cover different DQ defects and state changes, and which context they address. To demonstrate this, a first example analysis in Table 7 shows how our work allows to scrutinize existing definitions of currency.

Data Quality Defect	Context	Group of Ballou et al. [10], Pernici and Scannapieco [79], Pipino et al. [81]	Group of Heinrich and Klier [41], Zak and Even [97]
Heterogeneity in time reference	Feature values with fixed shelf life	**X**	**X**
	Feature values without fixed shelf life	--	**X**
Outdated (temporal) data	Features	--	--
	Unchanged feature values … with fixed shelf life	**X**	**X**
	Unchanged feature values … without fixed shelf life	--	**X**
	Updated feature values	--	--
Outdated reference	Feature values with fixed shelf life	**X**	**X**
	Feature values without fixed shelf life	--	**X**
Coalesced tuples		--	--
False state tuple		--	--
Entity change not reflected		--	--
Inaccurate update		--	--

Table 7. Exemplary Analysis of Existing Currency Definitions

Our analysis shows that in their seminal work, Ballou et al. [10] cover the currency defects "Heterogeneity in time reference", "Outdated (temporal) data" and "Outdated reference" when defining currency (in their terms, timeliness). More precisely, they focus on the currency of feature values for cases where the (maximum) timespan for which the feature value remains accurate ("shelf life") is fixed and the stored data remains unchanged. Similar definitions are proposed by, for instance, Pipino et al. [81] and Pernici and Scannapieco [79]. Further, our analysis yields that the definition of Heinrich and Klier [41] covers the same three currency defects, but in additional contexts (feature values without fixed shelf life). Their view is taken up by, for instance, Zak and Even [2017]. Further currency defects such as "Coalesced tuples" and "Entity change not reflected" are not covered by these definitions.

Moreover, even though our specifications of DQ dimensions are clearly separated from the assessment of DQ, our work supports the evaluation of DQ metrics. Given the analysis of definitions in Table 7, existing currency metrics based on these definitions can be scrutinized revealing

the opportunity for benchmarking (groups of) DQ metrics. For instance, the metrics by Heinrich and Klier [2015] and Zak and Even [97] can be applied in the same contexts regarding Table 7. This allows a comparative evaluation based on public datasets comprising different currency defects (e.g., the National Long-Term Care Survey datasets [68] or the MIMIC dataset [53]) to reveal their performance (e.g., true positives, false positives etc.) for each of these currency defects. Thus, benchmarking of DQ definitions and metrics and replicability of results using public datasets is enabled, which is important to document the progress of DQ research in IS. Further and from a practical perspective, the assessment and selection of DQ metrics for concrete applications is supported, since the metric that is most suitable for the application context at hand can be chosen.

Finally, based on our mathematical analysis and the discussed state changes, it can be shown that all currency metrics based on definitions presented in Table 7 focus on case $L6$ (changes in the entity state space) only, thus disregarding the cases $L5$ (changes in the entity state space and the data state space) and $L7$ (changes in the data state space), which leads to currency defects not covered by these metrics. In fact, extending the analysis presented in Table 7 to further currency metrics exposes that there are several currency defects which are not covered by any existing metric (e.g., “False state tuple”, “Inaccurate update”). Thus, a further result of our work is the need for the future design of novel currency metrics or the extension of existing metrics which provide sound assessment of DQ and decision support in contexts where these currency defects occur.

Third, the relation between accuracy and currency including the interplay between accuracy, currency, and event-based state transitions is rigorously examined.

The relation between accuracy and currency has already been discussed in literature. For instance, Scannapieco et al. [85] mention that “noncurrent data are often perceived as not accurate” and Zak and Even [96] point out that “as data becomes less current it is also likely to become less accurate”. Other works indicate a similar relation [e.g., 89]. Our formal specifications enable a well-founded examination (cf. Section “Mathematical Analysis”) yielding clear results: *Currency is only defined for cases where accuracy at a former point in time is given* and *currency implies accuracy* (i.e., if currency holds at a point in time, so does accuracy). Regarding the interplay between accuracy, currency and event-based state transitions, our examination further reveals that *each currency defect is caused by an event-based state transition (the actuator) resulting in inaccurate data*. This event-based state transition can be associated to a change in the entity state space (case $L6$), a change in the data state space (case $L7$), or a combination of both (case $L5$) which is not obvious from the currency defect itself. This result sheds light upon the interplay between accuracy, currency, and event-based state transitions. Overall, our insights constitute a first but important step to disentangle the relation between accuracy and currency and contribute to an improved understanding of both DQ dimensions.

As an example of how event-based state transitions can serve as a basis to address currency defects, we analyzed the mortality prediction task outlined above. We built on the data and ML

pipeline proposed by Harutyunyan et al. [38] and published in the respective benchmark repository.[15] Our main idea was to use our specifications to model drug administration as a DQ-related event that induces state transitions (cf. Definition 7) by changing the values of the clinical features affected by the respective drug. Note that, with regard to the dataset, this event is only one of many DQ-related events but it is one we analyzed systematically and in depth on the basis of the proposed specifications and state transitions. To incorporate the impact of this DQ-related event, we additionally extracted data son drug administrations from the raw data of the MIMIC dataset [53]. As examples, we selected the drugs Insulin, Acetaminophen, and Hydralazine because of their frequent administration and their clear effect on glucose, temperature, and diastolic blood pressure, respectively, but other drugs could be used analogously. Based on the training data (as provided by the benchmark) and the drug administration event data, we used automated rule extraction to model event-based state transitions (details provided in Appendix A.3). For the challenging and high-stakes task of mortality prediction, this way of modeling only *one* DQ-related event resulted in a recall of 41.8% for patients receiving at least one of these drugs. In contrast, not considering event-based state transitions for addressing currency defects (i.e., just "filling forward" old data) led to a recall of just 36.4% for these patients, meaning that proper addressing of currency defects based on our theoretical foundations improved recall by 5.4 percentage points (~15% relative increase). At the same time, precision rose to 69.7% (from 62.5%), F1-score increased to 0.523 (from 0.460), and overall accuracy improved to 89.9% (from 88.7%).

Thus, in practical use as well, our examination of state changes in Tables 4, 5, and 6 may serve as a promising basis for analyzing patterns of DQ defects. For instance, if a large amount of data in an IS is found to not hold currency, it can be analyzed which patterns referring to the cases $L5$, $L6$, and $L7$ mainly occur, and what the actuator is. This supports to pinpoint causes of currency defects and to enact goal-oriented DQ improvement measures. In case the analyzed actuator is of type $L6$ (i.e., the entity state space changes, but data is not updated accordingly in the database), for instance, customers can be motivated and incentivized to check and update their data when they login to the customer portal. In case the actuator is of type $L5$ (i.e., changes in the entity state space are updated in the database but not accurately, for instance, because of typos), processes to capture and gather data have to be examined. In case the actuator is of type $L7$ (e.g., the primary cause of currency defects is database integration), the related procedures such as schema matching may be revised. Indeed, currency defects are not caused in a univariate way. But the presented systematic of state changes and actuators can support organizations to identify their major causes for poor accuracy and currency. Further along this thought, patterns of state changes and actuators can also be used in a forward-looking way. For instance, such a pattern may reveal that in a customer database the feature value "teacher" of the feature "working status" regularly becomes noncurrent at the age of 65 in a certain country, as these teachers retire. Based on such a pattern, automatically generated proposals to update data can be generated supporting to ease poor DQ in databases.

[15] https://github.com/YerevaNN/mimic3-benchmarks

Literature Gap	Novel Group of DQ Defects (Description)	Example (Healthcare Database)	Example (Customer Database)
Existing literature does not cover currency defects on the entity level	Changes due to event-based state transitions which affect an entity and/or its data representation result in inaccurate data. This group of DQ defects on the entity level covers not reflected entity changes (missing updates), inaccurate record data integration, false updates over time and so on.	Data representing the entity "Mary Smith" has been accurate at t_0, as the entity "Mary Smith" is an inpatient of the hospital and the data representing "Mary Smith" is contained in the relation "inpatients" of the hospital's database. In case Mary Smith leaves the hospital later on while the respective data representation "Mary Smith" is not removed from the relation "inpatients", currency at the entity level does not hold anymore at a later point in time t_1. For the hospital, this could result in Mary Smith being wrongly scheduled for medical procedures in the hospital – whereas she is also wrongly not considered for crucial outpatient follow-up care.	Data representing the entity "Anna Williams" has been accurate at t_0, as the entity "Anna Williams" is a prospective customer (lead) of the company and the data representing "Anna Williams" is contained in the relation "leads" of the company's database. In case Anna Williams buys her first products later on while the respective data representation "Anna Williams" is not moved from the relation "leads" to "customers", currency at the entity level does not hold anymore at a later point in time t_1. For the company, this could result in Anna Williams being wrongly selected for specific acquisition campaigns for leads instead of intensifying her customer relationship.
Existing literature does not cover currency defects involving changes of both entity state space and data state space	After changes of both the entity state space and the data state space, the data representing an entity/feature/feature value is not accurate anymore. This group of DQ defects covers inaccurate updates such as cascades of feature value updates, inaccurately updated feature values due to data integration, (deliberate) manipulation of the data over time and so on.	The data "14" for the feature "Respiratory Rate" has been accurate for the entity "John Stones" at t_0. In case John Stones has trouble breathing later on and his respiratory rate drops to 7 while the respective data is (erroneously) changed to 17, currency at the feature value level does not hold anymore at a later point in time t_1. The need to initiate countermeasures such as mechanical ventilation would thus go unnoticed.	The data "Kansas City" for the feature "Place of Residence" has been accurate for the entity "Anna Williams" at t_0, as Anna Williams in fact lives in Kansas City. In case Anna Williams relocates to Seattle later on while the respective data is (erroneously) changed to "Syracuse", currency at the feature value level does not hold anymore at a later point in time t_1. Product campaigns would therefore no longer reach Anna Williams and other customers showing such an DQ defect.

Table 8. Groups of DQ Defects not yet Discussed in Literature

Fourth, our specifications do not only cover all accuracy and currency defects described in extant works but allow to identify further relevant groups of DQ defects not yet discussed in literature.

As shown in the Section "Literature-based Analysis", our specifications cover all related DQ defects from existing literature, thus substantiating these specifications. Further, based on the specifications and our analysis, we identify two additional groups of DQ defects, which have neither been identified nor discussed in literature yet. Table 8 illustrates the novel DQ defects in the context of two different domains.

First, the currency of data representing entities can change over time as a result of event-based state transitions. Specification (IV) covers such currency defects on the entity level. Second, there are also relevant currency defects where changes with respect to both entity state space and data state space occur (cf. also case $L5$ in Table 4). For instance, this group of currency defects exists if a change occurs in the entity state space and is recorded inaccurately in the data state space. While existing definitions and specifications of currency focus on data quality defects tracing back to changes in exactly one space—entity state space or data state space—but staying the same in the other (cf. Groups vi) and vii)), our specifications cover such defects.

Besides novel groups of DQ defects, the "Literature-based Analysis" does not only yield a list of DQ defects in literature, but also provides insights regarding their prevalence based on the number of works in which they are discussed. In practice, these results can support the analysis of (poor) DQ in concrete applications, as they offer hints and recommendations of major DQ defects to consider, such as "Approximate duplicate tuples" and "Heterogeneity in time reference" (cf. Table 3). Further along this thought, the provided list and groups of DQ defects can also be used to support the assessment of DQ tools in practice. A first analysis shows that the two DQ tools Apache Griffin and DataCleaner address partly different DQ defects and contexts. For instance, DataCleaner explicitly addresses DQ defects such as "Abbreviations/Truncated values" based on a synonym dictionary, whereas Apache Griffin can be used to address "Incorrect/Erroneous values" based on a suitable reference dataset. This allows to extend existing assessments of DQ tools functionalities [4] and supports tool selection in practice depending on which DQ defects occur in a concrete application.

Our contributions and implications for research and practice are summarized in Figure 3.

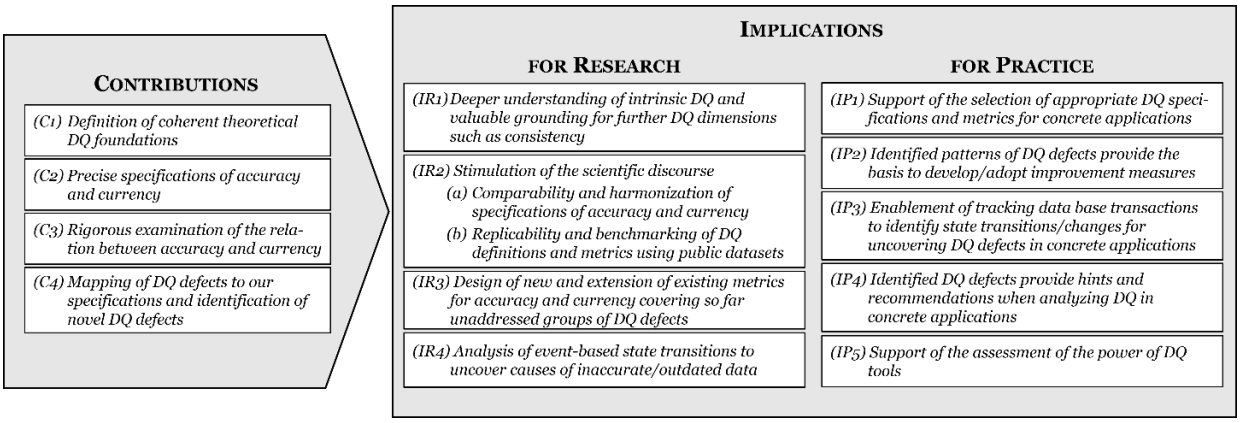


Figure 3. Contributions and Implications for Research and Practice

7 Conclusion, Limitations and Further Research

In this paper, we introduce theoretical foundations for intrinsic DQ, provide specifications of the DQ dimensions accuracy and currency, and conduct a literature-based as well as a mathematical analysis. Our contributions are as follows: First, our theoretical foundations can serve as a valuable basis for both the discussion of DQ and the specification of DQ dimensions in research and practice. Second, we establish precise specifications also covering the levels of entities and

features. Third, the proposed theoretical foundations and specifications allow for a rigorous examination of the relation between accuracy and currency. Fourth, our specifications do not only cover all accuracy and currency defects described in extant works but allow the identification of further relevant groups of DQ defects not yet discussed in literature.

Besides these contributions, there are also limitations that may constitute starting points for future research. First, we focused on intrinsic DQ, which means that our theoretical foundations rely on an objective perspective on DQ grounding on facts. Future works should extend these theoretical foundations to an extrinsic view including, for instance, context as well as user interpretations. In addition, we concentrated on the DQ dimensions accuracy and currency. Based on our theoretical foundations, future research should precisely specify further important DQ dimensions such as completeness and consistency, allowing for a broader discussion. Moreover, our foundations focus on structured data. However, DQ of unstructured data is also of major interest, e.g. in AI-driven decision-making. For example, the specification and assessment of currency of textual data yields challenging problems. Future works may thus extend our theoretical foundations to unstructured data to foster well-founded research in this area. Finally, another important direction for future research is the development of DQ metrics based on the presented theoretical foundations and specifications. In particular, our work revealed the urgent need for novel or extended currency metrics, since existing metrics mainly focus on feature values and are only applicable in certain contexts. Despite these limitations and directions for future research, we hope that our work will open doors for further discussions in this exciting area.

8 References

[1] Ahmed Abbasi, Suprateek Sarker, and Roger H. Chiang. 2016. Big data research in information systems: Toward an inclusive research agenda. *J. Assoc. Inform. Systems* 17, 2 (2016), 1–32.

[2] Daniel Aebi and Louis Perrochon. 1993. Towards improving data quality. In *Proceedings of the International Conference on Information Systems and Management of Data*, 273–281.

[3] Shahriar Akter and Samuel F. Wamba. 2016. Big data analytics in e-commerce: A systematic review and agenda for future research. *Electronic Markets* 26, 2 (2016), 173–194.

[4] Marcel Altendeitering and Martin Tomczyk. 2022. A functional taxonomy of data quality tools: Insights from science and practice. In *Proceedings of the 17th International Conference on Wirtschaftsinformatik*.

[5] Ofer Arazy, Rick Kopak, and Irit Hadar. 2017. Heuristic principles and differential judgments in the assessment of information quality. *J. Assoc. Inform. Systems* 18, 5 (2017), 403–432.

[6] Ofer Arazy, Oded Nov, Raymond Patterson, and Lisa Yeo. 2011. Information quality in Wikipedia: The effects of group composition and task conflict. *J. Management Inform. Systems* 27, 4 (2011), 71–98.

[7] Bart Baesens, Ravi Bapna, James R. Marsden, Jan Vanthienen, and J. L. Zhao. 2016. Transformational issues of big data and analytics in networked business. *MIS Quart.* 40, 4 (2016), 807–818.

[8] Xue Bai, Manuel Nunez, and Jayant R. Kalagnanam. 2012. Managing data quality risk in accounting information systems. *Inform. Systems Res.* 23, 2 (2012), 453–473.

[9] Donald P. Ballou and Harold L. Pazer. 1995. Designing information systems to optimize the accuracy-timeliness tradeoff. *Inform. Systems Res.* 6, 1 (1995), 51–72. https://doi.org/10.1287/isre.6.1.51.

[10] Donald P. Ballou, Richard Y. Wang, Harold L. Pazer, and Giri K. Tayi. 1998. Modeling information manufacturing systems to determine information product quality. *Management Sci.* 44, 4 (1998), 462–484.

[11] Brian Ballsun-Stanton. 2010. Asking about data: Experimental philosophy of information technology. In *Proceedings of the 5th International Conference on Computer Sciences and Convergence Information Technology*, 119–124.

[12] Brian Ballsun-Stanton. 2012. *Asking about data: Exploring different realities of data via the social data flow network methodology*. University of New South Wales.

[13] Carlo Batini, Cinzia Cappiello, Chiara Francalanci, and Andrea Maurino. 2009. Methodologies for data quality assessment and improvement. *ACM Comput. Surv.* 41, 3 (2009), 1–52. https://doi.org/10.1145/1541880.1541883.

[14] Carlo Batini and Monica Scannapieco. 2016. *Data and information quality*. Springer.

[15] Donald J. Berndt, James A. McCart, Dezon K. Finch, and Stephen L. Luther. 2015. A case study of data quality in text mining clinical progress notes. *ACM Trans. Manage. Inf. Syst.* 6, 1 (2015). https://doi.org/10.1145/2669368.

[16] Matthew Bovee, Rajendra P. Srivastava, and Brenda Mak. 2003. A conceptual framework and belief-function approach to assessing overall information quality. *Int. J. Intell. Systems* 18, 1 (2003), 51–74. https://doi.org/10.1002/int.10074.

[17] Gibson Burrell and Gareth Morgan. 1979. *Sociological paradigms and organisational analysis: Elements of the sociology of corporate life*. Routledge.

[18] Andrew Burton-Jones, Jan Recker, Marta Indulska, Peter Green, and Ron Weber. 2017. Assessing representation theory with a framework for pursuing success and failure. *MIS Quart.* 41, 4 (2017), 1307–1333. https://doi.org/10.25300/MISQ/2017/41.4.13.

[19] Li Cai and Yangyong Zhu. 2015. The challenges of data quality and data quality assessment in the big data era. *Data Sci. J.* 14 (2015), 2.

[20] Cinzia Cappiello, Chiara Francalanci, and Barbara Pernici. 2003. Time-related factors of data quality in multichannel information systems. *J. Management Inform. Systems* 20, 3 (2003), 71–92.

[21] Marco Casanova, Karin Breitman, Daniela Brauner, and André Marins. 2007. Database conceptual schema matching. *Computer* 40, 10 (2007), 102–104. https://doi.org/10.1109/MC.2007.342.

[22] Oleksiy Chayka, Themis Palpanas, and Paolo Bouquet. 2012. *Defining and measuring data-driven quality dimension of staleness*. University of Trento.

[23] Rongli Chen, Xiaozhong Chen, Lei Wang, and Jianxin Li. 2023. The core industry manufacturing process of electronics assembly based on smart manufacturing. *ACM Trans. Manage. Inf. Syst.* 13, 4 (2023). https://doi.org/10.1145/3529098.

[24] Corinna Cichy and Stefan Rass. 2019. An overview of data quality frameworks. *IEEE Access* 7 (2019), 24634–24648.

[25] Alexander Clark. 1998. The qualitative-quantitative debate: Moving from positivism and confrontation to post-positivism and reconciliation. *J. Advanced Nursing* 27, 6 (1998), 1242–1249. https://doi.org/10.1046/j.1365-2648.1998.00651.x.

[26] James L. Cross, Michael A. Choma, and John A. Onofrey. 2024. Bias in medical AI: Implications for clinical decision-making. *PLOS Digit. Health* 3, 11 (2024), e0000651. https://doi.org/10.1371/journal.pdig.0000651.

[27] Jérôme Euzenat and Pavel Shvaiko. 2007. *Ontology matching*. Springer.

[28] Xiao Fang, Olivia R. L. Sheng, and Paulo Goes. 2013. When is the right time to refresh knowledge discovered from data? *Oper. Res.* 61, 1 (2013), 32–44. https://doi.org/10.1287/opre.1120.1148.

[29] Kit Fine. 1982. First-order modal theories III: Facts. *Synthese* 53, 1 (1982), 43–120.

[30] Andreas Fügener, Dominik D. Walzner, and Alok Gupta. 2026. Roles of artificial intelligence in collaboration with humans: Automation, augmentation, and the future of work. *Management Sci.* 72, 1 (2026), 538–557. https://doi.org/10.1287/mnsc.2024.05684.

[31] Maryam Ghasemaghaei and Goran Calic. 2019. Can big data improve firm decision quality? The role of data quality and data diagnosticity. *Decision Support Systems* 120 (2019), 38–49. https://doi.org/10.1016/j.dss.2019.03.008.

[32] Maryam Ghasemaghaei and Ofir Turel. 2021. Possible negative effects of big data on decision quality in firms: The role of knowledge hiding behaviours. *Inform. Systems J.* 31, 2 (2021), 268–293. https://doi.org/10.1111/isj.12310.

[33] Fausto Giunchiglia and Pavel Shvaiko. 2003. Schema matching. *Knowledge Engrg. Rev.* 18, 3 (2003), 265–280. https://doi.org/10.1017/S0269888904000074.

[34] Shirley Gregor. 2006. The nature of theory in information systems. *MIS Quart.* 30, 3 (2006), 611–642. https://doi.org/10.2307/25148742.

[35] Tianjian Guo, Indranil R. Bardhan, Ying Ding, and Shichang Zhang. 2025. An explainable artificial intelligence approach using graph learning to predict intensive care unit length of stay. *Inform. Systems Res.* 36, 3 (2025), 1478–1501. https://doi.org/10.1287/isre.2023.0029.

[36] Nitin Gupta, Shashank Mujumdar, Hima Patel, Satoshi Masuda, Naveen Panwar, Sambaran Bandyopadhyay, Sameep Mehta, Shanmukha Guttula, Shazia Afzal, Ruhi S. Mittal, and Vitobha Munigala. 2021. Data quality for machine learning tasks. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 4040–4041.

[37] Michael Hagn, Bernd Heinrich, Thomas Krapf, and Alexander Schiller. 2025. Handling imperfection: A taxonomy for machine learning on data with data quality defects. *Decision Support Systems* 196 (2025), 114493. https://doi.org/10.1016/j.dss.2025.114493.

[38] Hrayr Harutyunyan, Hrant Khachatrian, David C. Kale, Greg Ver Steeg, and Aram Galstyan. 2019. Multitask learning and benchmarking with clinical time series data. *Nature Scientific Data* 6, 1 (2019), 96. https://doi.org/10.1038/s41597-019-0103-9.

[39] M. R. Hasan, Bastian Finkel, and Christine Legner. 2026. The data product canvas: Designing data products for sustained value from enterprise data. *Inform. Systems J.* 36, 1 (2026), 144–158. https://doi.org/10.1111/isj.12603.

[40] Anders Haug. 2021. Understanding the differences across data quality classifications: A literature review and guidelines for future research. *Indust. Management and Data Systems* 121, 12 (2021), 2651–2671. https://doi.org/10.1108/IMDS-12-2020-0756.

[41] Bernd Heinrich and Mathias Klier. 2015. Metric-based data quality assessment — Developing and evaluating a probability-based currency metric. *Decision Support Systems* 72 (2015), 82–96. https://doi.org/10.1016/j.dss.2015.02.009.

[42] Bernd Heinrich, Mathias Klier, Andreas Obermeier, and Alexander Schiller. 2025. Different but the Same? An Event-driven Approach to Determine Probabilities of Data Duplication. *MIS Quart.* (2025), 1–34. https://doi.org/10.25300/MISQ/2025/18178.

[43] Bernd Heinrich, Mathias Klier, Alexander Schiller, and Gerit Wagner. 2018. Assessing data quality – A probability-based metric for semantic consistency. *Decision Support Systems* 110 (2018), 95–106.

[44] Joyce C. Ho, Cheng H. Lee, and Joydeep Ghosh. 2014. Septic shock prediction for patients with missing data. *ACM Trans. Manage. Inf. Syst.* 5, 1 (2014). https://doi.org/10.1145/2591676.

[45] Gerald J. Holton. 1993. *Science and anti-science*. Harvard University Press.

[46] Informatica. 2025. *The surprising reason most AI projects fail – And how to avoid it at your enterprise* (2025). Retrieved November 19th, 2025 from https://www.informatica.com/blogs/the-surprising-reason-most-ai-projects-fail-and-how-to-avoid-it-at-your-enterprise.html.

[47] Mark Jansen, Hieu Q. Nguyen, and Amin Shams. 2025. Rise of the machines: The impact of automated underwriting. *Management Sci.* 71, 2 (2025), 955–975. https://doi.org/10.1287/mnsc.2024.4986.

[48] Matthias Jarke, Maurizio Lenzerini, Yannis Vassiliou, and Panos Vassiliadis. 2003. *Fundamentals of data warehouses*. Springer.

[49] Vimukthi Jayawardene, Shazia Sadiq, and Marta Indulska. 2013. *An analysis of data quality dimensions*. The University of Queensland.

[50] Vimukthi Jayawardene, Shazia Sadiq, and Marta Indulska. 2013. The curse of dimensionality in data quality. In *Proceedings of the 24th Australasian Conference on Information Systems*.

[51] Lei Jiang, Alex Borgida, and John Mylopoulos. 2008. Towards a compositional semantic account of data quality attributes. In *Proceedings of the 27th International Conference on Conceptual Modeling*, 55–68.

[52] Yiqun Jiang, Shaodong Wang, Qing Li, and Wenli Zhang. 2025. ICU outcome predictions using real-time signals with wavelet-transform-based deep learning method. *ACM Trans. Manage. Inf. Syst.* 16, 4 (2025). https://doi.org/10.1145/3727624.

[53] Alistair Johnson, Tom Pollard, and Roger Mark. 2016. *MIMIC-III clinical database (version 1.4)*, PhysioNet.

[54] Joao Josko. 2018. A formal taxonomy of temporal data defects. In *International Workshop on Data Quality and Trust in Big Data*.

[55] Won Kim, Byoung-Ju Choi, Eui-Kyeong Hong, Soo-Kyung Kim, and Doheon Lee. 2003. A taxonomy of dirty data. *Data Mining and Knowledge Discovery* 7, 1 (2003), 81–99. https://doi.org/10.1023/A:1021564703268.

[56] Henry Kon, Jacob Lee, and Richard Y. Wang. 1993. *A process view of data quality*. MIT TDQM Research Program.

[57] Ramayya Krishnan, James Peters, Rema Padman, and David Kaplan. 2005. On data reliability assessment in accounting information systems. *Inform. Systems Res.* 16, 3 (2005), 307–326.

[58] J. R. Landis and Gary G. Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics* (1977), 159–174. https://doi.org/10.2307/2529310.

[59] Yang W. Lee. 2003. Crafting rules: Context-reflective data quality problem solving. *J. Management Inform. Systems* 20, 3 (2003), 93–119.

[60] Yang W. Lee, Diane M. Strong, Beverly K. Kahn, and Richard Y. Wang. 2002. AIMQ: A methodology for information quality assessment. In *Proceedings of the 7th International Conference on Information Quality*, 133–146.

[61] Hong Liu. 2014. Philosophical reflections on data. *Procedia Computer Science* 30 (2014), 60–65. https://doi.org/10.1016/j.procs.2014.05.381.

[62] Liping Liu and Lauren Chi. 2002. Evolutional data quality: A theory-specific view. In *Proceedings of the 7th International Conference on Information Quality*, 292–304.

[63] Qi Liu, Gengzhong Feng, Giri K. Tayi, and Jun Tian. 2021. Managing data quality of the data warehouse: A chance-constrained programming approach. *Inform. Systems Frontiers* 23, 2 (2021), 375–389. https://doi.org/10.1007/s10796-019-09963-5.

[64] Qi Liu, Gengzhong Feng, Nengmin Wang, and Giri K. Tayi. 2018. A multi-objective model for discovering high quality knowledge based on data quality and prior knowledge. *Inform. Systems Frontiers* 20, 2 (2018), 401–416.

[65] Qi Liu, Gengzhong Feng, Weibo Zheng, and Jun Tian. 2022. Managing data quality of cooperative information systems: Model and algorithm. *Expert Systems with Appl.* 189 (2022), 116074.

[66] Roman Lukyanenko, Jeffrey Parsons, and Yolanda Wiersma. 2014. The IQ of the crowd: Understanding and improving information quality in structured user-generated content. *Inform. Systems Res.* 25, 4 (2014), 669–689.

[67] Roman Lukyanenko, Jeffrey Parsons, Yolanda Wiersma, and Madeh Maddah. 2019. Expecting the unexpected: Effects of data collection design choices on the quality of crowdsourced user-generated content. *MIS Quart.* 43, 2 (2019), 623–647.

[68] Kenneth G. Manton. 2010. *National long-term care survey: 1982, 1984, 1989, 1994, 1999, and 2004*. Inter-University Consortium for Political and Social Research.

[69] James R. Marsden and David E. Pingry. 2018. Numerical data quality in IS research and the implications for replication. *Decision Support Systems* 115 (2018), A1-A7. https://doi.org/10.1016/j.dss.2018.10.007.

[70] Heiko Müller and Johann C. Freytag. 2003. *Problems, methods, and challenges in comprehensive data cleansing*. Humboldt University Berlin.

[71] R. R. Nelson, Peter A. Todd, and Barbara H. Wixom. 2005. Antecedents of information and system quality: An empirical examination within the context of data warehousing. *J. Management Inform. Systems* 21, 4 (2005), 199–235.

[72] Kimberly A. Neuendorf. 2017. *The content analysis guidebook*. Sage.

[73] Eric W. T. Ngai, Angappa Gunasekaran, Samuel F. Wamba, Shahriar Akter, and Rameshwar Dubey. 2017. Big data analytics in electronic markets. *Electronic Markets* 27, 3 (2017), 243–245. https://doi.org/10.1007/s12525-017-0261-6.

[74] Balaji Padmanabhan, Xiao Fang, Nachiketa Sahoo, and Andrew Burton-Jones. 2022. Machine learning in information systems research. *MIS Quart.* 46, 1 (2022), iii–xix.

[75] Yoon S. Park, Lars Konge, and Anthony Artino. 2020. The positivism paradigm of research. *Academic Medicine* 95, 5 (2020), 690–694. https://doi.org/10.1097/ACM.0000000000003093.

[76] Amir Parssian, Sumit Sarkar, and Varghese S. Jacob. 2004. Assessing data quality for information products: Impact of selection, projection, and cartesian product. *Management Sci.* 50, 7 (2004), 967–982. https://doi.org/10.1287/mnsc.1040.0237.

[77] Amir Parssian, Sumit Sarkar, and Jacob Varghese. 2009. Impact of the union and difference operations on the quality of information products. *Inform. Systems Res.* 20, 1 (2009), 99–120. https://doi.org/10.1287/isre.1070.0161.

[78] Jiaxu Peng, Jungpil Hahn, and Ke-Wei Huang. 2023. Handling missing values in information systems research: A review of methods and assumptions. *Inform. Systems Res.* 34, 1 (2023), 5–26.

[79] Barbara Pernici and Monica Scannapieco. 2003. Data quality in web information systems. *J. Data Semantics* (2003), 48–68.

[80] Stacie Petter, William DeLone, and Ephraim McLean. 2013. Information systems success: The quest for the independent variables. *J. Management Inform. Systems* 29, 4 (2013), 7–62.

[81] Leo L. Pipino, Yang W. Lee, and Richard Y. Wang. 2002. Data quality assessment. *Comm. ACM* 45, 4 (2002), 211–218. https://doi.org/10.1145/505248.506010.

[82] Rosanne Price and Graeme Shanks. 2005. A semiotic information quality framework: Development and comparative analysis. *J. Inform. Tech.* 20, 2 (2005), 88–102.

[83] Thomas C. Redman. 1996. *Data quality for the information age*. Artech House.

[84] Christopher M. Sauer, Li-Ching Chen, Stephanie L. Hyland, Armand Girbes, Paul Elbers, and Leo A. Celi. 2022. Leveraging electronic health records for data science: common pitfalls and how to avoid them. *The Lancet Digit. Health* 4, 12 (2022), e893-e898.

[85] Monica Scannapieco, Paolo Missier, and Carlo Batini. 2005. Data quality at a glance. *Datenbank-Spektrum* 14 (2005), 6–14.

[86] Julian Senoner, Torbjørn Netland, and Stefan Feuerriegel. 2022. Using Explainable Artificial Intelligence to Improve Process Quality: Evidence from Semiconductor Manufacturing. *Management Sci.* 68, 8 (2022), 5704–5723. https://doi.org/10.1287/mnsc.2021.4190.

[87] Palaniappan Shamala, Rabiah Ahmad, Ali Zolait, and Muliati Sedek. 2017. Integrating information quality dimensions into information security risk management (ISRM). *J. Inform. Security and Appl.* 36 (2017), 1–10. https://doi.org/10.1016/j.jisa.2017.07.004.

[88] Sabrina Sicari, Alessandra Rizzardi, Daniele Miorandi, Cinzia Cappiello, and Alberto Coen-Porisini. 2016. A secure and quality-aware prototypical architecture for the internet of things. *Inform. Systems* 58 (2016), 43–55. https://doi.org/10.1016/j.is.2016.02.003.

[89] Howard Veregin. 1999. Data quality parameters. *Geographical Inform. Systems* 1 (1999), 177–189.

[90] Yair Wand and Richard Y. Wang. 1996. Anchoring data quality dimensions in ontological foundations. *Comm. ACM* 39, 11 (1996), 86–95. https://doi.org/10.1145/240455.240479.

[91] Jing Wang, Panagiotis G. Ipeirotis, and Foster Provost. 2017. Cost-Effective Quality Assurance in Crowd Labeling. *Inform. Systems Res.* 28, 1 (2017), 137–158. https://doi.org/10.1287/isre.2016.0661.

[92] Richard Y. Wang and Diane M. Strong. 1996. Beyond accuracy: What data quality means to data consumers. *J. Management Inform. Systems* 12, 4 (1996), 5–33. https://doi.org/10.1080/07421222.1996.11518099.

[93] Karl Werder, Balasubramaniam Ramesh, and Rongen Zhang. 2022. Establishing data provenance for responsible artificial intelligence systems. *ACM Trans. Manage. Inf. Syst.* 13, 2 (2022). https://doi.org/10.1145/3503488.

[94] Jingjun Xu, Izak Benbasat, and Ronald Cenfetelli. 2013. Integrating service quality with system and information quality: An empirical test in the e-service context. *MIS Quart.* 37, 3 (2013), 777–794.

[95] Youngjin Yoo, Ola Henfridsson, Jannis Kallinikos, Robert Gregory, Gordon Burtch, Sutirtha Chatterjee, and Suprateek Sarker. 2024. The next frontiers of digital innovation research. *Inform. Systems Res.* 35, 4 (2024), 1507–1523. https://doi.org/10.1287/isre.2024.editorial.v35.n4.

[96] Yuval Zak and Adir Even. 2015. A continuous Markov-chain model of data quality transition: Application in insurance-claim handling. In *Proceedings of the 10th International Conference on Design Science Research in Information Systems*, 199–214. https://doi.org/10.1007/978-3-319-18714-3_13.

[97] Yuval Zak and Adir Even. 2017. Development and evaluation of a continuous-time Markov chain model for detecting and handling data currency declines. *Decision Support Systems,* 103 (2017), 82–93. https://doi.org/10.1016/j.dss.2017.09.006.

[98] Ruojing Zhang, Marta Indulska, and Shazia Sadiq. 2019. Discovering data quality problems. *Bus. Inform. Systems Engrg.* 61, 5 (2019), 575–593. https://doi.org/10.1007/s12599-019-00608-0.

9 Appendices

9.1 Appendix A.1: List of all 98 Works referred to in Table 3

[A1] Ziawasch Abedjan, Xu Chu, Dong Deng, Raul C. Fernandez, Ihab F. Ilyas, Mourad Ouzzani, Paolo Papotti, Michael Stonebraker, and Nan Tang. 2016. Detecting data errors. *Proc. VLDB Endowment* 9, 12 (2016), 993–1004.

[A2] Khalid Adam, Mazlina A. Majid, and Younis Ibrahim. 2023. Smart Campus Reliability Based on the Internet of Things. In *International Conference on Information Systems and Intelligent Applications*. Lecture Notes in Networks and Systems. Springer International Publishing, Cham, 353–360. https://doi.org/10.1007/978-3-031-16865-9_27.

[A3] Samir Al-janabi, Abubaker Hamid, and Ryszard Janicki. 2017. DatumPIPE: Data Generator and Corrupter for Multiple Data Quality Aspects. In *Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*. Association for Computing Machinery, New York, 589–592.

[A4] Wesley G. de Almeida, Rafael de Sousa, Flávio de Deus, Georges Nze, and Fábio de Mendoca. 2013. Taxonomy of Data Quality Problems in Multidimensional Data Warehouse Models. In *Proceedings of the 8th Iberian Conference on Information Systems and Technologies*, 1–7.

[A5] Omar Almutiry, Gary Wills, and Richard Crowder. 2015. A dimension-oriented taxonomy of data quality problems in electronic health records. *e-Society* (2015), 122–132.

[A6] Saad B. Alotaibi. 2017. ETDC: An Efficient Technique to Cleanse Data in the Data Warehouse. In *Proceedings of the International Conference on Advances in Image Processing*. Association for Computing Machinery, New York, 135–138.

[A7] Danielle Arts, Nicolette de Keizer, and Gert J. Scheffer. 2002. Defining and improving data quality in medical registries: A literature review, case study, and generic framework. *J. Am. Med. Inform. Assoc.* 9, 6 (2002), 600–611.

[A8] José Barateiro and Helena Galhardas. 2005. A survey of data quality tools. *Datenbank-Spektrum* 14, 48 (2005), 15–21.

[A9] Carlo Batini and Monica Scannapieco. 2016. Data Quality Issues in Data Integration Systems. In *Data and Information Quality*, Carlo Batini and Monica Scannapieco, Eds. Springer, Cham, 279–307.

[A10] Nisrine Berros, Youness Filaly, Fatna E. Mendili, and Younes E. B. E. L. Idrissi. 2024. Uncovering Data Quality Issues in Big Healthcare Data: Implications for Accurate

Analytics. In *Artificial Intelligence, Data Science and Applications*, Yousef Farhaoui, Amir Hussain, Tanzila Saba, Hamed Taherdoost and Anshul Verma, Eds. Lecture Notes in Networks and Systems. Springer Nature Switzerland, Cham, 499–505. https://doi.org/10.1007/978-3-031-48573-2_72.

[A11] Yannis Bertrand, Rafaël van Belle, Jochen de Weerdt, and Estefanía Serral. 2023. Defining Data Quality Issues in Process Mining with IoT Data. In *Process Mining Workshops*, Marco Montali, Arik Senderovich and Matthias Weidlich, Eds. Lecture Notes in Business Information Processing. Springer Nature Switzerland, Cham, 422–434. https://doi.org/10.1007/978-3-031-27815-0_31.

[A12] Shuchismita Biswas, Tianzhixi Yin, Syed Ahsan Raza Naqvi, Jim Follum, Antos Cheeramban Varghese, Tawsif Ahmad, and Pavel Etingov. 2025. An Open-Access Repository of Synchrophasor Data Quality Examples: Curation and Example Applications. *IEEE Access* 13 (2025), 66445–66457. https://doi.org/10.1109/ACCESS.2025.3560602.

[A13] Jagadeesh Bose, Mans R., and Wil van der Aalst. 2013. Wanna Improve Process Mining Results? In *IEEE Symposium on Computational Intelligence & Data Mining*. IEEE, Singapore, 127–134.

[A14] Michael Bowie, Edmon Begoli, Byung Park, and Jeevith Bopaiah. 2017. Towards an LSTM-Based Approach for Detection of Temporally Anomalous Data in Medical Datasets. In *Proceedings of the 2017 MIT International Conference on Information Quality*, Little Rock, 1–13.

[A15] Surajt Chaudhuri, Kris Ganjam, Venky Ganti, Rahul Kapoor, Vivek Narasayya, and Theo Vassilakis. 2005. Data Cleaning in Microsoft SQL Server 2005. In *Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data*, Baltimore, 918–920.

[A16] Ricardo J. Cruz-Correia, Pedro Rodrigues, Alberto Freitas, Filipa Almeida, Rong Chen, and Altamiro Costa-Pereira. 2009. Data quality and integration issues in electronic health records. *Inform. Discovery on Electronic Health Records* (2009), 55–95.

[A17] Csaba Csáki. 2018. Towards Open Data Quality Improvements Based on Root Cause Analysis of Quality Issues. In *Electronic Government*, Peter Parycek, Olivier Glassey, Marijn Janssen, Hans J. Scholl, Efthimios Tambouris, Evangelos Kalampokis and Shefali Virkar, Eds. Springer, Cham, 208–220.

[A18] Eduardo Dalcin. 2005. *Data Quality Concepts and Techniques Applied to Taxonomic Databases*. Doctoral dissertation. University of Southampton, Southampton.

[A19] Tamraparni Dasu. 2013. Data Glitches: Monsters in Your Data. In *Handbook of Data Quality*, Shazia Sadiq, Ed. Springer, Berlin, 163–178.

[A20] Xiaoou Ding, Hongzhi Wang, Genglong Li, Haoxuan Li, Yingze Li, and Yida Liu. 2022. IoT data cleaning techniques: A survey. *Intell. and Converged Netw.* 3, 4 (2022), 325–339. https://doi.org/10.23919/ICN.2022.0026.

[A21] Jill Dyché and Evan Levy. 2011. *Customer Data Integration: Reaching a Single Version of the Truth*. John Wiley & Sons.

[A22] Wayne Eckerson. 2002. Data warehousing special report: Data quality and the bottom line. *Appl. Development Trends* 1, 1 (2002), 1–9.

[A23] Mohamed Elbadri, Ahmed Altaher, and Sharafedeen Alkawan. 2024. Data Quality Considerations for ERP Implementation: Techniques for Effective Data Management. In *Information and Communications Technologies*. Communications in Computer and Information Science. Springer Nature Switzerland, Cham, 210–220. https://doi.org/10.1007/978-3-031-62624-1_17.

[A24] Adir Even and Gopalan Shankaranarayanan. 2009. Dual assessment of data quality in customer databases. *J. Data & Inform. Quality* 1, 3 (2009), 1–29.

[A25] Ralph Foorthuis. 2021. On the nature and types of anomalies: A review of deviations in data. *Int. J. Data Sci. and Analytics* 12, 4 (2021), 297–331.

[A26] J. A. Freitas, T. Silva-Costa, B. Marques, and Altamiro Costa-Pereira. 2010. Implications of Data Quality Problems within Hospital Administrative Databases. In *XII Mediterranean Conference on Medical and Biological Engineering and Computing*. Springer, Berlin, 823–826.

[A27] Christian Fürber. 2016. Semantic Technologies. In *Data Quality Management with Semantic Technologies*. Springer Gabler, Wiesbaden, 56–68.

[A28] Christian Fürber and Martin Hepp. 2010. Using Semantic Web Resources for Data Quality Management. In *Knowledge Engineering and Management by the Masses*. *17th International Conference on Knowledge Engineering and Knowledge Management*. Springer, Berlin, 211–225.

[A29] Christian Fürber and Martin Hepp. 2011. Towards a Vocabulary for Data Quality Management in Semantic Web Architectures. In *1st International Workshop on Linked Web Data Management*. Association for Computing Machinery, New York, 1–8.

[A30] Helena Galhardas. 2006. Data Cleaning and Transformation Using the AJAX Framework. In *Generative and Transformational Techniques in Software Engineering*. Springer, Berlin, 327–343.

[A31] Jerry Gao, Chunli Xie, and Chuanqui Tao. 2016. Big Data Validation and Quality Assurance - Issuses, Challenges, and Needs. In *IEEE Symposium on Service-Oriented System Engineering*. IEEE, Oxford, 433–441.

[A32] Mouzhi Ge and Markus Helfert. 2008. Modeling Data Quality in Information Chain. In *Proceedings of the International Conference on Business Innovation & Information Technology*. Logos, Berlin.

[A33] Ralf Gitzel. 2016. Data Quality in Time Series Data: An Experience Report. In *Proceedings of the CBI 2016 Industrial Track*, Paris, 41–49.

[A34] Kanika Goel, Sareh Sadeghianasl, Robert Andrews, Arthur ter Hofstede, Moe T. Wynn, Dakshi Kapugama Geeganage, Sander J. J. Leemans, James McGree, Rebekah Eden, Andrew Staib, and others. 2023. Digital health data imperfection patterns and their manifestations in an Australian digital hospital. In *Proceedings of the 56th Annual Hawaii International Conference on System Sciences,*, 3235–3244.

[A35] Arda Goknil, Phu Nguyen, Sagar Sen, Dimitra Politaki, Harris Niavis, Karl J. Pedersen, Abdillah Suyuthi, Abhilash Anand, and Amina Ziegenbein. 2023. A Systematic Review of Data Quality in CPS and IoT for Industry 4.0. *ACM Comput. Surv.* 55, 14s (2023), 1–38. https://doi.org/10.1145/3593043.

[A36] Theresia Gschwandtner, Johannes Gärtner, Wolfgang Aigner, and Silvia Miksch. 2012. A Taxonomy of Dirty Time-Oriented Data. In *Proceedings of the 7th International Conference on Availability, Reliability and Security*. IEEE Computer Society, Washington D.C., 58–72.

[A37] Solanki Gupta, Vivek K. Singh, and Sumit K. Banshal. 2024. Altmetric data quality analysis using Benford's law. *Scientometrics* 129, 7 (2024), 4597–4621. https://doi.org/10.1007/s11192-024-05061-9.

[A38] Sunil Gupta, Ravi S. Iyer, and Sanjeev Kumar. 2025. Digital Twin Challenges. In *Digital Twins*, Sunil Gupta, Ravi S. Iyer and Sanjeev Kumar, Eds. Springer Nature Switzerland, Cham, 19–42. https://doi.org/10.1007/978-3-031-76564-3_2.

[A39] Abraham G. Hartzema, Christian G. Reich, Patrick B. Ryan, Paul E. Stang, David Madigan, Emily Welebob, and J. M. Overhage. 2013. Managing data quality for a drug safety surveillance system. *Drug Safety* 36, 1 (2013), 49–58.

[A40] Bernd Heinrich and Mathias Klier. 2015. Metric-based data quality assessment — developing and evaluating a probability-based currency metric. *Decision Support Systems* 72 (2015), 82–96.

[A41] Joseph Hellerstein. 2008. Quantitative data cleaning for large databases. *United Nations Economic Commission for Europe* 25 (2008), 1–42.

[A42] Kaveen Hiniduma, Suren Byna, and Jean L. Bez. 2025. Data Readiness for AI: A 360-Degree Survey. *ACM Comput. Surv.* 57, 9 (2025), 1–39. https://doi.org/10.1145/3722214.

[A43] Carsten A. Holz. 2004. China's statistical system in transition: challenges, data problems, and institutional innovations. *Review of Income and Wealth* 50, 3 (2004), 381–409.

[A44] Kai Hüner. 2011. *Method for Specifying Business-Oriented Data Quality Metrics*. University of St. Gallen, Institute of Information Management, St. Gallen.

[A45] Ihab F. Ilyas and Xu Chu. 2015. Trends in cleaning relational data: Consistency and deduplication. *Foundations and Trends in Databases* 5, 4 (2015), 281–393.

[A46] Ihab F. Ilyas and Felix Naumann. 2022. Data Errors: Symptoms, Causes and Origins. *IEEE Data Eng. Bull.* 45, 1 (2022), 4–9.

[A47] Philip M. Johnson and Anne M. Disney. 1999. A critical analysis of PSP data quality: Results from a case study. *Empirical Software Engrg.* 4, 4 (1999), 317–349.

[A48] Joao Josko. 2018. A Formal Taxonomy of Temporal Data Defects. In *International Workshop on Data Quality and Trust in Big Data*. Springer, Cham, 94–110.

[A49] Joao Josko and Joao Ferreira. 2017. Visualization properties for data quality visual assessment: An exploratory case study. *Inform. Visualization* 16, 2 (2017), 93–112.

[A50] Joao Josko, Marcio Oikawa, and Joao Ferreira. 2016. A Formal Taxonomy to Improve Data Defect Description. In *International Conference on Database Systems for Advanced Applications*. Springer, Cham, 307–320.

[A51] Sean Kandel, Ravi Parikh, Andreas Paepcke, Joseph Hellerstein, and Jeffrey Heer. 2012. Profiler: Integrated Statistical Analysis and Visualization for Data Quality Assessment. In *Proceedings of the International Working Conference on Advanced Visual Interfaces*. Association for Computing Machinery, New York, 547–554.

[A52] Aimad Karkouch, Hajar Mousannif, Hassan Al Moatassime, and Thomas Noel. 2016. Data quality in internet of things: A state-of-the-art survey. *J. Network & Computer Appl.* 73 (2016), 57–81.

[A53] Arno Kesper, Viola Wenz, and Gabriele Taentzer. 2020. Detecting Quality Problems in Research Data: A Model-Driven Approach. In *Proceedings of the 23rd ACM/IEEE International Conference on Model Driven Engineering Languages and Systems*. Association for Computing Machinery, New York, 354–364.

[A54] Ritu Khare, Byron J. Ruth, Matthew Miller, Joshua Tucker, Levon H. Utidjian, Hanieh Razzaghi, Nandan Patibandla, Evanette K. Burrows, and L. C. Bailey. 2018. Predicting causes of data quality issues in a clinical data research network. *AMIA Summits in Translational Sci. Proc.* 2018 (2018), 113–121.

[A55] Won Kim, Byoung J. Choi, Eui K. Hong, Soo K. Kim, and Doheon Lee. 2003. A taxonomy of dirty data. *Data Mining and Knowledge Discovery* 7, 1 (2003), 81–99.

[A56] W. Kim and J. Seo. 1991. Classifying schematic and data heterogeneity in multidatabase systems. *Computer* 24, 12 (1991), 12–18.

[A57] Marilyn Kletke and Dursun Delen. 2005. Data Quality in Very Large, Multiple-Source, Secondary Datasets for Data Mining Applications. In *Proceedings of the 11th American Conference on Information Systems*. Association for Information Systems, Atlanta, 1512–1516.

[A58] Matthias Knapp and Florian Hasibether. 2011. Material Master Data Quality. In *17th International Conference on Concurrent Enterprising*. IEEE, New York, 1–8.

[A59] Zlatinka Kovacheva, Ina Naydenova, and Kalinka Kaloyanova. 2022. A Multidimensional Rendering of Error Types in Sensor Data. In *Intelligent Sustainable Systems*, Jennifer Raj, Ram Palanisamy, Isidoros Perikos and Yong Shi, Eds. Springer, Singapore, 139–149.

[A60] Nuno Laranjeiro, Seyma N. Soydemir, and Jorge Bernardino. 2015. A Survey on Data Quality: Classifying Poor Data. In *IEEE 21st Pacific Rim International Symposium on Dependable Computing*. IEEE, New York, 179-188.

[A61] Judith T. Lewis, Jeremy Stephens, Beverly Musick, Steven Brown, Karen Malateste, Cam Ha Dao Ostinelli, Nicola Maxwell, Karu Jayathilake, Qiuhu Shi, Ellen Brazier, Azar Kariminia, Brenna Hogan, and Stephany N. Duda. 2022. The IeDEA harmonist data toolkit: A data quality and data sharing solution for a global HIV research consortium. *Journal of biomedical informatics* 131 (2022), 104110. https://doi.org/10.1016/j.jbi.2022.104110.

[A62] Lin Li, Taoxin Peng, and Jessie Kennedy. 2011. A rule based taxonomy of dirty data. *GSTF J. on Comput.* 1, 2 (2011), 140–148.

[A63] Caihua Liu, Patrick Nitschke, Susan P. Williams, and Didar Zowghi. 2020. Data quality and the internet of things. *Comput.* 102, 2 (2020), 573–599.

[A64] Grace Liu. 2020. Data quality problems troubling business and financial researchers: A literature review and synthetic analysis. *J. Business & Finance Librarianship* 25, 3-4 (2020), 315–371.

[A65] Niels Martin. 2021. Data Quality in Process Mining. In *Interactive Process Mining in Healthcare*. *Health Informatics*, Carlos Fernandez-Llatas, Ed. Springer, Cham, 53–79.

[A66] Heiko Müller and Johann C. Freytag. 2003. Problems, methods, and challenges in comprehensive data cleansing. *Technical Report HUB-IB-164* (2003).

[A67] Henning Neumann, Moritz Müller-Roden, Seth Schmitz, Andreas Gützlaff, and Günther Schuh. 2022. Data Quality Assessment to Apply Process Mining in Production Processes. In *Production at the Leading Edge of Technology*. Lecture Notes in Production Engineering. Springer International Publishing, Cham, 515–524. https://doi.org/10.1007/978-3-030-78424-9_57.

[A68] Shinta Oktaviana R, Putu W. Handayani, Achmad N. Hidayanto, and Bambang B. Siswanto. 2025. Healthcare data governance assessment based on hospital management perspectives. *International Journal of Information Management Data Insights* 5, 1 (2025), 100342. https://doi.org/10.1016/j.jjimei.2025.100342.

[A69] Paulo Oliveira, Fátima Rodrigues, and Pedro Henriques. 2006. Data profiling versus data quality problems. *Actes du 2nd Atelier Qualité des données et des connaissances en conjonction avec EGC 2006* (2006), 9–15.

[A70] Paulo Oliveira, Fátima Rodrigues, and Pedro R. Henriques. 2005. A Formal Definition of Data Quality Problems. In *Proceedings of the 2005 International Conference on Information Quality*, Cambridge, 1–14.

[A71] Paulo Oliveira, Fátima Rodrigues, Pedro R. Henriques, and Helena Galhardas. 2005. A Taxonomy of Data Quality Problems. In *2nd International Workshop on Data & Information Quality*, 219–233.

[A72] Barbara K. A. Porfírio, Nicolle A. Adaniya, João M. B. Josko, and Márcio K. Oikawa. 2020. Classification of Errors in Geographic Data Using ISO 19157. In *Proceedings of the IEEE International Symposium on Geoscience & Remote Sensing*. IEEE, New York, 3231–3234.

[A73] Lydia Rabia, Idir A. Amarouche, and Kadda B. Bey. 2018. Rule-Based Approach for Detecting Dirty Data in Discharge Summaries. In *International Symposium on Programming and Systems*. IEEE, New York, 1–6.

[A74] Erhard Rahm and Hong H. Do. 2000. Data cleaning: Problems and current approaches. *IEEE Data Engrg. Bulletin* 23, 4 (2000), 3–13.

[A75] Andrea Reid and Miriam Catterall. 2005. Invisible data quality issues in a CRM implementation. *J. Database Marketing & Customer Strategy Management* 12, 4 (2005), 305–314.

[A76] Andrea Reid and Miriam Catterall. 2015. Hidden Data Quality Problems in CRM Implementation. In *Marketing, Technology and Customer Commitment in the New Economy*, Harlan E. Spotts, Ed. Springer, Berlin, 184–189.

[A77] Joachim Schmid. 2004. The Main Steps to Data Quality. In *Advances in Data Mining. 4th Industrial Conference on Data Mining*. Springer Science & Business Media, Berlin, 69–77.

[A78] Fatimah Sidi, Payam Panahy, Lilly Affendey, Marzanah Jabar, Hamidah Ibrahim, and Aida Mustapha. 2012. Data Quality: A Survey of Data Quality Dimensions. In *International Conference on Information Retrieval and Knowledge Management*. IEEE, New York, 300–304.

[A79] Jaspreeti Singh and Srishti Vashishtha. 2015. Data quality tools for datawarehouse models. *Int. J. Engrg. Sci. and Technology* 7, 5 (2015), 181–191.

[A80] Ranjit Singh and Kawaljeet Singh. 2010. A descriptive classification of causes of data quality problems in data warehousing. *Int. J. Computer Sci. Issues* 7, 3 (2010), 41–50.

[A81] Dina Sukhobok, Nikolay Nikolov, and Roman Dumitru. 2017. Tabular Data Anomaly Patterns. In *International Conference on Big Data Innovations and Applications*. Springer, Berlin, 25–34.

[A82] Zhenglong Sun, Hao Long, Zewei Li, Bo Gao, Xiaoqiang Wei, Yoash Levron, and Juri Belikov. 2025. Wide-Area damping control enhanced by non-invasive data verification against false data injection attacks. *Electric Power Systems Research* 242 (2025), 111448. https://doi.org/10.1016/j.epsr.2025.111448.

[A83] Kaustubh Supekar, Anup Marwadi, Yugyung Lee, and D. Medhi. 2020. Fuzzy Rule-Based Framework for Medical Record Validation. In *Intelligent Data Engineering and Automated Learning*. Springer, Berlin, 447–453.

[A84] Ikbal Taleb, Rachida Dssouli, and Mohamed A. Serhani. 2015. Big Data Pre-Processing: A Quality Framework. In *IEEE International Congress on Big Data*. IEEE, New York, 191–198.

[A85] Ikbal Taleb, Hadeel El Kassabi, Mohamed A. Serhani, Rachida Dssouli, and Chafik Bouhaddioui. 2016. Big Data Quality: A Quality Dimensions Evaluation. In *International IEEE Conference on Ubiquitous Intelligence and Computing, Advanced and Trusted Computing, Scalable Computing and Communication, Cloud and Big Data Computing, Internet of People, and Smart World Congress*. IEEE, New York, 759–765.

[A86] T. C. Ting, Fred R. Sias, and Eric Campbell. 1976. On Data Error Control Problems in Medical Information Systems. In *Proceedings of the 14th Annual Southeast Regional Conference*. Association for Computing Machinery, New York, 362–368.

[A87] Hong-Linh Truong and Ngoc N. T. Nguyen. 2024. TENSAI - Practical and Responsible Observability for Data Quality-aware Large-scale Analytics. *J. Data & Inform. Quality* 16, 4 (2024), 1–43. https://doi.org/10.1145/3708014.

[A88] Amjad Umar, George Karabatis, Linda Ness, Bruce Horowitz, and Ahmed Elmagardmid. 1999. Enterprise data quality: A pragmatic approach. *Inform. Systems Frontiers* 1, 3 (1999), 279–301.

[A89] Carolina Valverde, Adriana Marotta, José I. Panach, and Diego Vallespir. 2022. Towards a model and methodology for evaluating data quality in software engineering experiments. *Information and Software Technology* 151 (2022), 107029. https://doi.org/10.1016/j.infsof.2022.107029.

[A90] Larysa Visengeriyeva and Ziawasch Abedjan. 2020. Anatomy of metadata for data curation. *J. Data & Inform. Quality* 12, 3 (2020), 1–30.

[A91] Yair Wand and Richard Y. Wang. 1996. Anchoring data quality dimensions in ontological foundations. *Comm. ACM* 39, 11 (1996), 86–95.

[A92] Shuhui Wang, Yaguo Lei, Bin Yang, Xiang Li, Yue Shu, and Na Lu. 2023. A graph neural network-based data cleaning method to prevent intelligent fault diagnosis from data contamination. *Engineering Applications of Artificial Intelligence* 126 (2023), 107071. https://doi.org/10.1016/j.engappai.2023.107071.

[A93] Yuechen Wang and Jun Shen. 2024. A Study on SBAS-RTK Integrated Positioning Technologies. In *China Satellite Navigation Conference (CSNC 2024) Proceedings*, Changfeng Yang and Jun Xie, Eds. Lecture Notes in Electrical Engineering. Springer Nature Singapore, Singapore, 3–12. https://doi.org/10.1007/978-981-99-6932-6_1.

[A94] Wang Zhan, Serhan Dagtas, John Talburt, Ahmad Baghal, and Meredith Zozus. 2019. Rule-based data quality assessment and monitoring system in healthcare facilities. *Studies in Health Technology and Informatics* 257 (2019), 460–467.

[A95] Hui-Juan Zhang, Can-Can Chen, Peng Ran, Kai Yang, Quan-Chao Liu, Zhe-Yuan Sun, Jia Chen, and Jia-Ke Chen. 2024. A multi-dimensional hierarchical evaluation system for data quality in trustworthy AI. *J Big Data* 11, 1 (2024). https://doi.org/10.1186/s40537-024-00999-2.

[A96] Yili Zhang. 2018. Development of a collaborative data quality improvement approach for healthcare organizations. *UMBC Student Collection* (2018).

[A97] Yili Zhang and Güneş Koru. 2020. Understanding and detecting defects in healthcare administration data: Toward higher data quality to better support healthcare operations and decisions. *J. Am. Med. Inform. Assoc.* 27, 3 (2020), 386–395.

[A98] Arturs Zogla, Inga Meirane, and Edgars Salna. 2015. Analysis of Data Quality Problem Taxonomies. In *Proceedings of the 17th International Conference on Enterprise Information Systems*. SCITEPRESS - Science and Technology Publications, Lda, Setubal, 445–450.

9.2 Appendix A.2: Coverage of DQ Defects by our Specifications

Based on seven Groups i)-vii) we show that all 52 DQ defects from literature (as presented in Table 2) are indeed covered by our specifications (for sake of clarity only, we drop the time index in the Groups i)-v)):

i): This group comprises basic DQ defects where data f^D representing a feature f or data $f^D(d)$ representing a feature value $f(e)$ do not agree. This could, for example, be caused by misspelling. The Specifications (II) and (III) of accuracy cover this group by means of the (syntactical/semantical) mappings $\delta_A^f(f, f^D) = \begin{cases} true & \text{, if } f = f^D \\ false & \text{, if } f \neq f^D \end{cases}$ and $\delta_A^{fv}(f(e), f^D(d)) = \begin{cases} true & \text{, if } f(e) = f^D(d) \\ false & \text{, if } f(e) \neq f^D(d) \end{cases}$.

ii): This group comprises DQ defects where the meaning of data $f^D(d)$ representing a feature value is not precisely given. For example, abbreviations can be unclear. This means that two potential values $(f(e))_1$ and $(f(e))_2 \in FV$ seem to be accurately represented by $f^D(d)$, but indeed this is the case for only one of them (say, $\big(f(e)\big)_1$). Our proposed specification of accuracy covers this group of DQ defects as follows: Let $\delta_A^{fv}: (FV_t \times FV_t^D)_A \to \{true, false\}$ be a mapping—as introduced in the Theoretical Foundations and used in Specification (III)—which, as a semantical mapping, factors in the meaning of the elements of entity state space and data state space. Since $\big\{\big((f(e))_1, f^D(d)\big), \big((f(e))_2, f^D(d)\big)\big\} \subset (FV \times FV^D)_A$ holds and Specification (III) yields $\mathcal{S}_A^{fv}\big((f(e))_1, f^D(d)\big) = \delta_A^{fv}((f(e))_1, f^D(d)) = true$ and $\mathcal{S}_A^{fv}\big((f(e))_2, f^D(d)\big) = \delta_A^{fv}((f(e))_2, f^D(d)) = false$, this Specification shows that $(f(e))_2$ is not accurately expressed by $f^D(d)$ in a semantical manner.

iii): This group comprises DQ defects where the meaning of data f^D representing a feature f is not precisely given. For example, the data "last purchase" referring to the feature "timestamp of the last purchase" could also refer to the feature "item that was last purchased". This means that there is a feature f' such that it seems that this feature is represented by f^D, but this is actually not the case. Our proposed specification of accuracy on feature level (cf. Specification (II)) covers this group of DQ defects. Precisely, if f^D does not represent f', then $\psi^F(f^D) \neq f'$ and hence also $\mathcal{S}_A^f(f', f^D) = false$.

iv): This group comprises DQ defects where there is data $d \in D$ such that it seems that an entity $e \in E$ is represented by d, but actually does not. This could, for example, be the case if there are two similar persons and the considered data d represents one of the persons but not the other. Our proposed specification of accuracy on entity level (cf. Specification (I)) covers this group of DQ defects. Precisely, if e is not represented by d, then $\psi(d) \neq e$ and hence also $\mathcal{S}_A^e(e, d) = false$.

v): This group comprises DQ defects where for data $d \in D$ there is no entity $e \in E$ at all that is represented by d. This could, for example, be the case if a user account control of a company was manipulated by inserting a user account with access rights even though the fabricated user does not exist at all in the real world. Our proposed specification of accuracy for entities (cf. Specification (I)) covers this group of DQ defects. Precisely, if such a DQ defect occurs, then $\psi(d) = 0$ (cf. Definition 3) and hence $\psi(d) \neq e$ holds for all $e \in E$. As a direct result, Specification (I) yields $\mathcal{S}_A^e(e, d) = false$ for each $e \in E$.

vi): This group comprises DQ defects caused by a change of a feature or a feature value in the entity state space (because of an event-based state transition, cf. Definition 7) while the corresponding data in the data state space has not been updated accordingly. In particular, for $f_t \in F$ or $f_t(e_t) \in FV$ and the representing data $f_t^D \in F^D$ or $f_t^D(d_t) \in FV^D$ respectively, accuracy holds at $t = t_0$, but does not hold anymore at $t = t_1$. Thus, DQ defects in this group are covered by the proposed specifications of currency on feature level (cf. Specification (V)) and feature value level (cf. Specification (VI)). More precisely, if the DQ defect occurs, $(f_{t_1}, f_{t_1}^D) \in (F_{t_1} \times F_{t_1}^D)_C^{t_0}$

or $(f_{t_1}(e_{t_1}), f_{t_1}^D(d_{t_1})) \in (FV_{t_1} \times FV_{t_1}^D)_C^{t_0}$ holds, since $\mathcal{S}_A^f((f_{t_0}, f_{t_0}^D)) = true$ or $\mathcal{S}_A^{fv}((f_{t_0}(e_{t_0}), f_{t_0}^D(d_{t_0}))) = true$, and the respective specifications yield $\mathcal{S}_{C,t_0}^f(f_{t_1}, f_{t_1}^D) = \mathcal{S}_A^f(f_{t_1}, f_{t_1}^D) = false$ or $\mathcal{S}_{C,t_0}^{fv}\left(f_{t_1}(e_{t_1}), f_{t_1}^D(d_{t_1})\right) = \mathcal{S}_A^{fv}\left(f_{t_1}(e_{t_1}), f_{t_1}^D(d_{t_1})\right) = false$, respectively.

vii): This group comprises DQ defects caused by a change of the data representing a feature or a feature value (because of an event-based state transition, cf. Definition 7) while the corresponding feature or feature value still is the same as before. This could be, for example, the case if the data representing the marital status of a person is falsely updated in a database while the corresponding person in the real world does not change their marital status. In particular, for $f_t \in F$ or $f_t(e_t) \in FV$ and the representing data $f_t^D \in F^D$ or $f_t^D(d_t) \in FV^D$, respectively, accuracy holds at $t = t_0$, but does not hold any more at $t = t_1$. Thus, DQ defects in this group are covered by the proposed specifications of currency on feature level (cf. Specification (V)) and feature value level (cf. Specification (VI)). More precisely, if the DQ defect occurs, $(f_{t_1}, f_{t_1}^D) \in (F_{t_1} \times F_{t_1}^D)_C^{t_0}$ or $(f_{t_1}(e_{t_1}), f_{t_1}^D(d_{t_1})) \in (FV_{t_1} \times FV_{t_1}^D)_C^{t_0}$ holds, since $\mathcal{S}_A^f((f_{t_0}, f_{t_0}^D)) = true$ or $\mathcal{S}_A^{fv}((f_{t_0}(e_{t_0}), f_{t_0}^D(d_{t_0}))) = true$, and the respective specifications yield $\mathcal{S}_{C,t_0}^f(f_{t_1}, f_{t_1}^D) = \mathcal{S}_A^f(f_{t_1}, f_{t_1}^D) = false$ or $\mathcal{S}_{C,t_0}^{fv}\left(f_{t_1}(e_{t_1}), f_{t_1}^D(d_{t_1})\right) = \mathcal{S}_A^{fv}\left(f_{t_1}(e_{t_1}), f_{t_1}^D(d_{t_1})\right) = false$, respectively.

9.3 Appendix A.3: Details on Application in Mortality Prediction Task

In our analysis, we focused on the task of mortality prediction as described by [1] based on the MIMIC dataset [2]. Here, mortality predictions are made based on data collected during the first 48 hours of a patient's ICU stay. Data collected include Glasgow coma scale data, typically used to assess the level of consciousness of patients, as well as the values of clinical features such as temperature and glucose levels, which are subject to substantial DQ defects such as feature values not being updated, especially after drug administrations (i.e., currency defects). To illustratively apply our insights in this setting, we proceeded as follows:

We built on the data and ML pipeline proposed by [1] and published in the respective benchmark repository. Our main idea was to use our specifications to model drug administration as a DQ-related event that induce state transitions (cf. Definition 7) by changing the values of the clinical features affected by the respective drug. Note that, with regard to the dataset, this event is only one of many DQ-related events but it is one we analyzed systematically and in depth on the basis of the proposed specifications and state transitions. To incorporate the impact of this DQ-related event, we additionally extracted data on drug administrations from the raw data of the MIMIC dataset [2]. Specifically, we focused on ICU stays (table: ICUSTAYS) where the data was collected from the Metavision health information system due to the details provided regarding drug administration. Using the D_ITEMS and INPUTEVENTS_MV tables, we extracted start times,

dosages and related data of drug administration events, focusing on drug push administrations because of the more pronounced and straightforward effects on clinical features. As examples, we selected the drugs Insulin, Acetaminophen, and Hydralazine because of their frequent administration and their clear effect on glucose, temperature, and diastolic blood pressure, respectively, but other drugs could be used analogously. Based on the training data (as provided by the benchmark) and the drug administration event data, we automatically generated rules for the effects of the drug administration event on the respective clinical features, thus modeling event-based state transitions. For instance, one of the rules suggested that the administration of 1,000 mg of Acetaminophen reduced temperature by around 1.3 degrees Celsius after 1-2 hours if temperature had been strongly elevated beforehand (>38.5 degrees Celsius). These rules were used to give estimations of current values of clinical features in the benchmark data according to our specification of currency.

To evaluate the effect on ML performance for the mortality prediction task, we followed the pipeline for logistic regression classification (as provided by the benchmark) and restricted the analyses to those patients receiving at least one of the drugs Insulin, Acetaminophen, and Hydralazine (resulting in 2,084 instances in the training data and 416 instances in the test data). As a baseline, we evaluated the performance of classifying based on the Glasgow coma scale data (eye opening, motor response, verbal response) as well as the clinical features (glucose, temperature, diastolic blood pressure) when continuing the use of previously recorded data (i.e., just "filling forward" old data). Against this baseline, we evaluated the performance of classifying based on the same data but including the addressing of currency defects by event-based state transitions as outlined above. The results are presented in Table 9.

	Recall	**Precision**	**F1-Score**	**Accuracy**
Baseline	0.364	0.625	0.460	0.887
Addressing Currency Defects	**0.418**	**0.697**	**0.523**	**0.899**

Table 9. Results in the Mortality Prediction Task

Addressing currency defects led to improved ML performance in the challenging and high-stakes mortality prediction task, in particular evidenced by a recall improvement of 5.4 percentage points (~15% relative increase) and a precision improvement of 7.2 percentage points (~12% relative increase), which serves as a testimony for positive effects enabled by thorough consideration and systematic addressing of DQ defects based on solid foundations.

9.4 References

[1] Hrayr Harutyunyan, Hrant Khachatrian, David C. Kale, Greg Ver Steeg, and Aram Galstyan. 2019. Multitask learning and benchmarking with clinical time series data. Nature Scientific Data 6, 1 (2019), 96. https://doi.org/10.1038/s41597-019-0103-9.

[2] Alistair Johnson, Tom Pollard, and Roger Mark. 2016. *MIMIC-III clinical database (version 1.4)*, PhysioNet.

2.3 Paper 3: Piecewise Linear Transformation – Propagating Aleatoric Uncertainty in Neural Networks

Current Status	Full Citation
This paper is **accepted** and **published** in the *Proceedings of the 38th AAAI Conference on Artificial Intelligence* (2024)	Krapf, T., Hagn, M., Miethaner, P., Schiller, A., Luttner, L., & Heinrich, B. (2024). Piecewise linear transformation – Propagating aleatoric uncertainty in neural networks. In *Proceedings of the 38th AAAI Conference on Artificial Intelligence*, Vancouver, Canada. https://doi.org/10.1609/aaai.v38i18.30029

Abstract:

Real-world data typically exhibit aleatoric uncertainty which has to be considered during data-driven decision-making to assess the confidence of the decision provided by machine learning models. To propagate aleatoric uncertainty represented by probability distributions (PDs) through neural networks (NNs), both sampling-based and function approximation-based methods have been proposed. However, these methods suffer from significant approximation errors and are not able to accurately represent predictive uncertainty in the NN output. In this paper, we present a novel method, Piecewise Linear Transformation (PLT), for propagating PDs through NNs with piecewise linear activation functions (e.g., ReLU NNs). PLT does not require sampling or specific assumptions about the PDs. Instead, it harnesses the piecewise linear structure of such NNs to determine the propagated PD in the output space. In this way, PLT supports the accurate quantification of predictive uncertainty based on the criterion exactness of the propagated PD. We assess this exactness in theory by showing error bounds for our propagated PD. Further, our experimental evaluation validates that PLT outperforms competing methods on publicly available real-world classification and regression datasets regarding exactness. Thus, the PDs propagated by PLT allow to assess the uncertainty of the provided decisions, offering valuable support.

Keywords: Probabilistic inference, decision/utility theory, other foundations of reasoning under uncertainty, probabilistic programming, uncertainty representations

Please note that the overall formatting of the paper has been adjusted to the remainder of the dissertation.

1 Introduction

Neural networks (NNs) have been deployed in data-driven decision-making in many tasks and fields, such as autonomous driving (Chen et al. 2021; Huang et al. 2021), medical diagnostics (Takenaka et al. 2020; Yu et al. 2021), cyber security (Pawlicki, Kozik, and Choraś 2022; Vigneswaran et al. 2018), industrial processes (Nunez et al. 2020; Zhang et al. 2021), and many more. However, despite being applied in high-risk and safety-critical domains, traditional NNs only yield deterministic point estimates without further valid information about the confidence or uncertainty of their predictions (Gal 2016; Ayhan and Berens 2018). For example, this is particularly relevant for cyber threat intelligence data, which typically suffer from a high number of missing or uncertain values and extreme class imbalance (because actual cyber-attacks are rather scarce). Indeed, inherent scores such as the softmax output of a NN have been shown to be no valid measure for the confidence of the prediction (Hendrycks and Gimpel 2017; Nguyen, Yosinski, and Clune 2015). Exacerbating this issue, NNs are prone to overconfident predictions (Guo et al. 2017; Wilson and Izmailov 2020) and opaque due to their 'black-box' nature (Roy et al. 2019). Hence, the uncertainty of a prediction needs to be incorporated and accurately assessed in NN-based decision-making. Predictive uncertainty in the NN output can be induced by the ubiquitous noise in the data analyzed (aleatoric uncertainty), by uncertainty in the model and its parameters (epistemic uncertainty), or a combination of both (Gal 2016).

In recent years, many works on uncertainty in NNs have emerged, in which uncertainty is typically represented in probabilistic terms such as probability distributions (PDs). Most of these works focus on epistemic uncertainty, e.g., by incorporating uncertainty in the weights of the NN in a probabilistic setting (e.g., Gal 2016; Kendall and Gal 2017; Goulet, Nguyen, and Amiri 2021). In such Bayesian NNs, each weight parameter is treated as a PD, thus representing uncertainty in the NN itself. In addition, due to the high presence of noise and data quality defects in real-world data, other works focusing on aleatoric uncertainty and its effect on the NN output have been published, which aim to propagate aleatoric uncertainty (represented by PDs) through NNs (e.g., Abdelaziz et al. 2015; Gast and Roth 2018; Jin, Dundar, and Culurciello 2015). However, regarding exactness, these methods suffer from significant approximation errors and are unable to accurately quantify predictive uncertainty in the NN output. This is mainly due to restrictive and approximative assumptions about the PDs in the input, hidden, or output layer (e.g., the assumption of any post-activation PD being a Gaussian). Therefore, the following research question arises:
How can PDs representing aleatoric uncertainty in input data be propagated through NNs regarding exactness?

To address this research question, we propose Piecewise Linear Transformation (PLT) in this paper, a novel method for propagating aleatoric uncertainty. Our main idea is to harness the locally simple piecewise linear structure of NNs with piecewise linear activation functions, such as ReLU or Leaky ReLU NNs (cf. e.g., Hanin and Rolnick 2019; Sattelberg et al. 2020). Note that our method is still generally applicable, since any activation function can be approximated

by piecewise linear functions (Hu et al. 2020; Liao et al. 2023). Our main contributions can be summarized as follows:

- First, we propose PLT, a method for propagating PDs representing aleatoric uncertainty in the input data of a NN with piecewise linear activation functions. PLT makes no restrictive assumptions about the characteristics or type of the input PDs (e.g., an assumption about the input PD being Gaussian) and is thus able to propagate arbitrary PDs.
- Second, PLT supports the accurate quantification of predictive uncertainty based on the criterion exactness of the propagated PD in the output.
- Third, we show the exactness of PLT in theory by proving error bounds for our propagated PDs. We evaluate our method on several real-world datasets induced with aleatoric uncertainty represented by PDs, and validate exactness of our propagation compared to results of competing methods from literature.

2 Related Work

Literature already provides several works aiming to propagate PDs through NNs. These works can be structured in two groups: function approximation-based approaches and sample-based approaches.

The core idea of the first group is to approximate (possibly arbitrary) PDs in the input layer or the hidden layers of a NN with well-known, parametrical PDs to facilitate their propagation: Astudillo and Neto (2011) and Abdelaziz et al. (2015) assume the PDs in each layer to be Gaussian and focus on their propagation through sigmoid NNs. To this end, they approximate the sigmoid activation function with two piecewise exponential functions and derive closed-form formulas for the mean and variance of a post-activation Gaussian on this basis. Considering Leaky ReLU and ReLU NNs, Gast and Roth (2018) use Gaussians to approximate the post-activation PD of each neuron in each layer. To this end, they propose closed-form analytical formulas to obtain the optimal (with respect to the Kullback-Leibler divergence) means and variances for the approximating Gaussians. Jin, Dundar, and Culurciello (2015) follow a similar approach in the context of Convolutional NNs and propose formulas for the mean and variance of a Gaussian after a max-pooling layer. Titensky, Jananthan, and Kepner (2018) also assume the PDs in the hidden layers to be Gaussian. To estimate their mean and covariance matrix, the Extended Kalman Filter technique (cf. Julier and Uhlmann 1997) is applied. Zhang and Shin (2021) approximate input PDs with Gaussian Mixture Models (GMMs) and demonstrate their propagation through one activation layer. Their main idea is that GMMs with a sufficiently high number of components can approximate arbitrary PDs well. Then, the propagation of the whole PD is split up into simpler propagations of individual Gaussian components based on the Unscented Transform technique (cf. Julier and Uhlmann 2004).

In summary, the parametrical form of the approximating (or assumed) PD often enables a closed-form representation, leading to a straightforward propagation of PDs. However, since not all input and post-activation PD can be approximated well by a parametrical PD, propagation via

existing function approximation-based approaches results in substantial errors. This holds even for generic NNs when Gaussians are used to approximate post-activation PDs (for a theoretical analysis of the approximation error cf. Theorem 7 and 8 in Appendix D). Moreover, these substantial approximation errors are also evident in our experimental findings (cf. Section 'Evaluation'), providing empirical evidence. Hence, these approaches are not able to perform an exact or near exact propagation of arbitrary input PDs.

The second group of sample-based approaches aims to propagate PDs by mapping a set of samples through the NN based on which either characteristics of the output PD or the output PD itself should be derived. A very well-known sample-based approach is the Monte Carlo simulation which utilizes a high number of random samples drawn from the input PD (e.g., Abdelaziz et al. 2015; Truong 2021). After propagation through the NN, the output samples are used to aggregate an output PD. However, because drawing a very large number of random samples is associated with high computational cost, lightweight sample-based approaches are necessary and have been proposed in literature: Abdelaziz et al. (2015) suggest using the Unscented Transform (UT) technique to estimate the first two statistical moments of the output PD based on a set of specific, systematically chosen samples. Ji, Ren, and Law (2019) aim to find a lower-dimensional 'active' subspace (AS) of the input space which is designed to describe most of the variation of the NN as an input-to-output function. Then, as an approximation of the NN, a less complex response surface defined on the lower-dimensional subspace is estimated. Finally, Monte Carlo samples are propagated through the response surface to approximate the output PD. Smieja et al. (2018) use GMMs to model aleatoric uncertainty in the context of missing data. An analytical formula is derived for the first statistical moment (i.e., the mean) of the activated GMM after a first ReLU hidden layer. In this sense, the uncertainty is discarded after the first layer and the mean (as a single sample) is propagated through the NN, thus yielding a deterministic output only. Finally, Jia et al. (2019) propose an approach for uncertainty propagation through nonlinear systems which aims to compute the first statistical moments of the output PD based on their integral-based definitions. To solve the integrals efficiently, a sparse grid-based technique which identifies a small representative set of grid point samples is chosen.

These lightweight sample-based approaches share the drawback that the whole output PD has to be estimated based on very limited information. More precisely, this information about the output PD is either given by a small set of samples or a small number of statistical moments. However, because equality of a finite set of statistical moments of two PDs does not imply equality of the PDs themselves, this leads to substantial errors. For instance, a Gaussian and a uniform distribution with the same mean and variance still differ substantially due to their different shape. Indeed, we obtain significant errors in our experimental results for lightweight sample-based approaches for this reason (cf. Section 'Evaluation').

3 Uncertainty Propagation via Piecewise Linear Transformation

In this section, we present our method – Piecewise Linear Transformation (PLT) – for propagating PDs through NNs with piecewise linear activation functions (e.g., ReLU activation). To this end, we first establish the fact that such NNs can be represented by a piecewise linear function. Thus, we begin by deriving an exact formula for the propagation of an arbitrary PD through a single, affine linear mapping. Crucially for our method, we further extend this formula for NNs with piecewise linear activation functions utilizing their piecewise linear form. Finally, we show how PLT can be used to 1) exactly evaluate the propagated PD in arbitrary output space points and 2) obtain a piecewise constant form of the propagated PD on the whole output space.

We first elaborate on the mathematical structure of NNs with piecewise linear activation functions. Without loss of generality, we exclusively focus on NNs of such structure for the remainder of this paper. Following Sattelberg et al. (2020), we recall definitions of the necessary mathematical concepts of polytopes and piecewise linear functions. The piecewise linear structure of NNs has been recognized in literature and is key in the discussion of expressiveness and complexity of NNs (e.g., Hanin and Rolnick 2019).

Definition 1 (Polytope, polytopic subdivision). *Let $m \geq 0$ be an integer. A bounded convex polyhedron $A \subset \mathbb{R}^m$ is called a polytope. A polytopic subdivision of a bounded set $D \subset \mathbb{R}^m$ is a set of finitely many polytopes $\mathbb{A} = \{A_1, A_2, \dots, A_k\}$ such that $D = \bigcup_i A_i$. A polytopic subdivision is called disjoint if for every $i \neq j$ such that $\dim(A_i) = \dim(A_j)$ the intersection of A_i and A_j is either empty or a polytope of lower dimension $\dim(A_i \cap A_j) < \dim(A_i)$.*

Definition 2 (Piecewise linear function, linear regions). *A function $f: D \to \mathbb{R}^n$ is called piecewise linear with respect to a polytopic subdivision $\mathbb{A} = \{A_1, A_2, \dots, A_k\}$ of D if the restriction $f_i \coloneqq f|_{A_i}$ is an affine linear function for all $A_i \in \mathbb{A}$. In this case, we call $A_1, A_2, \dots, A_k$ linear regions of f.*

Definition 3 (NNs as piecewise linear functions). *Let $D \subset \mathbb{R}^m$ and $f: D \to \mathbb{R}^n$ be the function representing a NN (with n-dimensional output space). Then the function f has the form*

$$f(x) = \begin{cases} W_1 x + b_1, & if\ M_1 x \leq c_1, \\ W_2 x + b_2, & if\ M_2 x \leq c_2, \\ \dots & \\ W_t x + b_t, & if\ M_t x \leq c_t \end{cases} \tag{1}$$

with $W_i \in \mathbb{R}^{n \times m}, M_i \in \mathbb{R}^{u_i \times m}, b_i \in \mathbb{R}^n, c_i \in \mathbb{R}^{u_i}$ for $t \in \mathbb{N}, u_i \in \mathbb{N}, 1 \leq i \leq t$.

The inequalities $M_i x \leq c_i$ in Eq. 1 divide D into polytopes $A_i = \{x \in D \mid M_i x \leq c_i\}$, which are disjoint (in the sense of Definition 1) and satisfy $D = \bigcup_{i=1}^{t} A_i$. On each polytope A_i, f is given by an affine linear function defined by W_i and b_i. Thus, f is a piecewise linear function with respect to the linear regions A_i, which in particular also define a polytopic subdivision of D. In

order to rigorously define probability density functions (PDFs) on such subdivisions in the input, hidden, and output layers of NNs, we now introduce our notion of piecewise PDFs.

Definition 4 (Piecewise PDF). *Let $\mathcal{A} = \mathfrak{B}(D)$ be the Borel σ-Algebra of $D \subset \mathbb{R}^m$. We call a function $P: \mathcal{A} \to \mathbb{R}$ piecewise with respect to a set of subsets $\mathbb{A} = \{A_1, A_2, \dots, A_k | A_i \subset D\}$ if there exists a set of functions $\{p_i: A_i \to \mathbb{R}\}$, such that $P(X) = \sum_{i=1}^{k} \int_{A_i \cap X} p_i d\mu_{A_i}$ holds for all $X \in \mathcal{A}$. Hereby μ_{A_i} denotes the $\dim(A_i)$-dimensional Lebesgue measure. If P is additionally a PD on D, P is called a piecewise PD (PPD) on D with respect to $\mathbb{A}$, and the set of underlying functions p_i is called the piecewise PDF (PPDF).*

An example for a PPD and its associated PPDF is provided in Appendix A. In other words, a PD on D is piecewise with respect to a set $\mathbb{A} = \{A_1, A_2, \dots, A_k\}$ if it is described by parts of PDFs with each part being restricted to a subset $A_i \in \mathbb{A}$. In particular, any PD defined by a PDF $p: D \to \mathbb{R}$ satisfies this condition if $\dim(A_i) = m$ for all i. Crucially however, our notation of a PPDF allows to describe PDs which are (partially) defined by PDFs on degenerate polytopes. This case of a polytopic subdivision containing degenerate polytopes occurs regularly, especially in the output space of NNs as a consequence of degenerate linear matrices W_i (cf. Eq. 1). Hence, propagated PDs in general cannot be described by traditional PDFs (with respect to the Lebesgue measure of the output space), but a more general notion of PDFs as in Definition 4 is needed.

For the following discussions, we fix the notation of an arbitrary bounded subset $D \subset \mathbb{R}^m$ together with a disjoint polytopic subdivision $\mathbb{A} = \{A_1, A_2, \dots, A_k\}$. If D itself is a polytope, we may assume without loss of generality that D is not degenerate itself (i.e., that $\dim(D) = m$). Moreover, let $f: D \to \mathbb{R}^n$ be a function representing a NN such that f is piecewise linear with respect to $\mathbb{A}$. Further, let P be a PPD on D with a PPDF p with respect to $\mathbb{A}$ and functions p_i on A_i. In this setting, we refer to D as the input space and to P as the input PD.

Crucial for the exactness of our method, we next address dependencies between neurons in the same layer. Thus, we first model the input layer of a NN with $m \in \mathbb{N}$ real-valued input neurons as a multivariate random variable (RV) $\bar{X}: (\Omega, \mathcal{F}, \mu) \to \mathbb{R}^m$ for a probability space $(\Omega, \mathcal{F}, \mu)$. Following common notation, we denote the pushforward measure on $\mathbb{R}^m$ induced by the measure μ via the RV $\bar{X}$ by $\bar{X}_* \mu$.

As the RVs representing neurons in subsequent layers emanate from the same RV representing the neurons in the input layer, they obviously exhibit substantial dependencies (for a deeper analysis, cf. Appendix D). However, these dependencies are often disregarded in literature and independence between the univariate RVs representing single neurons in the same layer of a NN is typically assumed (Gast and Roth 2018; Wang, Shi, and Yeung 2016; Wu et al. 2018; Jin, Dundar, and Culurciello 2015). Even for the input layer, this assumption of independence severely restricts the set of input PD that can be considered for propagation. In Theorem 8 (Appendix D) we prove that even in the simple case of Gaussians, disregarding dependency substantially impairs the exactness of the propagation. Therefore, instead of considering univariate PDFs of each neuron in a layer individually, we utilize their joint PPDF as a whole. Hence, we preserve

dependencies between the neurons in all layers of the NN (including the input layer) because the information of dependency is contained in the joint PPDF. Note that the PPDFs of single neurons can easily be deduced from the joint PPDF by aggregating their respective marginal PPDF.

For the propagation of a PPDF through a NN via PLT, we first determine the linear regions that the input PPDF lies in, and then aggregate the desired output PPDF based on the formulas for PDFs on single linear regions. Therefore, let $W\colon \mathbb{R}^m \to \mathbb{R}^n$ be a surjective linear map and p_X be a (traditional) PDF describing the input PD P on a linear region. As any linear map W is surjective onto its image, we may replace W by $W\colon \mathbb{R}^m \to \mathrm{Im}(W)$ without loss of generality. By the Radon-Nikodym Theorem the pushforward measure W_*P admits a PDF with respect to the n-dimensional Lebesgue measure. We denote this PDF by W_*p_X, i.e., $W_*P = W_*p_X\mathrm{d}x$. In other words, the pushforward of the PD P admits the pushforward of the PDF p_X as a PDF again.

Theorem 1 (Propagation of PDFs through linear operators). *Let $W\colon \mathbb{R}^m \to \mathbb{R}^n$ be a surjective linear map and p_X be a PDF as above. Then $p_Y = W_*p_X$ follows the formula*

$$p_Y(y) = \int_{W^{-1}(y)} p_X(x)\left|\det \widetilde{W}^{-1}\right|\mathrm{d}x$$

almost everywhere. Hereby $\widetilde{W}$ is given by the restriction of W to $\ker W^{\perp}$. If W is bijective, this formula simplifies to

$$p_Y(y) = p_X(W^{-1}y)\left|\det W^{-1}\right|.$$

Proof. Let $Y \subset \mathbb{R}^n$ be a measurable subset and $\pi\colon \mathbb{R}^m \to \ker W^{\perp}$ the orthogonal projection. By transformation of variables, we get

$$\begin{aligned}\int_Y p_Y(y)\mathrm{d}y &= \int_{W^{-1}Y} p_X(x)\mathrm{d}x = \int_{\ker W + \pi(W^{-1}Y)} p_X(x)\mathrm{d}x \\ &= \int_{\pi(W^{-1}Y)} \int_{\ker W} p_X(x+y)\mathrm{d}x\mathrm{d}y = \int_Y \int_{\ker W} p_X\left(x+\widetilde{W}^{-1}y\right)\left|\det \widetilde{W}^{-1}\right|\mathrm{d}x\mathrm{d}y.\end{aligned}$$

Since a measurable preimage under W exists for any measurable subset $Y \subset \mathbb{R}^n$, this yields the desired identity almost everywhere. □

Theorem 1 presents an exact formula for the propagation of PDFs through affine linear operators representing the NN on its linear regions. By utilizing joint PDFs in this formula, we preserve the dependencies between the neurons and thus retain crucial information for exact propagation. Next, we extend this result to piecewise linear functions and PPDFs.

Theorem 2 (Propagation of PPDFs through piecewise linear functions). *For all $1 \le i \le k$ let f_i denote the affine linear map such that $f|_{A_i} = f_i$. Then the (propagated) PPD f_*P is represented by the PPDF f_*p defined by*

$$f_*p(y) = \sum_{\substack{i=1 \\ y\in f_i(A_i)}}^{k} \frac{1}{\det \widetilde{f_i}} \int_{f_i^{-1}(y)\cap A_i} p_i(x)\mathrm{d}x. \tag{2}$$

In this formula, $\widetilde{f_i}$ denotes the restriction of f_i to $ker(f_i)^{\perp}$. Moreover, the integrals are defined with respect to the $\dim(A_i)$-dimensional Lebesgue measure.

Proof. First consider a single polytope $A_i \in \mathbb{A}$ together with $p_i: A_i \to \mathbb{R}$ and the function $f_i: \mathbb{R}^m \to \mathbb{R}^n$. The function p_i defines a measure M on A_i and we aim to find $f_* p_i$ on $\mathbb{R}^n$ which defines the pushforward measure $f_* M$. This problem can be traced back to the formula for linear operators by extending p_i to a function $\overline{p_i}$ on $\mathbb{R}^m$ by

$$\overline{p_i}(x) \coloneqq \begin{cases} p_i(x), & if\ x \in A_i, \\ 0, & else. \end{cases}$$

The pushforward measure associated to $\overline{p_i}$ is also given by $f_* M$, and from Theorem 1 (with $(f_i)_* p_i = p_Y$) we obtain

$$(f_i)_* p_i(y) = \frac{1}{\det(\widetilde{f_i})} \int_{f_i^{-1}(y)} \overline{p_i}(x) \mathrm{d}x = \frac{1}{\det(\widetilde{f_i})} \int_{f_i^{-1}(y) \cap A_i} p_i(x) \mathrm{d}x.$$

Note that $(f_i)_* p_i(y)$ only attains nonzero values if $y \in f(A_i)$. Hence, it is fully described by its restriction to $f(A_i) = f_i(A_i)$, which also is a polytope. As the intersection of images of different polytopes under f can be non-empty, the formula for the pushforward of a PPD along f has to be obtained by taking the sum over all polytopes in the subdivision $\mathbb{A}$. Moreover, the PD $f_* P$ on $f(D)$ is also piecewise and can be described by a PPDF with respect to $f(\mathbb{A}) \coloneqq \{f(A_1), \dots, f(A_k)\}$ and the functions $(f_i)_* p_i: f(A_i) \to \mathbb{R}$. In summary, the statement given by Eq. 2 follows. □

The formula in Eq. 2 for input PD propagation is now applicable to NNs, which poses a vital step of our PLT method proposed in this paper. However, it involves computing integrals of the form as in Eq. 2, which is not possible in closed form for arbitrary PPDFs, e.g., PPDFs without a closed analytical formula. In order to cope with this problem, we next introduce a grid-based approximation technique. We simplify the above formula in Eq. 2 for such approximate PPDFs, enabling a tractable propagation. In particular, by approximating the PPDF on a fine-grained grid, the shape of the input PD can be preserved all the way through the propagation process.

In a first step, we approximate p with another piecewise constant PPDF p'. We construct this approximate PPDF to be piecewise with respect to a very fine-grained grid of polytopes to minimize the approximation error. A discussion of this theoretical error is part of the following section and is deepened further in Appendix C. More precisely, the initial polytopic subdivision $\mathbb{A} = \{A_1, A_2, \dots, A_k\}$ of D is refined as follows: First, each polytope A_i is subdivided into $\dim(A_i)$-dimensional simplices by applying Delaunay triangulation (Delaunay 1934). As a result, we obtain a (more fine-grained) polytopic subdivision of D solely consisting of simplices. Moreover, any subdivision of this form can be subdivided further using the edgewise subdivision technique (cf. Edelsbrunner and Grayson 2000) satisfying the property that for any $b \in \mathbb{N}$, each simplex of dimension d can be subdivided into b^d sub-simplices of dimension d and equal volume. In this way, an arbitrarily, evenly fine-grained grid can be obtained (Edelsbrunner and Grayson 2000, also cf. Appendix B for more details).

To control the granularity of the grid, we fix a small threshold $\varepsilon > 0$. We require that for each (simplicial) polytope A in the final subdivision, every edge of A is shorter than ε (i.e.,

$dist(c_1, c_2) < \varepsilon, \forall c_1, c_2 \in C(A)$ with $C(A)$ denoting the set of all vertices of A). This can either be achieved by choosing the subdivision parameter b large enough or by iteratively applying the edgewise subdivision. The result is a fine-grained polytopic subdivision

$$\mathbb{A}_{edge}^{\varepsilon} = \{A_{1,1}^{\varepsilon}, A_{1,2}^{\varepsilon}, \dots, A_{1,l_{1,\varepsilon}}^{\varepsilon}, A_{2,1}^{\varepsilon}, \dots, A_{k,l_{k,\varepsilon}}^{\varepsilon}\}, \tag{3}$$

where $l_{i,\varepsilon}$ denotes the number of edgewise simplices of the polytope A_i, and the length of any edge of any simplex $A_{i,j}^{\varepsilon} \subset A_i$ is smaller than ε. We then define the approximative PPDF p' by

$$\begin{gathered} p' \colon D \to \mathbb{R}, p'(x) = c_{ij} \\ \text{with } c_{ij} \coloneqq \frac{1}{\dim(A_i) + 1} \sum_{c \in C\left(A_{i,j}^{\varepsilon}\right)} p_i(c), \\ \text{if } x \in A_{i,j}^{\varepsilon} \in \mathbb{A}_{edge}^{\varepsilon}. \end{gathered} \tag{4}$$

By this definition, p' is constant on each edgewise simplex $A_{i,j}^{\varepsilon}$, attaining the average value of p evaluated in the vertices of $A_{i,j}^{\varepsilon}$. Applying Theorem 2 to p' now yields the following result:

Theorem 3. *If the PPDF p is approximated using the piecewise constant PPDF p' given by Eq. 4, the formula from Theorem 2 simplifies to:*

$$f_* p'(y) = \sum_{i=1}^{k} \sum_{\substack{j=1, \\ y \in f_i(A_{i,j})}}^{l_i} \frac{c_{ij}}{\det \widetilde{f_i}} \operatorname{vol}\left(f_i^{-1}(y) \cap A_{i,j}\right). \tag{5}$$

Proof. Follows directly from Theorem 2 and the definition of piecewise constant PPDFs in Eq. 4. □

With Eq. 5, we have derived a formula based on PLT to evaluate the propagated PPDF at any point $y \in \mathbb{R}^n$ in the output space. Hence, we can obtain the propagated PPD as a whole by iteratively applying Eq. 5 on the points of a grid over the output space (e.g., the grid induced by the edgewise subdivision on the output space as described above).

Moreover – instead of evaluating a set of grid points in the output space – PLT can also be used to infer the whole propagated PPDF from an input grid (for which again the edgewise subdivision is a natural choice). More precisely, the core idea is to map the input grid points together with their closely adjacent probability mass into the output space via the simple linear operator of the respective linear region. Finally, the probability mass is assigned to a cuboid bin in the output space. By defining a fine-grained grid of such cuboid bins on the output space, we also obtain the output PPDF in a piecewise constant form, similar to a (multidimensional) histogram. In Theorem 5 (cf. Appendix C) we show that this piecewise constant form also can be rendered arbitrarily exact for bins with increasingly small volume.

Formally, let $\mathbb{A}_{edge}$ be the edgewise subdivision of the input space as in Eq. 3. Then, the function f representing the NN is also affine linear on each $A_{i,j} \in \mathbb{A}_{edge}$. Moreover, let S be a bin grid of disjoint bins of equal size on the output space. For a n-dimensional output space we can choose cuboid bins with side lengths $l_1, \dots, l_n$ for each dimension, ensuring equal size of bins and easier

computation. Since on each polytope $A_{i,j}$ with center t_{ij} the input PPDF is given by a constant value $c_{ij} \in \mathbb{R}$ (cf. Eq. 4), the probability mass of $A_{i,j}$ (equal to $c_{ij} \cdot \mathrm{vol}(A_{ij})$) can be propagated and assigned to the bin $B \in S$ containing $f(t_{ij})$. Thus, the estimated constant PPDF on a bin B is given by

$$f_* p(y) = \frac{1}{l_1 \cdot \ldots \cdot l_n} \sum_{i=1}^{k} \sum_{\substack{j \\ f(t_{ij}) \in B}} \mathrm{vol}(A_{i,j}) \cdot c_{ij} \tag{6}$$

for all $y \in B$. While the formula in Eq. 5 is particularly advantageous when evaluating the output PPDF in a certain set of points, propagation via Eq. 6 yields the output PPDF as a whole, since all probability mass in the input is propagated and assigned to the respective bin in the output space. In Theorem 5 (cf. Appendix C) we prove that for increasingly fine subdivision $\mathbb{A}_{edge}$ and bin grid S this PPDF described in Eq. 6 becomes arbitrarily exact.

4 Evaluation

We first provide a theoretical evaluation by establishing the mathematical exactness of PLT. We discuss rigorous bounds for possible approximation errors made by PLT for which detailed proofs are provided in Appendix C. In Lemma 3 (cf. Appendix C) we show that for piecewise constant approximations of arbitrary input PDs as defined in Eq. 4, the approximation error converges to zero for increasingly small diameters (i.e., the maximum distance of two vertices) of edgewise simplices $A_{i,j}^{\varepsilon}$. As ε can be chosen arbitrarily small and the diameter of any $A_{i,j}^{\varepsilon}$ is smaller than ε by construction of the edgewise subdivision, the approximation error also becomes arbitrarily small. As a direct consequence of the simple piecewise linear form of NNs, we further show in Theorem 4 (cf. Appendix C) that the error after propagation through the NN in the output space is always bounded by the approximation error in the input space. From the combination of these results the theoretical exactness of our method and our formula of the propagated PD as in Eq. 6 follows: Since we obtain an approximative piecewise constant output PPDF with respect to a cuboid bin grid, the approximation error in the output again becomes arbitrarily small for an increasingly fine-grained bin grid. Hence, the PPDF in Eq. 6 indeed converges towards the true output PD (for details cf. Theorem 5, Appendix C).

In the following, we aim to substantiate our theoretical results with empirical evidence. To this end, we evaluate our method on a broad range of publicly available real-world datasets from various domains for both classification and regression tasks. Details about the datasets are provided in Table 2 (cf. Appendix E). We randomly split each dataset into training and test dataset and train a standard ReLU NN for classification or regression depending on the task associated to the dataset.

Since the datasets do not exhibit aleatoric uncertainty represented by PDs, we induce uncertainty according to the following procedure: In a first step, we analyze each dataset regarding feature importance. On this basis, we select a set of features with high feature importance. Each feature

value of these selected features is labeled as uncertain with a fixed probability (e.g., 50 percent, cf. '%unc' in Table 1). By focusing on features with high importance for the output result, we make sure that the induced aleatoric uncertainty indeed affects the NN prediction and non-trivial predictive uncertainty can be observed. For each data instance containing uncertain feature values, the multiple imputation method MICE (van Buuren and Groothuis-Oudshoorn 2011) is deployed (as if the values were missing), leading to a set of multiple values suggested for imputation. Finally, for each instance we apply a Gaussian kernel density estimation on this set of imputation values, thus obtaining a (potentially multivariate) continuous PD[1] over its uncertain features.

To evaluate our method with respect to the criterion exactness, the deviation of the propagated PD obtained by our method from an exact ground truth has to be quantified. We generate this ground truth output PD by utilizing a Monte Carlo simulation with a very high sample count. To ensure a high quality of this ground truth, we start with a fixed number of samples that are propagated through the NN and compare the resulting output PD to a more refined PD obtained analogously but with twice the number of samples. This process is iterated until this increasingly exact sequence of PDs converges, i.e., until the L1-distance between two consecutive PDs falls under a fixed, small threshold. The generation of a high-quality ground truth via Monte Carlo simulation comes with enormous computational effort, which we undertook once for each data instance in the datasets analyzed.

The PDs propagated by PLT and the other existing and applicable methods from literature[2] (cf. Related Work) are then evaluated against this ground truth. More precisely, we calculate L1-, L2-, and Hellinger distance between the ground truth and the PD obtained by each method as these are well-known and meaningful standard metrics for PDFs (for definitions and more details, cf. Appendix C). Evaluating deterministic performance metrics such as accuracy or mean squared error is not desirable in this context because of two reasons: First, the output PDs would be condensed into single, deterministic values, resulting in a significant loss of information. Second, the uncertainty-based ground truth often induces 'true' labels different to the ones given by the certain dataset. By quantifying PDF-based distances between the PD and the ground truth, more general metrics accounting for both of these points are considered.

The experimental results presented in Table 1 substantiate that PLT significantly outperforms existing methods regarding exactness across all metrics, thus confirming our theoretical results and validating the ability of our method for accurate uncertainty propagation with respect to exactness.

[1] The uncertain datasets resulting from this procedure are provided in the supplementary material.

[2] Code for the approach of Zhang and Shin (2021) was requested, but not provided by the authors. Therefore, this method could not be considered in the evaluation.

Data-sets	% unc	PLT (ours)			ADF (Gast and Roth)			AS (Ji, Ren, and Law)			UT (Abdelaziz et al.)		
		L1	L2	H	L1	L2	H	L1	L2	H	L1	L2	H
Appen-dicitis	25	**0.029**	**1.048**	**0.040**	1.623	27.490	0.813	0.333	8.129	0.198	0.889	25.615	0.539
	50	**0.045**	**3.381**	**0.052**	1.293	68.298	0.669	0.969	20.239	0.497	0.983	67.555	0.556
Banana	25	**0.114**	**0.694**	**0.104**	1.332	1.995	0.759	1.198	2.167	0.609	1.018	1.985	0.656
	50	**0.079**	**0.316**	**0.072**	1.346	1.434	0.761	1.313	1.780	0.657	1.037	1.433	0.669
Balance	25	**0.038**	**1.375**	**0.047**	0.793	40.805	0.405	0.502	19.539	0.264	0.654	38.936	0.321
	50	**0.044**	**1.222**	**0.042**	0.776	27.221	0.383	0.560	18.718	0.298	0.625	27.439	0.289
Bands	25	**0.130**	**22.331**	**0.078**	1.738	134.118	0.875	0.422	25.562	0.249	14.978	426.529	1.067
	50	**0.220**	**15.676**	**0.121**	1.695	76.088	0.861	0.452	18.525	0.270	8.602	221.450	1.238
Boston	25	**0.131**	**0.661**	**0.097**	0.859	1.734	0.429	0.368	1.078	0.193	0.565	1.330	0.301
	50	**0.083**	**0.156**	**0.065**	0.782	1.170	0.380	0.300	0.578	0.162	0.516	0.750	0.259
Breast	25	**0.151**	**5.301**	**0.099**	1.628	64.911	0.827	0.281	18.933	0.161	9.204	151.424	1.052
	50	**0.230**	**5.857**	**0.131**	1.632	50.158	0.826	0.281	9.110	0.173	22.927	486.339	1.431
Califor-nia	25	**0.147**	**11.925**	**0.074**	0.987	97.197	0.617	0.335	29.937	0.193	0.898	73.451	0.458
	50	**0.304**	**19.492**	**0.140**	0.836	64.378	0.435	0.415	30.829	0.242	0.750	54.400	0.387
Diabe-tes	25	**0.014**	**0.078**	**0.009**	0.534	3.057	0.266	0.259	1.588	0.136	0.420	2.235	0.219
	50	**0.012**	**0.059**	**0.008**	0.522	2.556	0.250	0.235	1.284	0.121	0.415	1.943	0.210
Iris	25	**0.023**	**0.330**	**0.037**	1.854	11.643	0.920	0.564	6.082	0.333	0.999	11.710	0.695
	50	**0.029**	**0.153**	**0.035**	1.856	4.204	0.907	0.380	1.368	0.259	0.999	4.257	0.701
Ma-chine	25	**0.031**	**5.032**	**0.027**	0.873	121.193	0.422	0.180	20.733	0.099	0.736	95.224	0.345
	50	**0.034**	**4.652**	**0.032**	0.820	88.004	0.392	0.186	16.689	0.103	0.661	67.060	0.305
Real estate	25	**0.021**	**2.381**	**0.011**	0.596	77.677	0.294	0.133	12.939	0.076	0.399	46.363	0.191
	50	**0.015**	**1.752**	**0.009**	0.613	70.441	0.296	0.148	13.043	0.083	0.430	44.167	0.207
Vehicle	25	**0.035**	**26.123**	**0.029**	0.887	460.033	0.471	0.103	60.106	0.059	0.681	275.497	0.340
	50	**0.045**	**25.491**	**0.038**	0.888	396.444	0.467	0.100	52.507	0.049	0.718	268.618	0.358
Wine	25	**0.042**	**7.587**	**0.043**	1.742	159.540	0.882	0.592	61.310	0.326	1.036	160.301	0.704
	50	**0.043**	**41.778**	**0.046**	1.736	902.390	0.870	0.801	115.544	0.422	1.696	910.188	0.778

Table 1: Experimental Results (cf. Table 2 in Appendix E for details on the datasets)

5 Discussion and Conclusion

In this paper, we proposed PLT, a novel method for the propagation of aleatoric uncertainty through NNs that is applicable to arbitrary PDs in the input space. To this end, we introduced the notion of PPDFs, which allows to generalize the concept of PDFs to the polytopic subdivisions occurring in NNs. By propagating a joint PD across all uncertain features, our method is able to preserve vital dependencies between neurons in each NN layer. We provided mathematical proofs that neuron dependencies must be considered as the simplifying assumption of independence leads to large approximation errors (cf. ADF in Table 1). Further, we showed that our method is able to approximate the true propagated PD up to an arbitrarily small error, allowing us to accurately quantify predictive uncertainty. We evaluated our method on a broad range of real-world datasets for both classification and regression tasks, where it achieved higher exactness than competing methods from literature. In particular, our evaluation shows that methods making restrictive assumptions (ADF and UT), such as independence of neurons in NN layers or Gaussian form of PDs, exhibit high errors regarding exactness, thus confirming our theoretical findings. Similarly, a competing sample-based method (AS) suffered from the drawback that complex PDs have to be estimated from limited information and yielded a worse exactness than PLT.

Moving forward, our method has broad application potential across various domains, particularly in areas where uncertainty and its quantification play a pivotal role, such as medicine, finance, and high-risk environments (e.g., self-driving cars). One limitation to acknowledge is that PLT requires an already trained NN to propagate aleatoric uncertainty. Hence, the quality of our

propagated PD is subject to the quality of the given NN model. Moreover, despite posing a potential starting point, uncertainty during the training phase of a NN is not yet addressed by PLT. Additionally, our results only hold for NNs with piecewise linear activation functions, thus PLT cannot be directly applied to NNs with other activation functions. However, as mentioned earlier, any activation function can be approximated by a piecewise linear function in order to apply PLT. Another intriguing direction to explore is the integration of PLT into different NN architectures (e.g., CNNs). These challenges represent interesting avenues for future research.

6 References

Abdelaziz, A. H.; Watanabe, S.; Hershey, J. R.; Vincent, E.; and Kolossa, D. 2015. Uncertainty Propagation through Deep Neural Networks. In Interspeech.

Astudillo, R. F. and Neto, J. 2011. Propagation of Uncertainty through Multilayer Perceptrons for Robust Automatic Speech Recognition. In Interspeech.

Ayhan, M. S. and Berens, P. 2018. Test-Time Data Augmentation for Estimation of Heteroscedastic Aleatoric Uncertainty in Deep Neural Networks. In Proceedings of the 1st Conference on Medical Imaging with Deep Learning.

Chen, L.; Lin, S.; Lu, X.; Cao, D.; Wu, H.; Guo, C.; Liu, C.; and Wang, F.-Y. 2021. Deep Neural Network Based Vehicle and Pedestrian Detection for Autonomous Driving: A Survey. *IEEE Transactions on Intelligent Transportation Systems*. 22 (6): 3234–3246. doi.org/10.1109/tits.2020.2993926.

Delaunay, B. 1934. Sur la Sphere Vide. *Bulletin of Academy of Sciences of the USSR*. 7 (6): 793–800.

Edelsbrunner, H. and Grayson, D. R. 2000. Edgewise Subdivision of a Simplex. *Discrete and Computational Geometry*. 24 (4): 707–719. doi.org/10.1007/s4540010063.

Gal, Y. 2016. Uncertainty in Deep Learning. PhD dissertation, Department of Engineering, University of Cambridge, Cambridge.

Gast, J. and Roth, S. 2018. Lightweight Probabilistic Deep Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition: 3369–3378. IEEE Computer Society.

Goulet, J.-A.; Nguyen, L. H.; and Amiri, S. 2021. Tractable Approximate Gaussian Inference for Bayesian Neural Networks. *The Journal of Machine Learning Research*. 22 (1): 11374–11396. doi.org/10.5555/3546258.3546509.

Guo, C.; Pleiss, G.; Sun, Y.; and Weinberger, K. 2017. On Calibration of Modern Neural Networks. In Proceedings of the 34th International Conference on Machine Learning: 1321–1330.

Hanin, B. and Rolnick, D. 2019. Deep ReLU Networks Have Surprisingly Few Activation Patterns. In Proceedings of the 33rd Conference on Neural Information Processing Systems: 361–370.

Hendrycks, D. and Gimpel, K. 2017. A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks. arXiv:1610.02136v3.

Hu, X.; Liu, W.; Bian, J.; and Pei, J. 2020. Measuring Model Complexity of Neural Networks with Curve Activation Functions. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.

Huang, Z.; Lv, C.; Xing, Y.; and Wu, J. 2021. Multi-Modal Sensor Fusion-Based Deep Neural Network for End-to-End Autonomous Driving With Scene Understanding. *IEEE Sensors Journal*. 21 (10): 11781–11790. doi.org/10.1109/jsen.2020.3003121.

Ji, W.; Ren, Z.; and Law, C. 2019. Uncertainty Propagation in Deep Neural Network Using Active Subspace. arXiv:1903.03989.

Jia, X. Y.; Jiang, C.; Fu, C. M.; Ni, B. Y.; Wang, C. S.; and Ping, M. H. 2019. Uncertainty Propagation Analysis by an Extended Sparse Grid Technique. *Frontiers of Mechanical Engineering*. 14 (1): 33–46.

Jin, J.; Dundar, A.; and Culurciello, E. 2015. Robust Convolutional Neural Networks under Adversarial Noise. arXiv:1511.06306v2.

Julier, S. J. and Uhlmann, J. K. 1997. New Extension of the Kalman Filter to Nonlinear Systems. In Signal Processing, Sensor Fusion, and Target Recognition VI: 182–193.

Julier, S. J. and Uhlmann, J. K. 2004. Unscented Filtering and Nonlinear Estimation. *Proceedings of the IEEE*. 92 (3): 401–422. doi.org/10.1109/jproc.2003.823141.

Kendall, A. and Gal, Y. 2017. What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision? In Proceedings of the 31st Conference on Neural Information Processing Systems: 5574–5584.

Liao, X.; Zhou, T.; Zhang, L.; Hu, X.; and Peng, Y. 2023. A Method for Calculating the Derivative of Activation Functions Based on Piecewise Linear Approximation. *Electronics*. 12 (2): 267. doi.org/10.3390/electronics12020267.

Nguyen, A.; Yosinski, J.; and Clune, J. 2015. Deep Neural Networks are Easily Fooled: High Confidence Predictions for Unrecognizable Images. In 2015 IEEE Conference on Computer Vision and Pattern Recognition.

Nunez, F.; Langarica, S.; Diaz, P.; Torres, M.; and Salas, J. C. 2020. Neural Network-Based Model Predictive Control of a Paste Thickener Over an Industrial Internet Platform. *IEEE Transactions on Industrial Informatics*. 16 (4): 2859–2867. doi.org/10.1109/tii.2019.2953275.

Pawlicki, M.; Kozik, R.; and Choraś, M. 2022. A Survey on Neural Networks for (Cyber-) Security and (Cyber-) Security of Neural Networks. *Neurocomputing*. 500: 1075–1087. doi.org/10.1016/j.neucom.2022.06.002.

Roy, A.; Conjeti, S.; Navab, N.; and Wachinger, C. 2019. Bayesian QuickNAT: Model Uncertainty in Deep Whole-Brain Segmentation for Structure-Wise Quality Control. *NeuroImage*. 195: 11–22.

Sattelberg, B.; Cavalieri, R.; Kirby, M.; Peterson, C.; and Beveridge, R. 2020. Locally Linear Attributes of ReLU Neural Networks. arXiv:2012.01940.

Smieja, M.; Struski, Ł.; Tabor, J.; Zieliński, B.; and Spurek, P. 2018. Processing of Missing Data by Neural Networks. arXiv:1805.07405v3.

Takenaka, K.; Ohtsuka, K.; Fujii, T.; Negi, M.; Suzuki, K.; Shimizu, H.; Oshima, S.; Akiyama, S.; Motobayashi, M.; Nagahori, M.; Saito, E.; Matsuoka, K.; and Watanabe, M. 2020. Development and Validation of a Deep Neural Network for Accurate Evaluation of Endoscopic Images from Patients With Ulcerative Colitis. *Gastroenterology*. 158 (8): 2150–2157. doi.org/10.1053/j.gastro.2020.02.012.

Titensky, J. S.; Jananthan, H.; and Kepner, J. 2018. Uncertainty Propagation in Deep Neural Networks Using Extended Kalman Filtering. In 2018 IEEE MIT Undergraduate Research Technology Conference: 1–4.

Truong, D. 2021. Estimating the Impact of COVID-19 on Air Travel in the Medium and Long Term Using Neural Network and Monte Carlo Simulation. *Journal of Air Transport Management*. 96: 102126. doi.org/10.1016/j.jairtraman.2021.102126.

van Buuren, S. and Groothuis-Oudshoorn, K. 2011. MICE: Multivariate Imputation by Chained Equations in R. *Journal of Statistical Software*. 45 (3): 1–67. doi.org/10.18637/jss.v045.i03.

Vigneswaran, R. K.; Vinayakumar, R.; Soman, K. P.; and Poornachandran, P. 2018. Evaluating Shallow and Deep Neural Networks for Network Intrusion Detection Systems in Cyber Security. In 9th International Conference on Computing, Communication and Networking Technologies.

Wang, H.; Shi, X. j.; and Yeung, D.-Y. 2016. Natural-Parameter Networks: A Class of Probabilistic Neural Networks. In Proceedings of the 30th Conference on Neural Information Processing Systems: 118–126.

Wilson, A. and Izmailov, P. 2020. Bayesian Deep Learning and a Probabilistic Perspective of Generalization. In Proceedings of the 34th Conference on Neural Information Processing Systems: 4697–4708.

Wu, A.; Nowozin, S.; Meeds, E.; Turner, R. E.; Hernández-Lobato, J. M.; and Gaunt, A. L. 2018. Deterministic Variational Inference for Robust Bayesian Neural Networks. arXiv:1810.03958v2.

Yu, H.; Yang, L. T.; Zhang, Q.; Armstrong, D.; and Deen, M. J. 2021. Convolutional Neural Networks for Medical Image Analysis: State-of-the-Art, Comparisons, Improvement and Perspectives. *Neurocomputing*. 444: 92–110. doi.org/10.1016/j.neucom.2020.04.157.

Zhang, B. and Shin, Y. C. 2021. An Adaptive Gaussian Mixture Method for Nonlinear Uncertainty Propagation in Neural Networks. *Neurocomputing*. 458: 170–183. doi.org/10.1016/j.neucom.2021.06.007.

Zhang, W.; Li, X.; Ma, H.; Luo, Z.; and Li, X. 2021. Federated Learning for Machinery Fault Diagnosis with Dynamic Validation and Self-Supervision. *Knowledge-Based Systems*. 213: 106679. doi.org/10.1016/j.knosys.2020.106679.

7 Appendix

7.1 Appendix A: Propagation of Densities

In this section, some additional information about PPDs and their propagation through linear operators is provided. We show a simple example of a PD on a union of polytopes of different dimensions which does not admit a PDF with respect to the Lebesgue measure in the traditional sense but does admit a PPDF in our sense with respect to Lebesgue measures of different dimensions. Additionally, we give a Corollary to Theorem 1 which shows that the dimension of the output PD is bounded by that of the input PD.

Example 1 (Piecewise probability distribution). *First, we provide a simple 2-dimensional example of a PD given by a PPDF on a polytopic subdivision consisting of 1-dimensional and 2-dimensional polytopes. Consider the 1-dimensional polytopes* A_0 *and* A_1 *defined by the closed intervals* $[0, 0.5] \times [2]$ *and* $[0.5, 1] \times [2]$*, respectively, and the 2-dimensional polytope* A_2 *given by the rectangle* $[0, 0.5375] \times [0,1]$*. Let* $D = A_0 \cup A_1 \cup A_2$*. The functions* $p_0 \colon A_0 \to \mathbb{R}, (x_1, x_2) \to 3x_1$*,* $p_1 \colon A_1 \to \mathbb{R}, (x_1, x_2) \to 0.3x_1^2$ *and* $p_2 \colon A_2 \to \mathbb{R}, (x_1, x_2) \to 1$ *define a piecewise function* $P \colon \mathcal{B}(D) \to \mathbb{R}$ *as in Definition 4. Additionally, the function* P *is* σ*-additive and satisfies* $P(D) = P(A_0) + P(A_1) + P(A_2) = 1$*. Note the first two summands are given by integrals with respect to the 1-dimensional Lebesgue measure whereas the third summand is given by an integral with respect to the 2-dimensional Lebesgue measure. It follows that* P *defines a PPD and the set* (p_0, p_1, p_2) *is the associated PPDF.*

Corollary 1 (Dimensionality is bounded). *We define the dimensionality of a PPDF to be the dimension of its support, i.e., the dimension of the highest-dimensional polytope on which the PPDF is defined. In the situation of Theorem 1, let* $d \in \mathbb{N}, d \leq m$*, be the dimensionality of* p_X*. Then the dimensionality* d' *of* $p_Y = W_* p_X$ *is bounded by* d*, i.e.,* $d' \leq d$*.*

Proof. The support $supp(p_X) \subset \mathbb{R}^m$ *is a* d*-dimensional manifold (with* $d \leq m$*). If* W *is injective, the support* $supp(p_Y) \subset \mathbb{R}^n$ *again is a* d*-dimensional manifold. If* W *is non-injective (i.e.* $ker(W)$ *is a* k*-dimensional subspace of* $\mathbb{R}^m, k \geq 1$*), the dimensionality of* $supp(p_Y)$ *is* $d - i$

with $0 \leq i \leq k$*, depending on the number of basis vectors of the kernel that generate* $supp(p_X)$.

□

7.2 Appendix B: Edgewise Subdivision

In PLT, we use edgewise subdivision to obtain a fine-grained grid of simplices in the input space on which the input PD is approximated by a piecewise constant PPDF. We now elaborate on edgewise subdivision by first explaining the construction of the subdivision and then discussing some useful properties.

Definition 5 (Corners and dimensions of a polytope). *Let* $m \geq 0$ *be an integer and* $A \subset \mathbb{R}^m$ *a polytope. There is a unique minimal set of points* $C(A)$ *in* $\mathbb{R}^m$ *such that A is the convex hull of these points. We call an element of* $C(A)$ *a corner of A. The dimension* $dim(A)$ *is given by the dimension of the affine hyperplane spanned by* $C(A)$.

We call

$$h(A) = \max_{a_i, a_j \in C(A)} |a_i - a_j|$$

the diameter of A and set

$$\rho(A) = 2 \sup\{R > 0 \colon B_R(x_0) \subset A, x_0 \in A\}$$

as the diameter of the biggest inner ball contained in A.

We now follow Edelsbrunner & Grayson (2000) to construct the edgewise subdivision of a simplex. Let A be a simplex of dimension d. For a natural number $k \in \mathbb{N}$, each 1-dimensional facet between exactly two corners of the simplex is divided into $k \in \mathbb{N}$ pieces of equal length creating a grid of points. The edgewise subdivision of A comprises the k^d (sub-)simplices spanned by adjacent (in direction of the vectors spanning the simplex) points of the grid. The edgewise subdivision for $k = 2$ on a 2-simplex and a 3-simplex is illustrated in Figure 1.

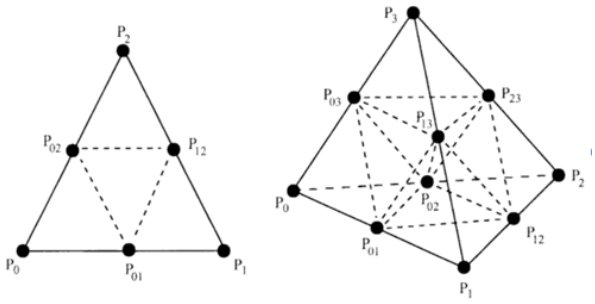


Figure 1: Edgewise Subdivision of a 2-simplex (left) and a 3-simplex (right) for $k = 2$ *(Edelsbrunner & Grayson, 2000)*

The subdivision of A obtained by this method is denoted by $\mathbb{A}^k_{edge}(A)$. It follows immediately from the definition of edgewise subdivision that for sufficiently large $k \in \mathbb{N}$ the distance between any two corners of any simplex in $\mathbb{A}^k_{edge}(A)$ becomes arbitrarily small. Thus, for every $\varepsilon > 0$

there exists a $k_\varepsilon \in \mathbb{N}$ such that $d(c_1, c_2) < \varepsilon$ for all $c_1, c_2 \in C(A_i)$ and every simplex $A_i \in \mathbb{A}_{edge}^{k_\varepsilon}(A)$. The minimal choice of $k_\varepsilon \in \mathbb{N}$ depends on A.

Lemma 1. *Let $k, l \in \mathbb{N}$. Then edgewise subdivision with respect to kl is equivalent to first subdividing with respect to l and then further subdividing each simplex in $\mathbb{A}_{edge}^{l}(A)$ with respect to k, i.e.,*

$$\mathbb{A}_{edge}^{kl}(A) = \mathbb{A}_{edge}^{k}(\,\mathbb{A}_{edge}^{l}(A)).$$

Proof. Edelsbrunner & Grayson (2000). □

The simplices in $\mathbb{A}_{edge}^{k}(A)$ can be assigned to congruence classes, i.e., two simplices belong to the same congruence class if and only if they are congruent. We denote the set of congruence classes of $\mathbb{A}_{edge}^{k}(A)$ by $Cong\left(\mathbb{A}_{edge}^{k}(A)\right)$. It can be shown that there exists an upper bound for the number of congruence classes which does not depend on k:

Lemma 2. *For any simplex A of dimension d, the set $\mathbb{A}_{edge}^{k}(A)$ has at most $d!/2$ congruence classes.*

Proof. Edelsbrunner & Grayson (2000). □

7.3 Appendix C: Error Estimation

In this subsection we study the error made when approximating the PPDF p with a piecewise polynomial density p' of degree k. In particular, choosing constant polynomials yields an error estimation for the method described in the main chapter. We measure the error by means of p-norms and the Hellinger distance, which we will define in the following. We will then provide upper bounds for the approximation error both in the input (Lemma 3) and output space (Theorem 4) and show that if the input subdivision and the output bin grid are fine enough, our method PLT is able to approximate the true propagated PD arbitrarily closely (Theorem 5).

Definition 6 (p-Norm). *Let $(\Omega, \mathcal{A}, \mu)$ be a measure space and $f: \Omega \to \mathbb{R}$ a measurable function. The p-norm of f is defined as*

$$\|f\|_p = \|f\|_{L^p(\Omega)} = \left(\int_\Omega |f|^p \, d\mu\right)^{\frac{1}{p}}$$

for $p \in [1, \infty)$ and

$$\|f\|_\infty = \|f\|_{L^\infty(\Omega)} = ess\ sup_{x \in \Omega} |f(x)|$$

for $p = \infty$. We say $f \in L^p(\Omega)$ if $\|f\|_p < \infty$.

In our case, we always consider a measure space $\Omega \subset \mathbb{R}^m$ together with the Borel σ-Algebra $\mathcal{A} = \mathcal{B}(\Omega)$ and the Lebesgue measure μ .

Definition 7 (σ-finite measure). *Let $(\Omega, \mathcal{A}, \mu)$ be a measure space. The measure μ is called σ-finite if Ω can be covered by countably many measurable sets $B_1, B_2, \ldots \in \mathcal{A}$ such that $\mu(B_i) < \infty$ for all i.*

Definition 8 (Absolutely continuous measure). *Let μ_1, μ_2 be two measures on a measureable space X. The measure μ_1 is absolutely continuous with respect to μ_2 if every measurable set $B \subset X$ with $\mu_2(B) = 0$ also satisfies $\mu_1(B) = 0$.*

Definition 9 (Hellinger distance). *Let P_1, P_2 be PDs on a measurable space X. Choose a σ-finite measure μ with respect to which P_1 and P_2 are absolutely continuous. Note that such a measure always exists (for example, $\mu = P_1 + P_2$ is a possible choice). Then by the Radon-Nikodym Theorem P_1 and P_2 have PDFs p_1 and p_2, respectively, on X with respect to μ. The Hellinger distance between P_1 and P_2 is defined as*

$$H(P_1, P_2) = \sqrt{\frac{1}{2}\int_X \left(\sqrt{p_1} - \sqrt{p_2}\right)^2 \mathrm{d}\mu}.$$

By Pollard (2001), $H(P_1, P_2)$ does not depend on the choice of μ. Hence, the Hellinger distance is well-defined. The Hellinger distance defines a metric on the set of measures on the measurable space X. The maximum value of the Hellinger distance between P_1 and P_2 is 1, which is attained if and only if P_1 and P_2 are mutually singular (Pollard 2001), i.e., there exists a measurable subset S of X such that $P_1(S) = 0 = P_2(X \backslash S)$.

For a symmetric semi-positive definite (s.s.p.d.) $n \times n$-matrix Σ, we set $supp(\Sigma) := \mathrm{Im}(\Sigma)$. This is the unique linear subspace V of $\mathbb{R}^n$ such that the normal distribution $N(0, \Sigma)$ admits a PDF with respect to the Lebesgue measure on V.

To denote the Hellinger distance between normal distributions, we will use the abbreviation $H(\Sigma_1, \Sigma_2) := H(N(0, \Sigma_1), N(0, \Sigma_2))$.

We will now provide error estimations for our approximation of the true PPDF. While we only use a constant approximation of the PPDF on each polytope in $\mathbb{A}_{edge}(A_i)$, we give an upper bound for the approximation error in a more general case where for each polytope, the PPDF values on $C(A)$ are interpolated by a polynomial of degree k such that for any polynomial PPDF of degree $\leq k$ the interpolation polynomial agrees with the PPDF. Note that our constant approximation fulfills this condition for $k = 0$. We start by estimating the approximation error of the polynomial approximation by means of the 2-norm. We consider the case where the PPDF p on a polytope A_i is $(k+1)$-times differentiable in a weak sense and its weak derivatives have finite 2-norm. In particular, every function p for which the first $k+1$ (classic) derivatives exist and have finite 2-norm, is an element of $W^{k+1,2}(A_i)$.

Lemma 3 (Approximation error, 2-norm). *Let $p \in W^{k+1,2}(A_i)$ (i.e., p is $(k+1)$-times differentiable in a weak sense and the weak derivatives have finite 2-norm). Let p' be the polynomial*

approximation of degree k on an edgewise subdivision $\mathbb{A}^{\varepsilon}_{i,edge}(A_i)$ of A_i. Then there exist $C_0, C_1 > 0$ such that

$$\|p - p'\|_{L^2(A_i)} \leq C_0 h^{k+1} \|D^{k+1} p\|_{L^2(A_i)},$$
$$\|\nabla p - \nabla p'\|_{L^2(A_i)} \leq C_1 \sigma h^{k} \|D^{k+1} p\|_{L^2(A_i).}$$

where

$$h = \max_{A \in \mathbb{A}^{\varepsilon}_{i,edge}} h(A),$$

$$\sigma = \max_{A \in \mathbb{A}^{\varepsilon}_{i,edge}} \frac{h(A)}{\rho(A)} \text{ and}$$

$$\|D^{k+1} p\|_{L^2(A_i)} := \left(\int_{A_i} \sum_{|\alpha|=k+1} |D^{\alpha} p|^2 dx \right)^{\frac{1}{2}}$$

where $\alpha \in \mathbb{N}_o^m$ is the associated multi-index vector with $\sum_i \alpha_i = k + 1$. The constants C_0 and C_1 depend on both k and A_i.

Moreover, we have

$$\lim_{\varepsilon \to 0} \|p - p'\|_{L^2(A_i)} = 0,$$
$$\lim_{\varepsilon \to 0} \|\nabla p - \nabla p'\|_{L^2(A_i)} = 0$$

Note that p' is dependent on $\varepsilon > 0$ as it is defined based on the subdivision $\mathbb{A}^{\varepsilon}_{i,edge}$.

Proof. The first two inequalities are derived in Ciarlet (2002). To show the claims about the limit of the errors for $\varepsilon \to 0$, it suffices to show that σ and $\|D^{k+1} p\|_{L^2(A_i)}$ are bounded and that $h \leq \varepsilon$. For two congruent simplices $A_i, A_j \in c$ (where c denotes the associated congruence class) it is clear that

$$h(A_i)/\rho(A_i) = h(A_j)/\rho(A_j) =: \sigma_c$$

holds and σ_c is finite. By Lemma 2 $\mathbb{A}^{\varepsilon}_{i,edge}$ comprises at most $d!/2$ congruence classes. Hence, $\sigma = \max_{c} \sigma_c < \infty$ can be chosen as the maximum of the finite set $\{\sigma_c \mid c \in Cong(\mathbb{A}^{\varepsilon}_{i,edge})\}$ and is independent of ε. Moreover, $\|D^{k+1} p\|_{L^2(A_i)} < \infty$ holds by definition as $p \in W^{k+1,2}(A_i)$. It follows immediately from the construction of edgewise subdivision that for any $l \in \mathbb{N}$ we have $h(A_{i,j}) \leq 1/l \cdot h(A_i)$ for all simplices $A_{i,j} \in \mathbb{A}^{l}_{i,edge}(A_i)$. By definition of $\mathbb{A}^{\varepsilon}_{i,edge}$, $l \in \mathbb{N}$ is chosen large enough such that $h(A_{i,j}) \leq \varepsilon$ holds for all $A_{i,j} \in \mathbb{A}^{l}_{i,edge}(A_i)$ which yields the claim. □

The constants $\|D^{k+1} p\|_{L^2(A_i)}$ and σ can be interpreted as follows: If the $(k+1)$-th derivatives of p attain large values or variations on major parts of A_i, it is difficult to approximate p in these regions by polynomial interpolation. In this case, the value of $\|D^{k+1} p\|_{L^2(A_i)}$ (and hence, the right-hand side of the error estimation) is rather large. The value $\sigma = \max_{A \in \mathbb{A}^{\varepsilon}_{i,edge}} \frac{h(A)}{\rho(A)}$ is a measure

of how geometrically "flat" the simplices in $\mathbb{A}^{\varepsilon}_{i,edge}$ are. For a flat simplex the longest 1-dimensional facet of the simplex is relatively large when compared to the diameter of its largest inner circle. Thus, a polynomial interpolation of a function defined on this simplex is prone to error due to the relatively large distances of the points used for interpolation. A large value of σ accounts for this condition.

The approximation error on $\mathbb{A}^{\varepsilon}_{i,edge}(A_i)$ with respect to the 2-norm is of order $O(\varepsilon^{k+1})$ for $\|p - p'\|_{L^2(A_i)}$ and $O(\varepsilon^k)$ for $\|\nabla p - \nabla p'\|_{L^2(A_i)}$, respectively. Thus, for a higher-degree interpolation, the approximation error in the input space decreases faster. However, for any degree $k > 0$, the propagation of p' through a NN can no longer be done in an exact manner as it involves calculating integrals of p' over polytopes. Therefore, we only use constant approximations of the true PDF for our method PLT.

Theorem 4 (Error propagation, p-norm*). Let p_1, p_2 be PDFs on a polytope A, let $p \in [1, \infty)$ and $f: A \to \mathbb{R}^m$ a linear function on A. Then there exists $C > 0$ such that*

$$\|f_*(p_1) - f_*(p_2)\|_{L_p}^p \leq C \cdot \|p_1 - p_2\|_{L^p}^p$$

Proof. Let $\tilde{f}$ be the restriction of f to $\ker f^{\perp}$and $c \coloneqq |\det(\tilde{f}^{-1})|$. Using Hölder inequality and triangle inequality for integrals, we get

$$\begin{aligned}
&\|f_* p_1 - f_* p_2\|_{L_p}^p \\
&\quad = \int_{f(A)} \left| \int_{f^{-1}(x)\cap A} c \cdot p_1(y)\mathrm{d}y - \int_{f^{-1}(x)\cap A} c \cdot p_2(y)\mathrm{d}y \right|^p \mathrm{d}x \\
&\quad = \int_{f(A)} \left| \int_{f^{-1}(x)\cap A} c \cdot \big(p_1(y) - p_2(y)\big)\mathrm{d}y \right|^p \mathrm{d}x \\
&\quad \leq C \int_{f(A)} \int_{f^{-1}(x)\cap A} |p_1(y) - p_2(y)|^p \,\mathrm{d}y\,\mathrm{d}x \\
&\quad = C \int_A |p_1(z) - p_2(z)|^p \,\mathrm{d}z = C\|p_1 - p_2\|_{L^p}^p
\end{aligned}$$

for some constant C which depends on A and $\tilde{f}$. □

Finally, we prove that our propagated piecewise constant PPDF given by formula (6) can approximate the true output PD P_y arbitrarily closely if the bin grid in the output space and the edgewise subdivision of the input polytopes are chosen fine enough.

Theorem 5. *Let P be a PPD in the input space of a neural network f. Then the output distribution obtained by PLT can approximate the propagated PPD $P_y = f_* P$ up to an arbitrarily small error.*

Proof. We first claim that for a given bin grid S in the output space, our method can approximate the true probability mass of P_y in each bin $b \in S$ up to an arbitrarily small error. Consider a subdivision $\mathbb{A}_{edge}$ of the input space D such that f is linear and the input PD is approximated by a piecewise constant PD P' given by a constant PDF $c_i \in \mathbb{R}$ on each $A_i \in \mathbb{A}_{edge}$ as defined

in Eq. 3. We denote the pushforward of P' with respect to f by $P'_y = f_* P'$. Let $b \in S$ and $\varepsilon > 0$. The entire probability mass of P' in each A_i is assigned to the output bin containing the image $f(t_i)$ of the center t_i of A_i. We first show that the probability mass of $P_y{}'$ in B can be approximated up to an error ε by PLT. Denote the output PD obtained by PLT by P_{PLT}. For any $A_i \in \mathbb{A}_{edge}$, the fraction of the probability mass $P'(A_i)$ that is propagated into B is correct if $f(A_i) \subset B$ or $f(A_i) \cap B = \emptyset$. If $f(A_i)$ is only partially contained in B, then a fraction of the probability mass $P(A_i)$ is either incorrectly propagated into B (if $f(t_i) \in B$) or incorrectly not propagated into B (if $f(t_i) \notin B$). By the construction of the edgewise subdivision, the total volume (and, therefore, the total probability mass with respect to P') of all polytopes A_i with images $f(A_i)$ not contained in a single bin becomes arbitrarily small if the subdivision is chosen fine enough. In particular, there is a $K \in \mathbb{N}$ such that $\left|P'_y(B) - P_{PLT}(B)\right| < \varepsilon/2$ for all $B \in S$ and all $k \geq K$ if $\mathbb{A}^k_{edge}$ is chosen as the input subdivision. Furthermore, the difference between P and P' becomes arbitrarily small for fine enough subdivision by Lemma 3. Hence, the difference between their respective pushforwards also becomes arbitrarily small. Therefore, there exists a $K' \in \mathbb{N}$ such that $\left|P_y(B) - P'_y(B)\right| < \varepsilon/2$ for all $b \in S$ and for all $k \geq K'$ if $\mathbb{A}^k_{edge}$ is chosen as input subdivision. It follows immediately that $\left|P_y(B) - P_{PLT}(B)\right| < \varepsilon$ for all bins $B \in S$ if k is chosen large enough which proves our first claim.

It is well-known that the true output distribution can be approximated arbitrarily closely by a piecewise constant PD if the underlying bin grid S is chosen fine-granular enough. Together with the first claim, this yields that for fine enough bin grid and input subdivision, the piecewise constant output PD obtained by PLT can approximate the true output PPDF up to an arbitrarily small error. □

Thus, we have shown that for increasingly fine-grained subdivision in the input and bin grid in the output, the result of PLT converges to the true propagated distribution.

7.4 Appendix D: Dependencies of Neurons

In this section, we will show that in general neurons in the same layer of a NN are dependent and that the assumption of independence between neurons can lead to large errors in the propagated PD. To achieve this, we consider a setting where the input neurons are given by independent Gaussian distributions, i.e., the input PD is a Gaussian with diagonal covariance matrix. We proof that even in this case, the neurons in the first hidden layer are only independent if the weight matrix satisfies specific requirements (Lemma 5). We then analyze the difference between the true distribution in a NN layer and the distribution resulting from assuming independent neurons, measured by the Hellinger distance introduced in Appendix C. We give a maximality criterion to determine when the Hellinger distance between two normal distributions assumes the maximal value of 1 (Lemma 6). We then use this maximality criterion to proof theorems yielding a wide range of examples of covariance matrices and weight matrices where the assumption of independent neurons in the hidden layer results in a maximal Hellinger distance of 1 between the true PD in the hidden layer and the PD obtained by assuming independence

(Theorems 7 and 8). Finally, we show some formulas for the Hellinger distance between a Gaussian distribution with non-diagonal covariance matrix and the Gaussian distribution resulting from restricting the covariance matrix to its diagonal (Theorem 9). These allow quantifying the error that would be made even in the input space if the input neurons are dependent and normally distributed but are modelled as independent. The formula depends on a set of eigenvalues, and it can be seen that it may result in large errors, measured by the Hellinger distance, as well.

Let k be a positive integer, μ be an element of $\mathbb{R}^k$ and Σ be a symmetric positive definite (s.p.d.) $k \times k$-matrix. We denote the PDF of the Gaussian distribution with mean μ and covariance matrix Σ by $N(\mu, \Sigma)$ and the identity matrix of dimension n by E_n.

Lemma 4 (Propagation of Gaussian distributions). *Let W be a non-singular $k \times k$-matrix. Then $W_* N(\mu, \Sigma) = N(W\mu, W\Sigma W^T)$.*

Proof. By the bijective case of Theorem 1, we can directly compute

$$W_*\mathrm{N}(\mu, \Sigma)(y) = \mathrm{N}(\mu, \Sigma)(W^{-1}y)|W^{-1}|$$

$$= \frac{|W^{-1}|}{(\sqrt{(2\pi)^k|\Sigma|}} \exp\left(-\frac{1}{2}(W^{-1}y - \mu)^{\mathrm{T}}\Sigma^{-1}(W^{-1}y - \mu)\right)$$

$$= \frac{1}{|W|\sqrt{(2\pi)^k|\Sigma|}} \exp\left(-\frac{1}{2}\big(W^{-1}(y - W\mu)\big)^{\mathrm{T}}\Sigma^{-1}\big(W^{-1}(y - W\mu)\big)\right)$$

$$= \frac{1}{\sqrt{(2\pi)^k|W\Sigma W^{\mathrm{T}}|}} \exp\left(-\frac{1}{2}(y - W\mu)^{\mathrm{T}}W^{-\mathrm{T}}\Sigma^{-1}W^{-1}(y - W\mu)\right) = \mathrm{N}\big(W\mu, W\Sigma W^{\mathrm{T}}\big)$$

which proves the claim. □

Definition 10. *We call a $k \times k$-matrix W permutation diagonal if W is diagonal up to a permutation of its columns.*

Lemma 5. *Let W be a non-singular $k \times k$-matrix such that for all diagonal s.p.d. matrices Σ the matrix $W\Sigma W^T$ is diagonal. Then W is permutation diagonal.*

Proof. By replacing W with W^{T} we can assume that $W^{\mathrm{T}}\Sigma W$ is diagonal for each diagonal s.p.d. matrix Σ. Denote the columns of W by $(w_i)_{i=1,\dots,k}$. For any $i \neq j$, we obtain

$$0 = \big(W^{\mathrm{T}}\Sigma W\big)_{i,j} = e_i^{\mathrm{T}}W^{\mathrm{T}}\Sigma W e_j = w_i^{\mathrm{T}}\Sigma w_j.$$

In other words, w_i is orthogonal to Σw_j for all $i \neq j$. As W is non-singular and $\Sigma = E_k$ is a valid choice, $(w_i)_i$ is an orthogonal basis. Now fix an index j. Because Σw_j is orthogonal to w_i for all $i \neq j$, it follows that Σw_j is a multiple of w_j. Since this holds for all diagonal s.p.d. matrices Σ, we can conclude that the vector w_j has a single non-zero entry. Since W is non-singular, this proves the claim. □

Theorem 6. *We denote by pr_i the projection to the i-th component. For a non-singular $k \times k$-matrix W the following are equivalent:*

1. *For each probability measure μ on $\mathbb{R}^k$ such that the family of random variables $(pr_i)_{i=1,\dots,k}$ is independent, $(pr_i W)_{i=1,\dots,k}$ is independent as well.*
2. *The matrix W is permutation-diagonal.*

Proof. We first show that 2 implies 1. Let W be permutation-diagonal. Then $(\mathrm{pr}_i W)_i = (a_i \mathrm{pr}_{\sigma(i)})_i$ for a permutation σ and a_i in $\mathbb{R}$. The independence of $(\mathrm{pr_i})_{i=1,\dots,k}$ implies the independence of $(a_i \mathrm{pr}_{\sigma(i)})_{i=1,\dots,k}$. To prove the converse, specializing condition 1 to non-degenerate normal distributions and applying Lemma 4 yields that for each s.p.d. diagonal matrix Σ the matrix $W\Sigma W^{\mathrm{T}}$ is diagonal. By Lemma 5, this implies that W is permutation-diagonal. □

Lemma 6 (Maximality criterion). *Let Σ_1, Σ_2 be s.s.p.d. $n \times n$ matrices. Then $H(\Sigma_1, \Sigma_2) = 1$ if and only if $supp(\Sigma_1) \neq supp(\Sigma_2)$.*

Proof. As stated in Appendix C, $H(\Sigma_1, \Sigma_2) = 1$ if and only if $N(0, \Sigma_1)$ and $N(0, \Sigma_2)$ are mutually singular. This is equivalent to $supp(\Sigma_1) \neq supp(\Sigma_2)$. □

For the following discussion, we fix a $k \times n$ matrix W and a s.s.p.d. matrix Σ. The matrix Σ defines a scalar product $\langle \cdot, \cdot \rangle_\Sigma$ on $\mathbb{R}^n$ by $\langle a, b \rangle_\Sigma = a^T \Sigma b$ for $a, b \in \mathbb{R}^n$. We denote by w^i the i-th row of W viewed as a column vector. Further let $(e_i)_{i=1,\dots,k}$ denote the standard basis of $\mathbb{R}^k$. For any matrix M, we denote by M_d the diagonal matrix containing only the diagonal entries of M, i.e., $(M_d)_{ij} = M_{ij}$ for $i = j$ and $M_{ij} = 0$ for $i \neq j$.

Lemma 7. *The Hellinger distance $H(W\Sigma W^T, (W\Sigma W^T)_d)$ is maximal if and only if*

$$supp(W\Sigma W^T) \neq \langle e_i | \langle w^i, \Sigma w^i \rangle \neq 0 \rangle.$$

Proof. By the maximality criterion, it suffices to show $\langle e_i | \langle w^i, \Sigma w^i \rangle \neq 0 \rangle = supp((W\Sigma W^T)_d)$. The support of $(W\Sigma W^T)_d$ is given by $\langle e_i | ((W\Sigma W^T)_d)_{ii} \neq 0 \rangle$. By definition the i-th diagonal entry of $(W\Sigma W^T)_d$ is given by $\langle w^i, \Sigma w^i \rangle$ and the claim follows immediately. □

Theorem 7. *If W is not surjective and does not have a zero row and Σ is non-singular, then*

$$H(W\Sigma W^T, (W\Sigma W^T)_d) = 1.$$

Proof. Since W is not surjective, we have $\dim supp(W\Sigma W^T) \neq k$. As each row of W is non-zero the dimension of $\left\langle e_i \middle| \langle w^i, w^i \rangle_\Sigma \neq 0 \right\rangle = \mathbb{R}^k$ is k. Thus, it follows from Lemma 7 that $H(W\Sigma W^T, (W\Sigma W^T)_d) = 1$. □

For s in $\{1, \dots, n\}$, we denote by D_s the $n \times n$ matrix satisfying $(D_s)_{s,s} = 1$ where every other matrix entry is 0.

Lemma 8. *If the s-th column of W has 2 non-zero entries, then $H(WD_sW^T, (WD_sW^T)_d) = 1$.*

Proof. Since D_s has rank 1, it follows that WD_sW^T has rank at most 1. In other words, $supp(WD_sW^T)$ has dimension at most 1. We can compute

$$(WD_sW^T)_d = diag\left(W_{1,s}^2, \dots, W_{k,s}^2\right).$$

Therefore, the dimension of $supp((WD_sW^T)_d)$ is given by the number of non-zero entries of the s-th column of the matrix W, which is greater than 1. This implies $supp(WD_sW^T) \neq supp((WD_sW^T)_d)$. □

Using the maximality criterion from Lemma 6, we characterize the matrices W for which there exists some s.s.p.d. matrix Σ with the property that $H(W\Sigma W^T, (W\Sigma W^T)_d) = 1$.

Theorem 8. *The following are equivalent:*

1. *There exists a diagonal s.s.p.d. matrix Σ such that* $H(W\Sigma W^T, (W\Sigma W^T)_d) = 1$.
2. *The matrix* W *has a column with at least 2 non-zero entries.*

Proof. The fact that 2 implies 1 follows from Lemma 8 as the matrix D_s is s.s.p.d for all $s = 1, \dots, n$. We show that 2 implies 1 via contradiction. Assume that W does not have a column with at least 2 non-zero entries and let Σ be a diagonal s.s.p.d matrix. Then a direct computation shows $W\Sigma W^T = (W\Sigma W^T)_d$, which implies $H(W\Sigma W^T, (W\Sigma W^T)_d) = 0 \neq 1$. □

From Theorems 7 and 8, we get a large set of diagonal s.s.p.d matrices Σ and matrices W for which the Hellinger distance $H(W\Sigma W^T, (W\Sigma W^T)_d)$ is maximal. In particular, this means that if the input distribution is given by a normal distribution with covariance matrix Σ and the weight matrix of the first layer of a NN is given by a matrix W such that W and Σ fulfill the conditions discussed in Theorem 7 or Theorem 8, the assumption of independent neurons in the first hidden layer of the neural network results in a distribution $N(0, (W\Sigma W^T)_d)$ with maximal Hellinger distance $H(W\Sigma W^T, (W\Sigma W^T)_d) = 1$ to the true distribution $N(0, W\Sigma W^T)$ in the first hidden layer. Note that a mean of 0 can be assumed here without loss of generality.

In the remainder of this section, we want to give a formula for the distance $H(W\Sigma W^T, (W\Sigma W^T)_d)$ – under appropriate assumptions on W and Σ – if it is not maximal. More precisely, we will consider the case $supp(W\Sigma W^T) = supp((W\Sigma W^T)_d) = \mathbb{R}^k$.

Lemma 9. *Let* Σ_1, Σ_2 *be s.p.d. matrices, then*

$$H^2(\Sigma_1, \Sigma_2) = 1 - \left(\frac{det(\Sigma_1)\, det(\Sigma_2)}{det\left(\frac{\Sigma_1 + \Sigma_2}{2}\right)^2} \right)^{\frac{1}{4}}.$$

Proof. Pardo (2005). □

For a s.p.d. matrix Σ, we define the correlation matrix associated to Σ by

$$\Sigma_{cor} := \sqrt{\Sigma_d}^{-1} \Sigma \sqrt{\Sigma_d}^{-1}.$$

Here, $\sqrt{\Sigma_d}$ is computed by applying the square root to each entry of Σ_d. Further, $\sqrt{\Sigma_d}^{-1}$ exists as each entry of Σ_d is positive. By the multiplicativity of the determinant, it follows that $H(\Sigma, \Sigma_d) = H(\Sigma_{cor}, 1)$.

Theorem 9. *Denote by $(\mu_i)_{i=1,\dots,n}$ the eigenvalues of Σ_{cor}. Then the equation*

$$H^2(\Sigma, \Sigma_d) = H^2(\Sigma_{cor}, E_n) = 1 - \left(\prod_{i=1}^{n} \frac{\mu_i}{\left(\frac{1+\mu_i}{2}\right)^2} \right)^{\frac{1}{4}}$$

holds.

Proof. We compute

$$H^2(\Sigma, \Sigma_d) = H^2(\Sigma_{cor}, E_n) = 1 - \frac{\det(\Sigma_{cor})^{\frac{1}{4}} \det(E_n)^{\frac{1}{4}}}{\det\left(\frac{\Sigma_{cor}+1}{2}\right)^{\frac{1}{2}}} = 1 - \left(\prod_{i=1}^{n} \frac{\mu_i}{\left(\frac{1+\mu_i}{2}\right)^2} \right)^{\frac{1}{4}}.$$

Here we used that the eigenvalues of $\frac{1}{2}(\Sigma_{cor} + E_n)$ are $\left(\frac{1}{2}(\mu_i + 1)\right)_i$. □

Note that if Σ is diagonal, Σ_{cor} is equal to E_n and the above formula yields a Hellinger distance of 0. Applying Theorem 9 to $W\Sigma W^T$ we obtain:

Corollary 2. *If W is surjective and Σ s.p.d, then*

$$H^2(W\Sigma W^T, (W\Sigma W^T)_d) = 1 - \left(\prod_{i=1}^{n} \frac{\sigma_i}{\left(\frac{1+\sigma_i}{2}\right)^2} \right)^{\frac{1}{4}},$$

where $(\sigma_i)_{i=1,\dots,k}$ denotes the family of eigenvalues of $(W\Sigma W^T)_{cor}$.

Proof. By assumption Σ is s.p.d., which means that $W\Sigma W^T$ is s.p.d. as well. Hence, we can apply Theorem 9 which proves the claim. □

Based on Theorem 9, we see that if the true input PD contains dependencies between neurons, the assumption of independence can lead to large errors depending on the eigenvalues of Σ_{cor}. Corollary 2 can be used to find more examples of covariance matrices and weight matrices where the assumption of independence in the propagated distribution can lead to large values of the Hellinger distance. In Theorems 7 and 8, we have already seen that even the maximal value of 1 can be attained depending on the weight matrix of the NN layer. Thus, we have shown that even under the restrictive assumption that the input neurons are independent and normally distributed, neglecting potential dependencies between neurons in subsequent layers generally results in large errors between the propagated PD and the true PD.

7.5 Appendix E: Datasets and Models

In this section, we provide additional information about the datasets on which our method was evaluated. The evaluation was performed on datasets widely used for classification and regression tasks which are shown in Table 2. All datasets are publicly available, and sources are provided.

Dataset	Features	Instances	Reference
Appendicitis	7	106	Wang, Zhang and Min 2019
Banana	2	5300	Jaichandaran 2023
Balance	4	625	Siegler 1976
Bands	35	541	Evans 1994
Boston	13	506	Harrison and Rubinfeld 1978
Breast	30	569	Wolberg 1990
California	8	20640	Kelley Pace and Barry 1997
Diabetes	10	442	Efron et al. 2004
Iris	4	150	Fisher 1936
Machine	9	209	Feldmesser 1987
Real Estate	7	414	Unknown 2018
Vehicle	18	847	Mowforth and Shepherd 1987
Wine	13	178	Aeberhard and Forani 1992

Table 2: Datasets

For each dataset, instances with naturally missing attribute values were excluded before deleting values and applying MICE imputation as described in the evaluation. We provide the code used for our experiments in a publicly available repository[3].

7.6 References

Aeberhard, S. and Forina, M. 1992. Wine. doi.org/10.24432/C5PC7J.

Ciarlet, P. G. 2002. The Finite Element Method for Elliptic Problems, Philadelphia, Pa.: Society for Industrial and Applied Mathematics. doi.org/10.1137/1.9780898719208.

Edelsbrunner, H. and Grayson, D. R. 2000. Edgewise Subdivision of a Simplex. Discrete and Computational Geometry. 24 (4): 707–719. doi.org/10.1007/s4540010063.

Efron, B.; Hastie, T.; Johnstone, I.; and Tibshirani, R. 2004. Least angle regression. The Annals of Statistics. 32 (2). doi.org/10.1214/009053604000000067.

Evans, B. 1994. Cylinder Bands. doi.org/10.24432/C50C7B

[3] https://github.com/URWI2/Piecewise-Linear-Transformation

Feldmesser, J. 1987. Computer Hardware. doi.org/10.24432/C5830D.

Fisher, R. A. 1936. Iris. doi.org/10.24432/C56C76.

Harrison, D. and Rubinfeld, D. L. 1978. Hedonic housing prices and the demand for clean air. Journal of Environmental Economics and Management. 5 (1): 81–102. doi.org/10.1016/0095-0696(78)90006-2.

Jaichandaran, S. Standard Classification (Banana Dataset). https://www.kaggle.com/datasets/saranchandar/standard-classification-banana-dataset?select=banana.csv. Accessed: 2023-14-08.

Kelley Pace, R. and Barry, R. 1997. Sparse spatial autoregressions. Statistics and Probability Letters. 33 (3): 291–297. doi.org/10.1016/S0167-7152(96)00140-X.

Mowforth, P. and Shepherd, B. Statlog (Vehicle Silhouettes). doi.org/10.24432/C5HG6N.

Pardo, L. 2006. Statistical Inference Based on Divergence Measures, London: CRC Press LLC.

Siegler, R. 1976. Balance Scale. doi.org/10.24432/C5488X.

Pollard, D. 2010. A User's Guide to Measure Theoretic Probability, Cambridge: Cambridge Univ. Press.

Unknown 2018. Real estate valuation data set. doi.org/10.24432/C5J30W.

Wang, M.; Zhang, Y.-Y.; and Min, F. 2019. Active Learning Through Multi-Standard Optimization. *IEEE Access*. 7: 56772–56784. doi.org/10.1109/ACCESS.2019.2914263.

Wolberg, W. 1990. Breast Cancer Wisconsin (Original). doi.org/10.24432/C5HP4Z.

2.4 Paper 4: The Currency of Wiki Articles – A Language Model-based Approach

Current Status	Full Citation
This paper is **accepted** and **published** in the *Proceedings of the 44th International Conference on Information Systems*	Heinrich, B., Huber, M. F., Krapf, T., & Schiller, A. (2023). The currency of wiki articles – A language model-based approach. In *Proceedings of the 44th International Conference on Information Systems*, Hyderabad, India. https://aisel.aisnet.org/icis2023/dab_sc/dab_sc/14

Abstract:

Wikis are ubiquitous in organizational and private use and provide a wealth of textual data. Maintaining the currency of this textual data is important and difficult, requiring large manual efforts. Previous approaches from literature provide valuable contributions for assessing the currency of structured data or whole wiki articles but are unsuitable for textual wiki data like single sentences. Thus, we propose a novel approach supporting the assessment and improvement of the currency of textual wiki data in an automated manner. Grounded on a theoretical model, our approach makes use of data retrieved from recently published news articles and a language model to determine the currency of fact-based wiki sentences and suggest possible updates. Our evaluation conducted on 543 sentences from six wiki domains shows that the approach yields promising results with accuracies over 80% and thus is well-suited to support assessment and improvement of the currency of textual wiki data.

Keywords: Data quality, currency, wikis, textual data, unstructured data, language model

Please note that the overall formatting of the paper has been adjusted to the remainder of the dissertation. Moreover, terms only common in British English have been converted to the corresponding American English terms.

The paper as published by AIS is available at: https://aisel.aisnet.org/icis2023/dab_sc/dab_sc/14

1 Introduction

Wikis provide an easy and efficient way to share, store, reuse, adapt, mobilize, and aggregate information (Beck et al. 2015; Yates et al. 2009). Their popularity, availability, and scope have increased over time, to the point where they are now widely applied across all areas and industries (Bhatti et al. 2018). Beyond organizational application, wikis have also become commonplace in private use (Alqahtani 2017). For instance, Wikipedia can not only be seen as the most comprehensive information repository in human history (Dang and Ignat 2017), but it also is one of the world's most popular websites, with close to 2 billion page visits per month (Wikimedia 2023). The English Wikipedia alone contains more than 6 million articles providing data about a variety of topics such as politics, science, culture, and sports as well as a large number of biographies (Wikipedia 2023).

Such a wealth of data, however, also comes with a drawback: It makes maintaining high data quality (DQ) in wikis, especially in a manual manner, very difficult and costly. This issue is further exacerbated by the fact that large parts of wiki articles consist of natural language text, which DQ maintenance methods have to cope with. DQ can be defined as "the measure of the agreement between the data views presented by an information system and that same data in the real world" (Orr 1998, p. 67; cf. also Heinrich et al. 2009). It is a multidimensional construct (Cichy and Rass 2019; Wang and Strong 1996) comprising several DQ dimensions such as currency, completeness, accuracy, and consistency (Nelson et al. 2005). In accordance with DQ literature (Heinrich and Klier 2015; Nelson et al. 2005; Zak and Even 2017), we refer to currency in this paper, expressing whether a data view presented by an information system, which was originally accurately captured, still is an accurate representation at the time of assessment. It is related to, but different from other time-related dimensions such as timeliness, which in DQ literature has mainly been defined as being directly dependent on the age of a data value and the maximum length of time the value of the considered attribute remains up-to-date (Ballou et al. 1998; Heinrich et al. 2018). Timeliness in this sense is not applicable to wikis, as there is no fixed and known maximum length of time for which certain data in a wiki can remain up-to-date (for a detailed discussion cf. Heinrich and Klier 2015, pp. 83-84). In contrast, currency is of particular importance for wikis as outdated data makes the wiki less helpful, less credible and leads to detrimental effects such as misinformation and less efficient use (Bhatti et al. 2018; He and Yang 2016; Shah et al. 2015). Moreover, there is a strong correlation between the currency of data contained within a wiki and the frequency of its use, as well as the satisfaction of its users (Bhatti et al. 2018). Thus, we focus on the currency of wikis in this paper. In the context of wikis, currency expresses whether a wiki article (and especially the textual wiki data) still describes the current state of the corresponding entity in the real world at the time of assessment (Klier et al. 2021; Lewoniewski 2019).

As already pointed out, wiki articles usually mainly consist of (unstructured) textual data, sometimes paired with data in structured form (e.g., information boxes) or other unstructured data (e.g., pictures or short videos). While for structured data, many approaches to assess and improve

currency have been proposed (e.g., Heinrich and Klier 2015; Zak and Even 2017), there are only a few initial contributions regarding the currency of textual data (e.g., Batini and Scannapieco 2016; Hao et al. 2020). Yet, textual data is the main component of wiki articles and frequently becomes outdated due to events in the real world (Klier et al. 2021). Due to the sheer number of articles and the velocity of required updates, keeping textual wiki data current is highly challenging. Indeed, it has been noted that even Wikipedia, a highly popular wiki with a very large number of voluntary authors, has difficulties keeping articles current (Hatcher-Gallop et al. 2009). Consequently, it is crucial to assess and improve the currency of textual wiki data with automated support (Seibert et al. 2011). Thus, the research question we aim to address in this paper is as follows:

How can the assessment and improvement of the currency of textual wiki data be supported in an automated manner?

The remainder of the paper is organized as follows. In the next section, we discuss the problem context and present a running example. Subsequently, we provide an overview of related prior work. We then develop a novel language model-based approach to assess and improve the currency of textual wiki data. Thereafter, we instantiate our approach using a dataset from Wikipedia and evaluate and discuss its performance. Finally, we conclude, reflect on limitations, and provide an outlook on future research.

2 Problem Context and Running Example

Textual wiki data usually contains a large amount of factual data which may become outdated over time due to changes in the real world. To illustrate this idea, we consider a sentence about the political position of the prime minister (PM) of the United Kingdom currently held by Rishi Sunak, and use it as a running example throughout this paper (any fact-based sentences from other Wikipedia domains can be used analogously). This is typical data which may be provided by Wikipedia or other wikis containing general information. Sunak's predecessor Liz Truss took office in early September 2022 (point in time t_1 in Figure 1), and this data may have been stored correctly in a wiki shortly after (point in time t_2 in Figure 1). At this time, the data provided by the wiki is current. However, in late October 2022, Sunak succeeded Truss as PM (point in time t_3 in Figure 1). This is a change in the real world, which renders the data previously stored in the wiki outdated. The data provided by the wiki remains outdated until it is correctly updated (point in time t_4 in Figure 1), which restores its currency. The same sequence takes place for all previous changes of office (e.g., May succeeding Cameron, Johnson succeeding May, etc.). Ideally, only a short amount of time elapses between the real-world change and the corresponding wiki update (e.g., between t_3 and t_4). However, there are cases where articles remain outdated for weeks before the respective data is (manually) updated.

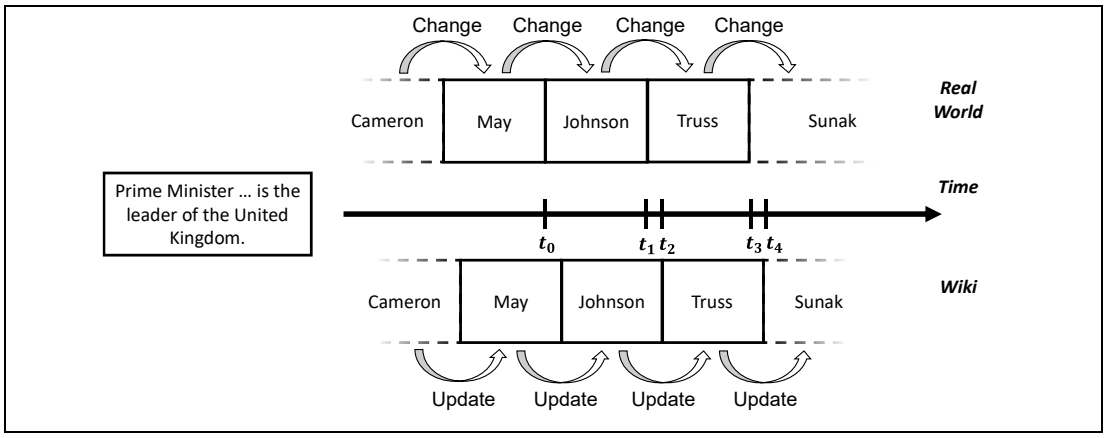


Figure 1. Illustration of Textual Wiki Data becoming Outdated

3 Related Work

The literature offers a large body of work for the assessment and improvement of currency for structured data (e.g., Heinrich and Klier 2015; Zak and Even 2017) as well as some initial contributions on unstructured data (e.g., Batini and Scannapieco 2016; Hao et al. 2020). Moreover, there are works discussing the assessment and improvement of DQ with a focus on wiki articles (e.g., Klier et al. 2021; Wang and Li 2020). This section is thus structured as follows. First, we briefly summarize existing work on currency in structured data and whether this work can be transferred to unstructured data. We then provide an overview of contributions on (textual) unstructured data. Subsequently, we review works focusing on the DQ of wiki articles in general, and works dealing with the currency of wiki articles in particular.

3.1 Currency of Structured Data

One of the earliest and most influential contributions to the assessment of time-related DQ dimensions of structured data was made by Ballou et al. (1998). They use parameters such as attribute value's age at the time of assessment, the (fixed and given) maximum lifespan of the attribute, and a sensitivity parameter to adjust the assessment based on the context. However, not all attributes have a predetermined lifespan, which led other authors to bring forward probability-based metrics (Heinrich et al. 2009; Heinrich and Klier 2011, 2015; Wechsler and Even 2012). Heinrich et al. (2009) propose a general procedure for developing probability-based metrics for the currency of structured data. Heinrich and Klier (2011) and Wechsler and Even (2012) both suggest metrics for the assessment of currency assuming an exponential distribution of attribute value lifespan. Heinrich and Klier (2015) relax this assumption, introduce additional metadata and provide a metric based on conditional probabilities, resulting in an interval-scaled and interpretable metric value. Taking up ideas from Heinrich and Hristova (2014) and Wechsler and Even (2012), Zak and Even (2017) suggest a continuous-time Markov chain model for detecting and handling currency declines. These approaches form important contributions and can be applied in an automated manner to assess the currency of structured data. Yet, being developed for structured data, they crucially rely on values of given, defined attributes (e.g., as contained in a relational database). Such attributes do not exist in the unstructured texts of wiki articles,

rendering the approaches unsuitable for the direct application to this kind of data. To alleviate this issue, Batini et al. (2011) suggest to pre-process textual data and then apply an approach for structured data. However, pre-processing textual data still does not yield standardised and homogenous attributes across large bodies of text data (e.g., wiki articles) as required for the application of these approaches. Moreover, even if attributes were somehow derived from the textual data, the attributes are then themselves erroneous, as they are based on the textual data with imperfect DQ. Thus, these approaches cannot be applied to assess the currency of (unstructured) textual data.

3.2 Currency of Unstructured Data

The currency of unstructured data is commonly assessed based on the most recent update time of the considered data (cf., e.g., Batini and Scannapieco 2016; Firmani et al. 2016; Shah et al. 2015). The most recent update time intuitively appears valuable in assessing currency. Additionally, it is usually automatically saved as metadata and is thus commonly available. Nevertheless, relying on the time elapsed since the last update is not sufficient to assess currency by far. For instance, a wiki article about the Russian war in Ukraine may already be outdated even though it was updated just days or hours ago, while another wiki article about the football world cup in 1990 may still be current despite its last update having occurred years ago. Hao et al. (2020) suggest an initial way to mitigate this issue by additionally considering average update frequencies. While these frequencies can be valuable in assessing currency, they have a drawback in that they can only be based on past data (i.e., previous updates). They cannot capture a change in pace of suddenly occurring events regarding a certain topic. Moreover, using the most recent update time, the average update frequency or other metadata does not offer suggestions for the improvement of outdated textual data, as the actual semantical content of the text is not considered at all.

3.3 DQ of Wiki Articles

The literature contains a large body of work on the DQ of wiki articles in general. These approaches generally strive to assess the DQ of whole articles (i.e., no indication of the DQ of, e.g., individual sentences, is provided). Several works aim to classify wiki articles into different quality categories, based on, for example, the length of the article or the reputation of its authors (cf., e.g., Blumenstock 2008; Shen et al. 2017; Wang and Li 2020; Wöhner et al. 2015; Zhang et al. 2020). Wikipedia employs a scoring system called ORES which assigns quality scores to single edits and full articles (MediaWiki 2023). These approaches have the ability to assess the quality of a wiki article based on its edit history and other factors. However, a shared issue among all these works is that they perceive DQ as a unidimensional construct, rather than a multidimensional one. Consequently, various aspects of DQ are compressed into a single rating, making it impossible to pinpoint specific quality problems (like outdated sentences). Therefore, capturing the currency of textual wiki data is out of the scope of these approaches.

3.4 Currency of Wiki Articles

To the best of our knowledge, despite the topic's undeniable relevance, only a handful of approaches dealing specifically with currency in wiki articles have been proposed. Stvilia et al. (2005) assess the currency of a wiki article based on the time it was last updated. Thus, this approach shares the drawback of the works for unstructured data presented above in that the time since the last update is not a sufficient indicator of currency. Moreover, such an approach does not offer suggestions for improvement of the data. Tran and Cao (2013) aim to find outdated data contained in the information boxes of Wikipedia articles by searching the Web for related data based on pattern recognition. This is an intriguing approach, but it is restricted to structured data contained in information boxes (which are not present in every wiki). In particular, it is not applicable to the textual content of wiki articles for the reasons outlined above. Lewoniewski (2017) proposes to improve the quality of Wikipedia articles by enriching the content of information boxes with data from information boxes of other language versions of the same article, taking into account DQ and popularity metrics. This approach may also be useful to address currency issues. However, it relies on multiple language versions of the same article being available. Moreover, it is limited to information boxes, and not applicable to the textual content of the article. Lewoniewski (2019) proposes to assess the currency of information boxes by examining the recent update frequencies, which is subject to the limitations discussed above. Finally, Klier et al. (2021) suggest an event-driven approach for the assessment of currency of wiki articles. They aim to detect events which affect the currency of wiki articles by monitoring their pageviews, with a sudden spike in pageviews indicating a currency-related event. This approach supports the automated detection of potentially outdated articles, but it does not indicate which part of the article has actually become outdated. Moreover, it does not offer suggestions for the improvement of currency.

3.5 Summary

In summary, while there are some important contributions for assessing and improving the currency of structured and unstructured data, they are not suited for textual wiki data. This is because they either rely on structured data attributes, which are not available in the context of textual wiki data, or only consider simple metadata such as most recent update time, but not the textual content itself. Further, valuable work has been conducted on the quality of wiki articles, but most of these works provide just a general quality assessment and only a few studies have discussed currency of wiki articles specifically. These proposed approaches have drawbacks that limit their usefulness, such as only being applicable to certain structured elements of an article. They do not consider the textual content of the wiki articles itself and thus cannot improve its currency. Therefore, to address this research gap, we propose an approach for the assessment and improvement of the currency of textual wiki data.

4 Language Model-based Approach for the Currency of Wiki Articles

In this section, we first introduce a theoretical model and our approach. We then discuss the assessment of the currency of a wiki sentence based on our approach.

4.1 Theoretical Model and Approach

In general, changes occurring over time in the real world determine the currency of textual wiki data. As usually not all these real-world changes are known, uncertainty arises. To account for these uncertainties, we model the temporal changes in the real world as a stochastic process. Therefore, let $(\Omega, \mathcal{F}, P)$ be a probability space and $\{X_t, t \in T\}$ a stochastic process that is defined on $(\Omega, \mathcal{F}, P)$. In our setting, the parameter set T reflects time and contains all considered points in time. For each point in time $t \in T$, the function $X_t: \omega \to X_t(\omega), \Omega \to S$ is a random variable (which is discrete or continuous) taking values in the data space S. In the case of certainty (i.e., all real-world changes regarding the data are known), $X_t(\omega)$ attains the single value $s \in S$ with probability 1. In reality however, certainty is not always given and $X_t(\omega)$ follows a (non-trivial) probability distribution.

In particular, X_{t_0} denotes the random variable describing real-world data at $t_0 = 0$, the time of creation of the data, where it attains one value $x_{t_0} \in S$ (under certainty). The real-world data then may change over time depending on $\{X_t, t \in T\}$, and at a certain point in time $t_a > t_0$ is stored in a wiki as the (certain) value $x_{t_a} \in S$. From then on, the real-world data may continue to change over time depending on $\{X_t, t \in T\}$, and at any later point in time $t_b > t_a$, the true value $x_{t_b} \in S$ may differ from the value $x_{t_a} \in S$ provided by the wiki. This stochastic process can also be used to reflect updates of the data in the wiki: If at a certain point in time t_c the data in the wiki is updated as the value $x_{t_c} \in S$, from then on, at any point in time $t_d > t_c$, the true value $x_{t_d} \in S$ may be different from the value $x_{t_c} \in S$ due to, once again, a change in the real world. For instance, in our running example, the stochastic process $\{X_t, t \in T\}$ begins at some point in time t_0, depending on the temporal horizon of interest (e.g., the start of Johnson's term). At a point in time t (between t_0 and t_1), this data is still unchanged (x_t) and stored in the wiki. At the point in time t_1, Truss becomes PM, and the true value $x_{t_1} \in S$ (Truss) is different from the value $x_t \in S$ (Johnson) provided by the wiki: The data provided by the wiki is outdated. It remains so until the point in time t_2, when the wiki is updated to reflect $x_{t_1} \in S$ (Truss), but then again becomes outdated when Sunak becomes PM.

In our model, at any point in time $t_z > t_a$ (the time at which the data was initially stored in the wiki), the data may be outdated. Today, manual effort would constantly be required to search and obtain the correct real-world value $x_{t_z} \in S$ under certainty. Using x_{t_z}, the data stored in the wiki can then either be current or has to be updated. However, our idea is to automatically assess the distribution of the random variable X_{t_z}, which may then be used to support the validation or

facilitate the update of the data stored in the wiki. In particular, if the distribution of X_{t_z} shows that the data stored in the wiki is potentially outdated (e.g., its probability falls below a threshold), the need for an update can be suggested, with probable current real-world values also suggested by X_{t_z}. Generally, the distribution of the random variable X_{t_z} could be assessed by the term

$$f_{X_{t_z}|\{X_t, t\in T, t<t_z\}}(x_{t_z}|\{x_t, t \in T, t < t_z\}) \quad (1)$$

where $f_{X_{t_z}|\{X_t, t\in T, t<t_z\}}$ is a function incorporating information about changes of real-world data over time (e.g., how often the PM of a country changes), $\{X_t, t \in T\}$ is the stochastic process modelling changes in the real world as introduced above, and $x_t \in S$ for all t. However, using formula (1) is rarely feasible in realistic situations, as this assessment requires the knowledge of all x_t (including for points in time $t < t_a$ where no data was stored in the wiki yet, and including for points in time very briefly before t_z). In fact, the only points in time at which x_t is known are the points in time at which the data was initially stored in the wiki, or the points in time at which it was updated previously. Denoting the set of all such points in time as $T_U \subset T$, one can then assess the distribution of X_{t_z} by the following term:

$$f_{X_{t_z}|\{X_t, t\in T_U\}}(x_{t_z}|\{x_t, t \in T_U\}) \quad (2)$$

Using this formula generally allows to support the validation or facilitate the update of the data stored in the wiki. However, it may not produce good results, in particular if the data is not updated frequently or the last update occurred a long time ago. Indeed, it may take a long time to detect changes that occurred in the real world and support an update of the data in the wiki. We thus propose to additionally use data y_t provided by recent external sources (e.g., news articles; details provided in the next section) at points in time $t \in T_E \subset T$, with T_E being defined as the subset of T, for which additional data are available. This data may serve as a proxy for x_t for points in time t close to t_z. The use of such data is crucial to detect real-world changes and to facilitate updates. The assessment of the distribution of X_{t_z} can then be conducted based on the following term:

$$f_{X_{t_z}|\{X_t, t\in T_U\},\{Y_t, t\in T_E\}}\left(x_{t_z}\middle|\{x_t, t \in T_U\}, \{y_t, t \in T_E\}\right) \quad (3)$$

To summarize, the approach in (3) allows taking into account

- known data stored in the wiki at the points in time $t \in T_U$
- data provided by external sources at the points in time $t \in T_E$
- a function incorporating information about changes of real-world data over time

to validate and potentially update data stored in a wiki at the point in time t_z.

With respect to a practical application, using the approach (3) to assess the distribution of X_{t_z} seems to require substantial manual effort for extracting the external data y_t and for conducting the assessment itself. However, language models (e.g., Brown et al. 2020; Devlin et al. 2019) can serve to address these problems and deliver a prediction for the distribution of X_{t_z} (cf. next section).

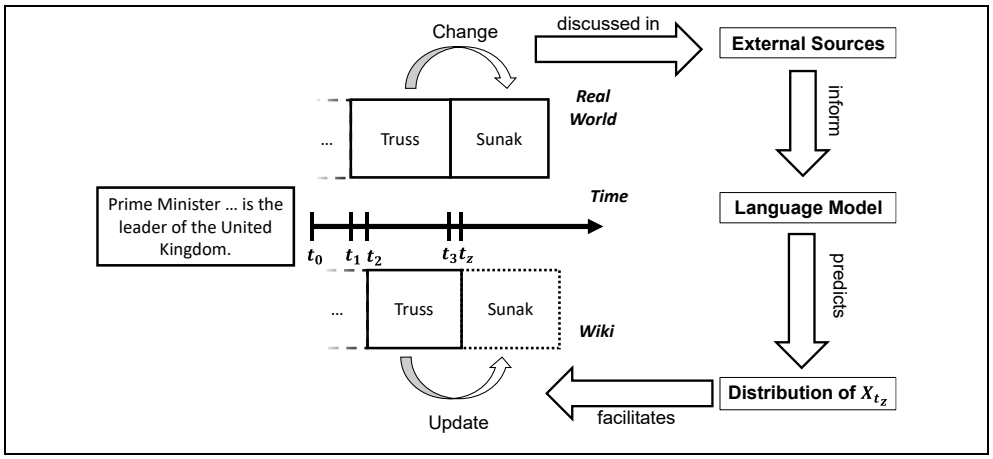


Figure 2. Overview of the Approach

Figure 2 illustrates our approach using the example of the latest change of the PM. The data that Truss is PM was stored in the wiki at the point in time t_2. At the point in time t_z (e.g., $t_z = 26/10/2022$; one day after the beginning of Sunak's tenure), the wiki still provides this data. Yet, Sunak has already taken office one day before t_z, at the point in time $t_3 = 25/10/2022$. The data provided by the wiki at t_z is outdated. However, Sunak's appointment as PM has been reported by external sources, for instance media outlets in their news articles, at times t_e with $t_3 \leq t_e \leq t_z$ (e.g., Ott 2022; BBC 2022). These news articles can be collected and processed in an automated manner. They support the prediction of the distribution of the random variable X_{t_z} by a language model according to formula (3). The language model's prediction of X_{t_z} suggests in our example that the PM is Sunak. In this way, the update of the wiki at the point in time t_z is facilitated. As the result of the update, Sunak is correctly provided as the PM by the wiki (dotted line in Figure 2).

4.2 Assessing the Currency of a Wiki Sentence

In this subsection, we elaborate on the assessment of the currency of a wiki sentence (called *focus sentence* in the following) based on the theoretical model and the approach introduced above. In particular, a focus sentence contains at least one *currency-related statement* which can either be current or outdated regarding the (real-world) fact it refers to. For example, the focus sentence "*Prime Minister Sunak is the leader of the United Kingdom*" contains, for example, the currency-related statements that **Sunak** is the current PM (and not, e.g., Truss), and that he is still alive ("**is** the leader"). In this sense, multiple different currency-related statements can be examined when assessing the currency of a focus sentence. Importantly, this example further illustrates that the currency of a currency-related statement crucially depends on (typically) one corresponding term ("is"/ "was", "Sunak"/ "Truss"/ "Johnson"/…), which we call the *currency-related term*. In our running example, we mainly consider the currency-related statement about the current UK PM for which the name of the PM is the currency-related term.

To assess and improve a currency-related statement of a focus sentence, we propose to traverse four steps: First, a) keywords are extracted from the focus sentence. Ideally, these keywords express the meaning of the focus sentence. Naturally, the currency-related term (in our example,

the name of the PM) is not used for the keyword extraction. Then, b) news articles from sources which contain the keywords are identified. Next, c) the news articles are divided into smaller paragraphs. Using semantic comparisons, the paragraphs potentially containing the sought update-relevant data are determined. Finally, d) the identified paragraphs are transferred to a language model and are used as input together with the focus sentence. If the paragraphs do indeed contain update-relevant data, the language model recognizes the semantic relationships between the paragraphs and the focus sentence resulting in an up-to-date prediction for the currency-related term. This prediction corresponds to the result of our approach in (3) introduced above. The four steps are shown in Figure 3 in the case of our running example of the PM and are described in more detail below.

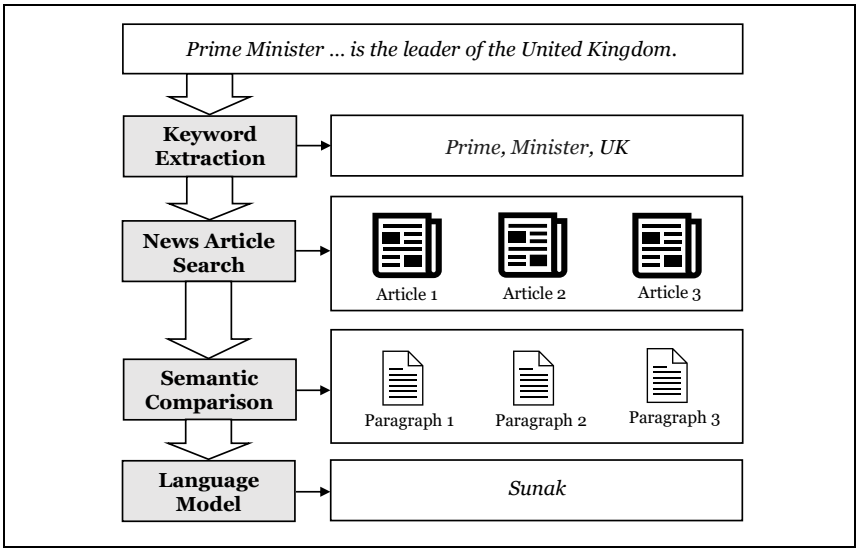


Figure 3. Assessing the Currency of a Wiki Sentence

Ad a): The starting point is the focus sentence, based on which keywords are extracted in the first step. For a valid result of the search for news articles, the keywords must contain the most important entities and terms mentioned in the focus sentence. For the keyword extraction, we propose to use the language model GPT-3.5 from OpenAI. This works well in a few-shot learning setting (Brown et al. 2020) in which a few exemplary sentences together with desirable keywords (e.g., "Djokovic currently holds … Australian Open titles" as sentence and {"Djokovic", "Australian", "Open", "titles"} as keywords) are provided. On this basis, GPT-3.5 can be instructed to return a number of keywords for a given focus sentence. In our running example, the keywords "Prime", "Minister" and "UK" are extracted.

Ad b): In the second step, news articles containing all extracted keywords are searched. For our example, the result of this search should thus be articles that mention the name of the PM. Various news sources can be used as a basis for this search, in particular, renowned news outlets such as the *New York Times* or *The Guardian*. Importantly, the choice of adequate sources is crucial for our approach as it determines the set of retrievable articles. For instance, using the *New York Times* as the only news source might not be helpful when assessing the currency of focus sentences from the domain of second division German football. Therefore, in the case of focus sentences from a specific domain, a news source which is known to cover this specific domain

should be employed. On the other hand, if the focus sentences stem from various domains, a broad mixture of different news outlets is preferable. We explicitly focus on news articles as opposed to making use of a language model-infused search engine such as Bing or Bing Chat for three reasons. First, news articles are usually reliable whereas the results of a general web search may not be. Second, we can thus focus on obtaining recently published data (e.g., by specifying a certain search period). Third, these search engines themselves often use Wikipedia as a source, making them unsuitable for assessing and improving the currency of Wikipedia articles.

To further ensure that only news articles containing update-relevant data are retrieved, a search period can be specified, limiting the maximum age of eligible articles. With the specification of a longer search period, the number of potentially found articles increases, however their currency might decrease since already outdated data could also be contained. This results in a trade-off between the number of articles found and their currency. In the case of our example, a search period which is chosen too high might also yield articles which still refer to Truss as PM.

Ad c): We proceed by dividing all articles found into smaller paragraphs. A paragraph is defined as an excerpt consisting of a certain number of consecutive sentences. Moreover, the division of an article into paragraphs can either be disjoint or allow paragraphs to overlap. Then, semantical text embeddings can be used to identify the paragraphs that are most similar to the focus sentence. More precisely, the text embedding converts each paragraph into a numerical vector of a certain dimension. After also embedding the focus sentence into the same vector space, vector similarities between each paragraph embedding and the focus sentence embedding can be computed. Our idea is that the paragraphs most similar to the focus sentence are also most likely to contain update-relevant data. In our running example, such a paragraph should consist of, for instance, three sentences of which at least one ideally mentions the current PM.

Ad d): Finally, the identified similar paragraphs and the focus sentence are passed to a language model. Based on these paragraphs, the language model predicts an answer for the currency-related term which is decisive for the currency assessment of the focus sentence. Similar as in step a), a few-shot learning setting can be established based on which the language model is instructed to make a current prediction based on the additional data provided by the news articles. This prediction corresponds to the result of approach (3) introduced above. If we instruct the language model to make a suggestion for the current version of the focus sentence, we would ideally obtain "Sunak" for the current PM.

5 Evaluation

To demonstrate the performance of our approach in assessing and improving the currency of textual wiki data, we evaluate it on a set of fact-based sentences from Wikipedia containing currency-related statements. To this end, we have to show that our approach indeed is able to indicate whether a given currency-related statement is current, and – especially in the case of an outdated statement – whether it can provide a suggestion about the actually current statement,

thus improving its currency. In this section, we first outline the methodology for our evaluation and describe how our approach was operationalized in detail. We further discuss the dataset used, and finally present our results.

5.1 Methodology

To evaluate our approach, we applied it to fact-based sentences including one or more potentially outdated currency-related statements. We masked currency-related terms and then let our approach predict the current state of the masked currency-related term (cf. Figure 4). For many language models such as BERT, filling in a masked word based on the implicit knowledge of the model already is an innate task (Devlin et al. 2019). This (masking and prediction) procedure can be applied to each currency-related term of a considered wiki sentence separately. By doing so, it can not only be assessed whether the whole focus sentence is current (i.e., in case at least one currency-related term is outdated), but it can also be determined which currency-related terms are outdated. For all outdated currency-related terms, particular suggestions for improvement are provided.

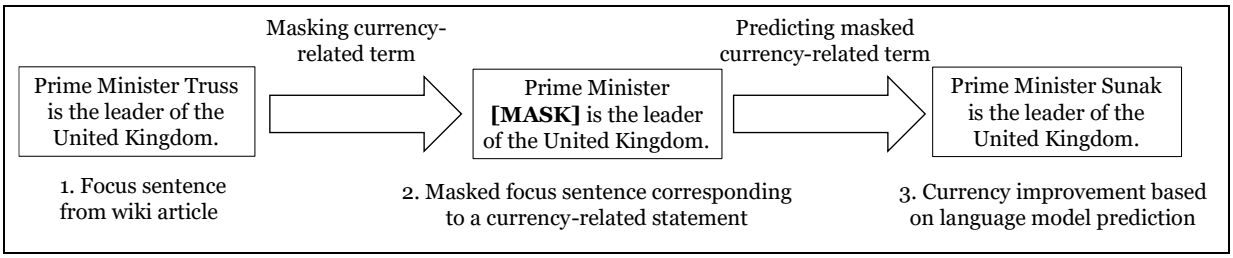


Figure 4. Currency Assessment and Improvement of a Focus Sentence

We evaluated and analyzed different variants of our language model-based approach, which mainly differ in the language model used and the additional external data. Hence, we analyzed the effectiveness of our approach along two dimensions: We first considered the two well-known language models BERT (Devlin et al. 2019) and GPT-3.5 (more precisely, the version gpt-3.5-turbo (OpenAI 2023a)) without any additional data. Since BERT was (inter alia) mainly pretrained on the English Wikipedia corpus in 2018 (Devlin et al. 2019), we did not expect BERT to have implicit knowledge about real-world changes which happened in 2019 or later. GPT-3.5, as a more recent and potentially more powerful language model, complements BERT as second baseline variant. As GPT-3.5 is mainly trained on data (including Wikipedia) from up to September 2021 (Brown et al. 2020; OpenAI 2023a), this variant can be expected to provide currency assessments and improvements for facts that have not changed in the real world since this point in time. By employing these baseline variants, we can separately analyze the performance of the sole language models as a foundation for currency assessment and improvement. This gives us the opportunity to isolate their respective contribution. More precisely, we strive to gain a deeper understanding of their differences by examining their predictions in terms of implicit knowledge and overall quality of predictions.

Second, the impact of additional data from news articles is evaluated in further variants. In particular, one of our goals in the evaluation is to analyze the influence of different sets of news

articles. To this end, we consider multiple variants using news sources which differ in quantity and characteristics of articles they comprise. Based on the performance of their variant, we can not only analyze their general suitability to support our approach, but also gain insights into the topics they cover, respectively. From this point on, we focus on GPT-3.5 as language model because in a preliminary test, its results were much more promising than BERT's. The first variant incorporating additional data from an external news source, *GPT + Guardian*, employs GPT-3.5 and whole online articles from the well-known British news outlet *The Guardian*[1]. Analogously, *GPT + Aylien* is an alternative variant utilizing additional data from the *Aylien News API*[2] by extracting excerpts of news articles from a multitude of different news outlets. Hence, the main difference between the *GPT + Aylien* and the *GPT + Guardian* variants lies in the pool of accessible news sources. In the case of *GPT + Aylien* this pool comprises a much higher number of different outlets, while in the case of the *GPT + Guardian* variant, it consists of the online articles of one single high-quality outlet which presumably only covers a certain share of real-world changes. As a fifth and final variant, we also consider the combination *GPT + News* which comprises both, the news articles from *The Guardian* and the *Aylien News API*. By doing so, we can examine whether the combination of multiple news sources positively affects the performance of our approach. The variants are summarized in Table 1.

Variant	Description
BERT	This variant uses the standard pre-trained BERT base model together with its inherent mask prediction task, which is part of BERT's pretraining procedure.
GPT	This variant uses the GPT-3.5 language model from OpenAI. The task for the language model is defined to respond with the current term for the [MASK].
GPT + Guardian	This variant uses the GPT-3.5 language model together with external data in the form of selected news articles from *The Guardian*.
GPT + Aylien	This variant uses the GPT-3.5 language model together with external data in the form of selected news articles provided by the *Aylien News API*.
GPT + News	This variant uses the GPT-3.5 language model together with external data in the form of selected news articles provided by *The Guardian* and the *Aylien News API*.

Table 1. Summary of the Variants of our Approach

5.2 Operationalization of the Approach

In the following, we describe the operationalization of our approach for the variants incorporating news articles in more detail.

a) Keyword Extraction: Keywords are extracted from each sentence using GPT-3.5 and two different prompts employing a few-shot learning setting (Brown et al. 2020) as described above. The two prompts (called Keywords_specific and Keywords_general) extract different sets of keywords, where Keywords_specific contains more keywords (which mandatorily must be

[1] https://open-platform.theguardian.com/

[2] https://aylien.com/product/news-api

contained in an article to be selected) than Keywords_general. Using different sets of keywords for the search increases the chances of finding suitable articles. They extracted on average 54 (Keywords_specific) and 89 (Keywords_general) articles, leading to a set of on average 115 articles without duplicates together. The full prompts are publicly available and provided online[3].

b) Article Search: For the news article search, the sources *The Guardian* and the *Aylien News API* are used. The search is limited to articles published from January 15th, 2023, to April 15 th, 2023. Each set of keywords is used for one search. *The Guardian* returns full news articles that contain all of the keywords. The *Aylien News API* returns only the summary of news articles whose titles contain all keywords. The results of both APIs are ordered by relevance to the search term.

c) Paragraph Extraction: Each result returned by the news sources is pre-processed by removing characters and strings not relevant to the content (e.g., HTML tags and layout information). Next, each article and summary is divided into smaller paragraphs consisting of three consecutive sentences. Each sentence is the start of one paragraph, meaning sentences 1, 2, and 3 form the first paragraph, sentences 2, 3, and 4 form the next paragraph, etc. Paragraphs that do not contain any of the keywords are removed, as it is very likely that they do not contain update-relevant data. Next, a text embedding of each paragraph and the masked sentence is calculated, which captures the semantic content of the text. To this end, we use the 1536-dimensional contextual text embedding *text-embedding-ada-002* from OpenAI (Greene et al. 2022) to convert text into comparable vectors of equal dimension. Then, the cosine similarity between the embedding of the masked sentence and the embedding of each paragraph is calculated and the paragraphs are ordered from highest to lowest similarity. If a sentence from a news source is contained in multiple paragraphs, only the paragraph with the highest cosine similarity is kept.

d) Selection of Paragraphs for the Language Model: The four most similar paragraphs that exceed a cosine similarity of an empirically determined threshold (0.78) are selected and passed to the language model as additional input. Using a prompt, the model is instructed to make a current prediction based on the paragraphs provided. This is done in a few-shot learning setting as described above. If no paragraph is selected, no additional data is passed to the language model and the prediction is obtained in the same manner as in the baseline variant.

5.3 Dataset

As there is no established dataset for the assessment and improvement of the currency of wiki sentences, we created a novel dataset comprising fact-based sentences from Wikipedia. To this end, we focused on sentences referring to facts that can become outdated at all, such as the current incumbent of a political position or the current club a certain football player is playing for, such that the sentences include at least one currency-related term. As Wikipedia was part of the training data for BERT and GPT 3.5, we also emphasized facts that have changed recently (2018

[3]https://docs.google.com/spreadsheets/d/188-iVFD0p0hL8UVBn-SyYT59IQmiOMLu_v4Lk3RTi9e8/edit?usp=sharing

or later). In this way, it could be avoided that large parts of our dataset are included in the training datasets of the language models. We extracted such facts based on Wikipedia articles and collected 543 sentences referring to those facts. To evaluate the performance of our approach across different domains, these sentences were collected across multiple domains of Wikipedia, namely football (164 sentences), (general) sports (34 sentences), politics (144 sentences), economics (98 sentences), science (62 sentences) and culture (41 sentences). Within these domains, sentences about people and facts with different backgrounds and varying degrees of public awareness were selected to further ensure a broad evaluation basis. In a pre-processing step, dispensable tokens such as phonetics and superfluous terms such as the second name of a person were removed from the Wikipedia sentences. On this basis, we identified the currency-related terms. We then masked one of the currency-related terms according to a fixed systematic approach determined beforehand (e.g., for a number of political positions, we always masked the incumbent). Finally, we labelled the sentence with the current term, which is consistent with the fact holding true in the real world, as gold token. The final labelling was conducted on April 21st, 2023 and each label was verified with an online search. For example, the gold token of the sentence "Prime Minister [MASK] is the leader of the United Kingdom" would be "Sunak", because he became PM in October 2022 and has been incumbent until April 21st, 2023. To provide full transparency, the dataset and all evaluations are publicly accessible[3].

5.4 Results

Even though it would be possible to evaluate multiple predictions (e.g., the top 3 predictions) of each variant, we deliberately focus on the top 1 prediction to make the evaluation very challenging. Providing only one prediction is also most helpful to support updates, as it allows for an easier decision (keep previously stored data or use the prediction). Thus, we generated the top 1 prediction for each currency-related term and for each variant of our approach as described in the subsection "Methodology". Two researchers manually checked every prediction for equality with the previously defined gold token and labelled the prediction as correct (thus current) in this case. Non-equal predictions potentially synonymous with the gold token were discussed by the researchers until consent was reached. The accuracy (percentage of semantically correct and current predictions) of each variant for all domains is presented in Table 2.

Variant / Dataset	BERT	GPT	GPT + Guardian	GPT + Aylien	GPT + News
Overall	0.335	0.615	0.713	0.816	0.821
Football	0.311	0.506	0.628	0.848	0.847
Sports (general)	0.294	0.647	0.676	0.764	0.705
Economy	0.296	0.612	0.653	0.684	0.673
Science	0.339	0.694	0.710	0.742	0.758
Culture	0.146	0.610	0.805	0.805	0.854
Politics	0.453	0.701	0.833	0.917	0.938

Table 2. Performance (Accuracy) of the Variants

5.5 Analysis of Predictions of GPT + News

For the strongest performing variant *GPT + News*, we additionally conducted a deeper analysis of the predictions. To this end, each prediction was examined manually. The results are presented in Table 3.

Correct predictions	**446**
(I): based on paragraphs	358
(II): based solely on implicit knowledge of the language model	88
Incorrect predictions	**97**
(III): no paragraphs are found	33
(IV): paragraphs are found, but do not contain update-relevant data	41
(V): paragraphs containing update-relevant data are found, but the data is not recognized	18
(VI): paragraphs contain retrospectives written in the present tense	5

Table 3. Detailed Evaluation of Variant GPT + News

Overall, 446 out of 543 predictions of the variant *GPT + News* are correct. In (I) 358 out of these 446 cases, paragraphs are extracted from external sources, and the correct predictions are made taking these paragraphs into account. In (II) 88 out of the 446 cases, no paragraphs are found or the paragraphs found do not contain update-relevant data, but still a correct prediction is made.

97 predictions of the variant *GPT + News* are incorrect. In (III) 33 of these cases, no paragraphs are found because no articles that contain the extracted keywords are available or none of the paragraphs have a similarity score above the threshold. Thus, the prediction is made based on the implicit knowledge of the model, which in this case is incorrect. In (IV) 41 cases, paragraphs are found, but they do not contain the update-relevant data for the currency-related term. As a result, the model gives an outdated or incorrect prediction. In (V) 18 cases, paragraphs containing update-relevant data are extracted from external sources, but the model does not recognize the update-relevant data and makes a wrong prediction. In nine of these cases, the update-relevant data is contained directly in the paragraphs. In the nine other cases, the update-relevant data is contained indirectly in the paragraphs and can be understood from the context (e.g., that James Taiclet was (and is not anymore) the CEO of American Tower can be understood even though it is not directly mentioned in the paragraph: "… American Tower Corp Says CEO Thomas Bartlett's 2022 Total Compensation …"). Finally (VI), in five cases, the paragraphs contain

retrospectives written in the present tense, and thus the model gives an outdated prediction based on the retrospective.

6 Discussion

In the evaluation section, we considered five different variants of our approach and evaluated their performance. In this section, we discuss these results and compare the results of the different variants. We then explicate our choice of parameters for the different variants and discuss the robustness of the results.

6.1 Discussion of Results

As a first baseline variant, we examined the language model BERT without any additional data, using only BERT's innate mask-fill task to predict the masked currency-related term of a focus sentence. As shown in Table 2, BERT achieves the lowest accuracy across all domains (0.335 overall). A deeper analysis of BERT's predictions reveals two main reasons for its rather poor performance in currency assessment and improvement. First, in many cases BERT does not capture the semantical content and the currency relevance of the focus sentence. Rather, it tends to make predictions based on known syntactic structures and patterns of adjacent words which both are adopted from the training corpus. This is in line with discussions in literature (Kassner and Schütze 2020; Poerner et al. 2020). For example, for the masked sentence "The host of the most recent Summer Olympics was the city [MASK]", BERT predicted "hall". The expression "city hall" might be reasonable in some other contexts, but in our case, disregarding the given context of the host city of the last Olympic games leads to a nonsensical answer. Second, as already pointed out in the previous section, BERT was mainly pretrained on the English Wikipedia corpus in 2018. Hence, BERT's implicit knowledge about the real world is outdated regarding facts which underwent real-world changes since this point in time. For example, this phenomenon becomes evident in sentences about persons who died recently (e.g., the American actor Lance Solomon Reddick, who died in March 2023), or politicians who no longer hold a political position they indeed once held (e.g., former German chancellor Angela Merkel, who held this position until December 2021). Here, BERT gives outdated predictions.

GPT as a second baseline variant for our approach achieved significantly higher accuracy than BERT across all domains (0.615 overall). Based on a detailed analysis of GPT's predictions, we found that this substantial improvement in accuracy is mainly due to the alleviation of BERT's major weaknesses discussed above. First, GPT's predictions have a higher overall quality in terms of awareness for the context and currency of the focus sentence. More precisely, GPT is consistently able to provide predictions that semantically match the actual content of the focus sentence, while not resorting to known syntactical patterns adopted from the training data. Importantly, this enables GPT to perform valid currency assessment and improvement, because an actual currency-related prediction is made based on the content of the sentence, and not based on known patterns from other contextual different sentences. Second, as already pointed out, GPT's implicit knowledge is more up to date because of its more recent training data. Therefore,

unlike BERT, GPT has the potential to correctly assess the currency of statements which underwent changes in the real world between 2018 and 2021. For these two reasons, GPT provides the current prediction "Tokyo" for the masked focus sentence "The host of the most recent Summer Olympics was the city [MASK]" without any additional input (the most recent Summer Olympics took place in Tokyo in July and August 2021). Overall, we found that GPT provides high-quality predictions for the currency assessment and improvement of textual wiki data and thus poses a suitable language model for our approach.

Thus, the three further variants are based on GPT with each variant utilizing a different basis of sources of news articles. *GPT + Guardian* produces 387 correct predictions, resulting in an overall accuracy of 0.713, strongly improving over the baseline GPT variant. *GPT + Aylien* performs even better, with 56 more correct predictions and an overall accuracy of 0.816. These outperformances against the baseline demonstrate that the incorporation of news articles is an effective strategy to assess the currency of textual wiki data. For example, while the baseline variants predict that Queen Elizabeth still is Queen of the UK based on their outdated implicit knowledge, the variants which incorporate news identify news articles which report the passing of Queen Elizabeth or refer to her as former Queen of the UK. They thus correctly suggest that Queen Elizabeth no longer is Queen of the UK.

The better performance by *GPT + Aylien* compared to *GPT + Guardian* is mainly due to two factors: First, the *Aylien News API* provides news articles for 43 more sentences than *The Guardian*. Second, the average similarity score of the paragraphs found by the *Aylien News API* is higher, indicating that the articles found are more relevant. This is supported by the fact that the predictions of *GPT + Guardian* are wrong twice as often compared to *GPT + Aylien* even when paragraphs are found. The reason for this is most likely that the *Aylien News API* retrieves articles from a large number of outlets, and as a result, *GPT + Aylien* has a much broader base of articles from which to select the most relevant paragraphs. Indeed, topics such as local sports are more likely to be covered by an outlet contained in the *Aylien News API* as compared to *The Guardian*. Also, *The Guardian* provides full articles, while the *Aylien News API* only retrieves article summaries. These summaries contain the update-relevant data in a more compact form, which is beneficial for our approach as it only uses paragraphs of a maximum length of three sentences.

GPT + News which combines both sources of news articles, shows the best performance with an overall accuracy of 0.821. Our analysis of the predictions of *GPT + News* shows that, for most sentences in the dataset, this variant finds relevant paragraphs articles and makes a correct prediction taking these paragraphs into account. However, compared to *GPT + Aylien*, only a small improvement can be observed. The *Aylien News API* retrieves the summaries of articles from many different news outlets, including articles from *The Guardian*. While the full articles from *The Guardian* provide paragraphs leading to an additional correct prediction in a few cases, occasionally, it can also be observed that using only one of the news sources leads to a correct prediction, but adding the other source interferes with it and an incorrect prediction results. Thus, adding *The Guardian* on top of the *Aylien News API* only leads to a minor improvement in overall performance.

The differences in performance between the five variants can be seen across all six domains (cf. Table 2). The exception is that, in a few domains, a slight decrease in accuracy from *GPT + Aylien* to *GPT + News* can be observed because of the reason described above. Across all variants politics always performs best, with *GPT + Aylien* and *GPT + News* achieving an accuracy above 0.9. The reason for this is that politics is a very well-covered topic in many news outlets. Thus, a multitude of articles are available from which the most relevant paragraphs can be extracted, leading to a high accuracy of predictions. In summary, all three variants, *GPT + Guardian*, *GPT + Aylien* and *GPT + News*, substantially outperform the baseline variants (cf. Table 2), showing the effectiveness of our approach. In particular, *GPT + Aylien* and *GPT + News* exhibit a very strong performance with an accuracy above 0.8. Thus, these variants are suitable for supporting the assessment and improvement of the currency of textual wiki data.

6.2 Choice of Parameters and Robustness

Our approach involves several parameters. In the following, we make transparent our choice of parameters for the different variants of our approach and discuss the robustness of our results.

The first parameter in our approach is the threshold which each paragraph passed to the language model must satisfy to ensure it has a minimum level of relevance to the focus sentence. Based on empirical tests, a value of 0.78 was selected as the value for this parameter. Slightly higher or lower values could also be chosen without strongly affecting the results, as the similarity score of a paragraph to the focus sentence is only an approximation of its relevance. The second parameter is the maximum number of paragraphs which are passed to the language model. For this parameter, the value four was selected. Based on our observations, usually, the first paragraph contains the most update-relevant data. However, sometimes update-relevant data is only contained in the third or fourth paragraph, despite having lower similarity scores. This is the reason for us to always pass four paragraphs to the model, but more (or less, e.g., three) paragraphs could also be passed without strongly affecting the results. The third parameter is number of sentences included in each paragraph. Here, the value three was selected. Based on our observations, in many cases, one single sentence contains all update-relevant data. However, in some cases, all three sentences contain update-relevant data. Therefore, to not disregard any update-relevant data, we decide to include three sentences in one paragraph. Again, slightly altering the value of this parameter would not substantially change the results. The fourth parameter of our approach is the search period, which determines the maximum age of eligible articles. As described above, the trade-off between the number of articles and their currency must be considered when specifying this parameter. Using a longer search period results in more data, but some of this data may already be outdated, making it irrelevant or even detrimental. We chose a search period of three months because it is long enough to find data even for less popular topics, but not so long that outdated articles are retrieved. As with the other parameters, smaller changes would not significantly affect the results.

In general, the main factor in the choice of the described parameters is the trade-off between the amount of data and its relevance: By increasing the threshold, decreasing the number of

paragraphs, decreasing the number of sentences per paragraph, or decreasing the length of the search period, less but potentially more relevant data is passed to the model, which means focusing on relevance. By doing the opposite, more but potentially less relevant data is passed to the model, focusing on the amount of data. This is the focus of our choice of parameters. We instructed the model to recognize if the paragraphs do not contain any update-relevant data, and to make a prediction based on implicit knowledge in this case. This worked well for our model, as irrelevant data only very rarely led to an incorrect prediction (cf. Section "Analysis of Predictions of *GPT + News*"). Indeed, if the model can be instructed in a way that irrelevant data never leads to a wrong prediction, a strong focus on the amount of data is favorable. On the other hand, the easier the model can be distracted by irrelevant data, the more the choice should focus on relevance.

We further analyzed the different variants of our approach and their performance for currency improvement with respect to just recently changed sentences to infer whether the knowledge of the language models (attained during their training period) plays a significant role for our results. More precisely, we considered only sentences (of the overall 543 sentences) for which the last change of the gold token occurred after a certain cutoff date and let the cutoff date vary. The results of these analyses are presented in Table 4.

Variant Cutoff date	BERT	GPT	GPT + Guardian	GPT + Aylien	GPT + News	Number of sentences
01/01/18	0.251	0.456	0.600	0.758	0.763	355
01/01/19	0.231	0.436	0.585	0.751	0.757	337
01/01/20	0.214	0.382	0.553	0.730	0.737	304
01/01/21	0.219	0.310	0.498	0.709	0.716	261
01/01/22	0.235	0.164	0.399	0.634	0.667	183
01/01/23	0.156	0.190	0.483	0.621	0.672	58

Table 4. Performance (Accuracy) of the Variants regarding Different Cutoff Dates

The declining performance of the baseline variants *BERT* and *GPT* with more recent cutoff dates clearly indicates that their training data mainly lie a few years in the past. Importantly, the results show that incorporating news articles is beneficial for each cutoff date, with the most benefit realized for more recent cutoff dates (i.e., more recently changed facts). For instance, while the baseline variant *GPT* only achieved an accuracy of 0.190 for a cutoff date of 01/01/2023, the variant *GPT + News* achieved an accuracy of 0.672. This means that *GPT + News* could be used to improve the currency of 3.54 times as many sentences, stressing the importance of incorporating news articles for currency improvement, and thus, our approach.

Further, we analyzed the relation between the number of news articles found for a focus sentence and the performance of our approach. This would give us an idea whether our approach would benefit from including more (diverse) sources besides *The Guardian* and the *Aylien News API* leading to a higher number of found news articles. For that reason, we categorized each sentence in the dataset into one of the four categories "no news coverage" (0 news articles found in the sources for *GPT + News*), "limited coverage" (1-49 articles), "moderate coverage" (50-200

articles), and "extensive coverage" (>200 articles). The results of the variants are presented in Table 5. The results clearly show that higher coverage leads to improved performance for the variant *GPT + News* (naturally not for the baseline variants, which do not incorporate news), with its accuracy steadily increasing from 0.671 (no coverage) to 0.908 (extensive coverage). This further highlights the benefits of incorporating additional data for currency improvement and especially of including more and diverse sources (in the future).

Variant / Coverage	BERT	GPT	GPT + Guardian	GPT + Aylien	GPT + News	Number of sentences
No news coverage	0.303	0.632	0.671	0.671	0.671	76
Limited coverage	0.335	0.608	0.634	0.773	0.773	194
Moderate coverage	0.322	0.678	0.760	0.909	0.884	121
Extensive coverage	0.361	0.566	0.796	0.868	0.908	152

Table 5. Performance (Accuracy) of the Variants regarding News Coverage

Finally, we also examined which results our approach yields when applied with the new language model GPT-4 (more precisely, the version GPT-4-0613 (OpenAI 2023b)). To this end, we employed two additional variants of our approach, *GPT-4* and *GPT-4 + News*, which were constructed and applied analogously to the corresponding variants with GPT-3.5. *GPT-4* yielded an accuracy of 0.624, marginally outperforming the variant with GPT-3.5 (accuracy of 0.615). This can be explained by the fact that GPT-4 is a more powerful language model, but like GPT-3.5, is mainly trained on data from up to September 2021 (OpenAI 2023b). *GPT-4 + News* was the best performing variant overall with an accuracy of 0.840, as it was slightly better in using the additional data from news articles than the variant with GPT-3.5 (accuracy of 0.821). This indicates that our approach has the potential to provide even stronger results with possibly more powerful language models in the future.

7 Conclusion, Limitations and Future Work

Wikis are ubiquitous in organizational and private use and provide a wealth of textual data. Maintaining the currency of this textual data is both important and difficult, requiring large manual efforts. Previous approaches from literature provide valuable contributions for assessing the currency of structured data or wiki articles as a whole but are unsuitable for textual wiki data. Thus, we propose both a theoretical model and a novel approach which support the assessment and improvement of the currency of textual wiki data in an automated manner. The theoretical model comprehensively describes currency changes in wikis. Grounded on this theoretical model, our approach makes use of data retrieved from recently published news articles and a language model to determine the currency of fact-based wiki sentences and suggest possible updates. Our evaluation conducted on 543 sentences from six domains shows that the approach yields promising results with accuracies over 80% (focusing only on the top 1 prediction and a 3-month window for selecting news articles, which both make the evaluation very challenging). Thus, it is well-suited to support the assessment and improvement of the currency of textual wiki data. It can be employed to continually assess the currency of data which may become outdated, or to conduct

specific assessments (e.g., when the last update of a currency-related term was a year ago, or for all football-related data when the football transfer period ends).

Nevertheless, the work at hand has some limitations which could be a starting point for future research. To begin with, the presented operationalization of the approach does not yet take full advantage of the entire theoretical model and the general approach. Future work could incorporate known data stored in the wiki in the past, and it could explicitly consider information about changes of real-world data over time (e.g., leading to individual "decline rates" of textual data). This way, the performance could be further improved. In addition, we applied our approach with fixed parameter settings in the evaluation. The approach could be refined to adjust the parameters depending on the domain or data (e.g., using a shorter search period for data which changes highly frequently). Furthermore, our approach relies on obtaining data from recently published news articles to facilitate current predictions, and thus, updates. In the future, it could be extended to additionally consider further trustworthy sources (e.g., websites of universities and corporations). Moreover, our approach needs to be applied frequently to fact-based sentences in a wiki to assess their currency. An intriguing idea which could be explored is to apply our approach in "reverse order" by first extracting current data from news articles as soon as they are published, and then searching and updating respective textual data in the wiki. Finally, we focused on public wikis in this paper. It would be highly interesting to extend our approach to organizational knowledge bases, such as enterprise wikis or texts in a document management system. Here, updates could be facilitated by other, more recent documents in the same organization.

8 References

Alqahtani, F. H. 2017. "Users' Perspectives on Using Wikis for Managing Knowledge: Benefits and Challenges," *Journal of Organizational and End User Computing* (29:3), pp. 1-23.

Ballou, D. P., Wang, R. Y., Pazer, H. L., and Tayi, G. K. 1998. "Modeling Information Manufacturing Systems to Determine Information Product Quality," *Management Science* (44:4), pp. 462-484.

Batini, C., Barone, D., Cabitza, F., and Grega, S. 2011. "A Data Quality Methodology for Heterogeneous Data," *International Journal of Database Management Systems* (3:1), pp. 60-79.

Batini, C., and Scannapieco, M. 2016. *Data and Information Quality*, Cham: Springer.

BBC. 2022. "Rishi Sunak: World Leaders Welcome Next UK Prime Minister," available at https://www.bbc.com/news/uk-63378673, accessed on September 1st, 2023.

Beck, R., Rai, A., Fischbach, K., and Keil, M. 2015. "Untangling Knowledge Creation and Knowledge Integration in Enterprise Wikis," *Journal of Business Economics* (85:4), pp. 389-420.

Bhatti, Z. A., Baile, S., and Yasin, H. M. 2018. "Assessing Enterprise Wiki Success from the Perspective of End-Users: An Empirical Approach," *Behaviour & Information Technology* (37:12), pp. 1177-1193.

Blumenstock, J. E. 2008. "Size Matters: Word Count as a Measure of Quality on Wikipedia," in *Proceedings of the 17th International Conference on World Wide Web*, Beijing, China, pp. 1095-1096.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. 2020. "Language Models are Few-Shot Learners," in *Advances in Neural Information Processing Systems*, pp. 1877-1901.

Cichy, C., and Rass, S. 2019. "An Overview of Data Quality Frameworks," *IEEE Access* (7), pp. 24634-24648.

Dang, Q.-V., and Ignat, C.-L. 2017. "An End-to-End Learning Solution for Assessing the Quality of Wikipedia Articles," in *Proceedings of the 13th International Symposium on Open Collaboration,* Berlin, Germany, pp. 1-10.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. 2019. "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies,* Minneapolis, Minnesota, pp. 4171-4186.

Firmani, D., Mecella, M., Scannapieco, M., and Batini, C. 2016. "On the Meaningfulness of "Big Data Quality"," *Data Science and Engineering* (1:1), pp. 6-20.

Greene, R., Sanders, T., Weng, L., and Neelakantan. 2022. "New and improved embedding model," available at https://openai.com/blog/new-and-improved-embedding-model, accessed on April 30th, 2023.

Hao, S., Chai, C., Li, G., Tang, N., Wang, N., and Yu, X. 2020. "Outdated Fact Detection in Knowledge Bases," in *Proceedings of the IEEE 36th International Conference*, pp. 1890-1893.

Hatcher-Gallop, R., Fazal, Z., and Oluseyi, M. 2009. "Quest for Excellence in a Wiki-Based World," in *Proceedings of the IEEE International Professional Communication Conference*, pp. 1-8.

He, W., and Yang, L. 2016. "Using Wikis in Team Collaboration: A Media Capability Perspective," *Information & Management* (53:7), pp. 846-856.

Heinrich, B., and Hristova, D. 2014. "A Fuzzy Metric for Currency in the Context of Big Data," in *Proceedings of the 22nd European Conference on Information Systems,* Tel Aviv, Israel.

Heinrich, B., Hristova, D., Klier, M., Schiller, A., and Szubartowicz, M. 2018. "Requirements for Data Quality Metrics," *Journal of Data and Information Quality* (9:2), pp. 1-32.

Heinrich, B., and Klier, M. 2011. "Assessing Data Currency—A Probabilistic Approach," *Journal of Information Science* (37:1), pp. 86-100.

Heinrich, B., and Klier, M. 2015. "Metric-Based Data Quality Assessment — Developing and Evaluating a Probability-Based Currency Metric," *Decision Support Systems* (72), pp. 82-96.

Heinrich, B., Klier, M., and Kaiser, M. 2009. "A Procedure to Develop Metrics for Currency and its Application in CRM," *Journal of Data and Information Quality* (1:1), pp. 1-28.

Kassner, N., and Schütze, H. 2020. "Negated and Misprimed Probes for Pretrained Language Models: Birds Can Talk, But Cannot Fly," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 7811-7818.

Klier, M., Moestue, L., Obermeier, A., and Widmann, T. 2021. "Event-Driven Assessment of Currency of Wiki Articles: A Novel Probability-Based Metric," in *Proceedings of the 42nd International Conference on Information Systems,* Austin, TX.

Lewoniewski, W. 2017. "Enrichment of Information in Multilingual Wikipedia based on Quality Analysis," in *Business Information Systems Workshops. BIS 2017. Lecture Notes in Business Information Processing*, W. Abramowicz (ed.), Cham: Springer, pp. 216-227.

Lewoniewski, W. 2019. "Measures for Quality Assessment of Articles and Infoboxes in Multilingual Wikipedia," in *Business Information Systems Workshops. BIS 2018. Lecture Notes in Business Information Processing*, W. Abramowicz and A. Paschke (eds.), Cham: Springer, pp. 619-633.

MediaWiki. 2023. "ORES," available at https://www.mediawiki.org/wiki/ORES, accessed on April 30th, 2023.

Nelson, R., Todd, P., and Wixom, B. 2005. "Antecedents of Information and System Quality: An Empirical Examination within the Context of Data Warehousing," *Journal of Management Information Systems* (21:4), pp. 199–235.

OpenAI. 2023a. "GPT-3.5," available at https://platform.openai.com/docs/models/gpt-3-5, accessed on April 30th, 2023.

OpenAI. 2023b. "GPT-4," available at https://platform.openai.com/docs/models/gpt-4, accessed on September 1st, 2023.

Orr, K. 1998. "Data Quality and Systems Theory," *Communications of the ACM* (41:2), pp. 66-71.

Ott, H. 2022. "Rishi Sunak Appointed Prime Minister of the United Kingdom," available at https://www.cbsnews.com/news/rishi-sunak-appointed-prime-minister-of-the-united-kingdom/, accessed on September 1st, 2023.

Poerner, N., Waltinger, U., and Schütze, H. 2020. "E-BERT: Efficient-Yet-Effective Entity Embeddings for BERT," in *Findings of the Association for Computational Linguistics*, pp. 803-818.

Seibert, M., Preuss, S., and Rauer, M. 2011. *Enterprise Wikis: Die erfolgreiche Einführung und Nutzung von Wikis in Unternehmen*, Wiesbaden: Gabler Verlag.

Shah, A. A., Ravana, S. D., Hamid, S., and Ismail, M. A. 2015. "Web Credibility Assessment: Affecting Factors and Assessment Techniques," *Information Research* (20:1), pp. 1-28.

Shen, A., Qi, J., and Baldwin, T. 2017. "A Hybrid Model for Quality Assessment of Wikipedia Articles," in *Proceedings of the Australasian Language Technology Association Workshop*, pp. 43-52.

Stvilia, B., Twidale, M. B., Smith, L. C., and Gasser, L. 2005. "Assessing Information Quality of a Community-Based Encyclopedia," in *Proceedings of the International Conference on Information Quality,* Cambridge, MA, pp. 442-454.

Tran, T., and Cao, T. H. 2013. "Automatic Detection of Outdated Information in Wikipedia Infoboxes," *Research in Computing Science* (70:1), pp. 211-222.

Wang, P., and Li, X. 2020. "Assessing the Quality of Information on Wikipedia: A Deep-Learning Approach," *Journal of the Association for Information Science and Technology* (71:1), pp. 16-28.

Wang, R. Y., and Strong, D. M. 1996. "Beyond Accuracy: What Data Quality Means to Data Consumers," *Journal of Management Information Systems* (12:4), pp. 5-33.

Wechsler, A., and Even, A. 2012. "Using a Markov-Chain Model for Assessing Accuracy Degradation and Developing Data Maintenance Policies," in *Proceedings of the 10th International Conference on Design Science Research in Information Systems*, K. D. Joshi and Y. Yoo (eds.).

Wikimedia. 2023. "Wikimedia Statistics," available at https://stats.wikimedia.org/#/all-wikipedia-projects, accessed on April 30th, 2023.

Wikipedia. 2023. "Wikipedia: Size of Wikipedia," available at https://en.wikipedia.org/wiki/Wikipedia:Size_of_Wikipedia#:~:text=As%20of%2015%20March%202023,of%20all%20pages%20on%20Wikipedia., accessed on April 30th, 2023.

Wöhner, T., Köhler, S., and Peters, R. 2015. "Good Authors= Good Articles?-How Wikis Work," in *Proceedings of the Wirtschaftsinformatik,* Osnabrück, Germany, pp. 872-886.

Yates, D., Wagner, C., and Majchrzak, A. 2009. "Factors Affecting Shapers of Organizational Wikis," *Journal of the American Society for Information Science and Technology* (61:3), 543-554.

Zak, Y., and Even, A. 2017. "Development and Evaluation of a Continuous-Time Markov Chain Model for Detecting and Handling Data Currency Declines," *Decision Support Systems* (103), pp. 82-93.

Zhang, H., Ren, Y., and Kraut, R. E. 2020. "Mining and Predicting Temporal Patterns in the Quality Evolution of Wikipedia Articles," in *Proceedings of the 54th Hawaii International Conference on System Sciences,* Lahaina, HW, pp. 1-10.

3 Explaining Machine Learning Model Predictions

This section contains three research papers which focus on the second focal topic of this dissertation, explaining ML model predictions. In Section 3.1, model uncertainty and explanation uncertainty are conceptually decomposed and distinguished, allowing for a more detailed understanding and assessment of these uncertainties and their effect on the consistency and correctness of feature attribution explanations (RQ5 & RQ6). In Section 3.2, a method for the exact reconstruction of the decision boundaries of ML models is introduced and utilized for rigorous sensitivity analyses of model predictions (RQ7 & RQ8). Finally, in Section 3.3, this method is leveraged to derive faithful feature attribution and counterfactual explanations that are not affected by explanation uncertainty (RQ9).

3.1 Paper 5: Are Consistent Explanations Also Correct? The Impact of Explanation and Model Uncertainty

Current Status	**Full Citation**
This paper is **under review** (since 02/2026) for publication in the *Proceedings of the 2026 ACM SIGKDD Conference on Knowledge Discovery and Data Mining*	Dünisch, J., Heinrich, B., Krapf, T., & Miethaner, P. (2026). Are Consistent Explanations Also Correct? The Impact of Explanation and Model Uncertainty. *Working Paper*, University of Regensburg

Abstract:

Machine learning (ML) models are increasingly applied in high-stakes domains such as healthcare, finance, and autonomous driving. Despite their strong predictive performance, many state-of-the-art ML models operate as black boxes, limiting transparency and human oversight. Against this background, various Explainable AI (XAI) methods have been proposed, particularly feature attribution techniques, which aim to provide human-understandable explanations by assigning importance scores to input features. However, these explanations are subject to uncertainty, arising both from the XAI methods themselves and from the ML model being explained. Prior work addresses this uncertainty by assessing multiple explanations regarding their consistency, but it remains unclear whether consistent explanations are also correct, or whether they merely constitute consistently incorrect explanations. In this paper, we precisely examine the agreement between feature attribution explanations provided by XAI methods and the underlying ground truth. We formally de-compose explanation error into four distinct factors—XAI structural bias, XAI estimation variance, ML Model structural bias, and ML Model estimation variance—and evaluate their effects on the correctness of explanations provided by well-known XAI methods across various datasets in an extensive experimental analysis. Our results demonstrate that consistent explanations are rarely correct, which has severe implications for XAI research and for the use of XAI methods in high-stakes applications.

Keywords: XAI, feature attribution, model uncertainty, explanation uncertainty, explanation consistency, explanation correctness

Please note that the overall formatting of the paper has been adjusted to the remainder of the dissertation.

1 Introduction

While Machine Learning (ML) models advance in high-stakes domains [2, 5, 16, 22, 30, 36], their lack of transparency necessitates Explainable AI (XAI) to ensure accountability and regulatory compliance [13]. Thus, Feature Attribution (FA) methods such as LIME [38], SHAP [28], or Integrated Gradients [42] have become standard for quantifying feature importance. The goal of FA methods is to associate a weight/value to each feature, thus indicating its individual importance and attribution to the model's class prediction. However, it has been recognized that FA explanations are subject to uncertainty, potentially leading, e.g., to many differing explanations for the same data instance. This uncertainty is particularly critical in high-stakes applications and raises fundamental questions about their correctness [19, 25, 32, 34, 47].

We distinguish two uncertainty types affecting correctness (cf. Figure 1). First, *explanation uncertainty* stems from the XAI method itself, manifesting as disagreement between methods [10, 24, 34] or instability under sampling/perturbations [3, 11], even when re-applying the method on the exact same instance [43, 48]. Explanation uncertainty corresponds with the *Explanation Model agreement* in Figure 1. Second, *model uncertainty* arises because the ML model is merely an approximation of the unknown ground truth (GT), often failing to capture the underlying data-generating process perfectly. In Figure 1, this kind of uncertainty corresponds with the *Model GT agreement*. As a result, it is unclear whether FA explanations derived from a particular ML model reflect the actual importance of features with respect to the underlying GT, regardless of the uncertainty introduced by the used XAI method. This is caused by model uncertainty, i.e., alternative models with comparable predictive performance may yield substantially differing explanations for the same input. In ML, this phenomenon is linked to the Rashomon effect, which refers to the existence of multiple ML models (Rashomon set, also cf. Figure 1) that achieve similar predictive performance yet may exhibit substantially different decision behavior [6, 9]. This effect arises from limited or imperfect data, stochastic elements in training, and differences in ML model types and their inductive biases [7, 9, 14, 25]. Such models often yield various explanations that cannot be regarded as valid alternatives when they are contradictory. In such cases, at most one explanation can reflect the GT, while the others are artifacts of model generation rather than of reality. Consequently, model uncertainty implies that optimizing for predictive performance does not sufficiently constrain decision behavior to ensure explanatory correctness.

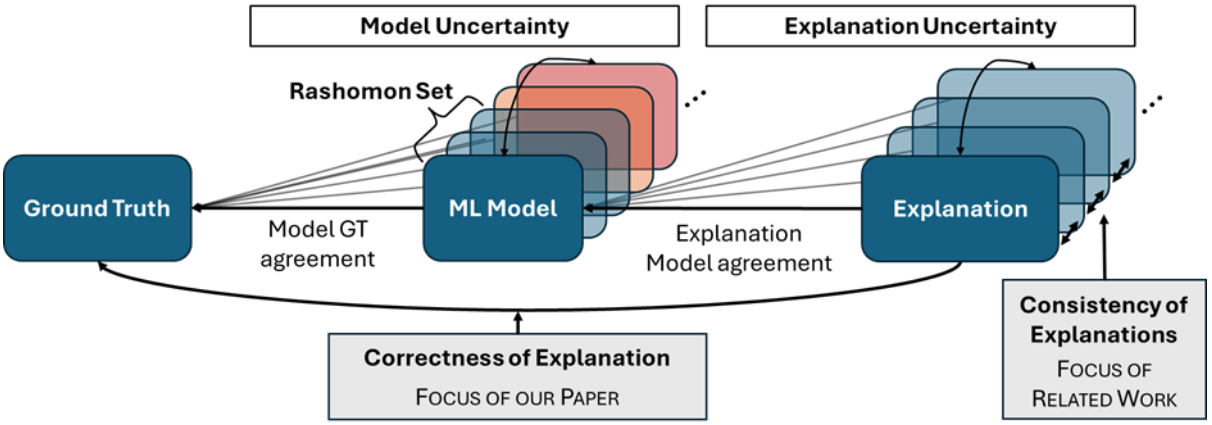


Figure 1: Overview of Model and Explanation Uncertainty, Correctness and Consistency of Explanations

Several works address the uncertainty of FA explanations. For example, focusing on explanation uncertainty, researchers have proposed to aggregate multiple explanations from the same or different XAI methods for a particular ML model and data instance [10, 33, 34, 40]. In Figure 1, this is referred to as consistency of multiple explanations for one ML model. Furthermore, other approaches specifically address the impact of model uncertainty on FA explanations, e.g., by extracting (consistent) patterns [25] or by constructing confidence intervals [14, 29] from the set of explanations obtained by different ML models from the Rashomon set.

While these works provide valuable insights for dealing with uncertain explanations, they primarily focus on (in)consistency rather than the actual correctness of FA explanations with respect to the GT. Crucially, it remains unproven whether consistent explanations actually reflect the true feature importance, or if they merely converge to a stable but incorrect value. This is of critical relevance, as in reality, explanations are often trusted just because they appear consistent across ML models. This shapes the assessments of fairness, accountability, or confidence for model predictions, which can have direct impact on human lives. Conflating consistency with correctness risks masking structural flaws of XAI methods, creating a false sense of trust in high-stakes scenarios. Hence, the following research questions arise: (1) *Does consistency across FA explanations serve as a reliable indicator for correctness, or does it mask structural bias regarding the GT?* (2) *How can the error of FA explanations be formally decomposed to isolate the specific effects of explanation and model uncertainty?*

While prior work focuses on reducing variance of explanations, we show that structural bias (expressiveness limits) renders explanations incorrect even when they are perfectly consistent reflecting a classic Bias-Variance trade-off regarding explanation correctness (cf. Figure 1). Our contributions can be summarized as follows:

1. Formal Error Decomposition: We propose a rigorous decomposition of explanation error into four distinct factors: XAI structural bias, XAI estimation variance, ML Model structural bias, and ML Model estimation variance. This allows us to isolate the sources of incorrect explanations.

2. The Consistency-Correctness Relationship: We empirically demonstrate that explanation consistency (low variance) is a necessary but insufficient condition for correctness. We identify the 'Consistency Trap', showing that increasing sample size stabilizes explanations around an incorrect value if the structural bias of the XAI method is high.
3. The Stability-Sensitivity Trade-off: We reveal that popular model-agnostic methods (e.g., LIME, SHAP) fail to reflect model degradation caused by data scarcity. While they appear stable, they lack the sensitivity to capture the shifting DBs of the underlying ML model, rendering them consistently incorrect.

2 Related Work

In this section, we first provide a brief overview of XAI methods that generate explanations in the form of FAs and summarize their key methodological principles. We then review approaches that aim to address the uncertainty inherent in XAI explanations, e.g., by analyzing their consistency or combining multiple explanations. Finally, we discuss works that address model uncertainty in the context of explanations, e.g., by extracting consistent patterns from explanations of different ML models within the Rashomon set.

Feature Attribution XAI Methods. Literature provides many local XAI methods m that assign FA vectors $\phi^m(x, f)$ to explain model predictions. Common strategies include gradient-based methods like the Integrated Gradients method [42]. LIME [38] and SHAP [28] (in their implementation of Kernel SHAP) fit linear regressions on perturbed samples to estimate importance, with SHAP grounding in cooperative game theory. Variants like ROLEX [22], LS [26] or LEAP [21] optimize sampling schemes for better local fidelity. Conversely, EXPLORE [17] avoids sampling by utilizing piecewise linear reconstruction to extract decision boundaries (DBs) functionally.

Explanation Uncertainty. Researchers have recognized substantial variations in the FA explanations for the same data instance provided by different XAI methods or even by repetitions of the same XAI method, raising concerns about their correctness. Aiming to address this explanation uncertainty, specific methods have been proposed that aggregate multiple explanations from the same or different XAI methods. Schulz et al. [40] generate multiple LIME explanations $\left(\phi_i^{LIME}(x, f)\right)_{i=1,\dots,n}$ by bootstrapping the sample set Z, and estimate the variability and consistency across the obtained explanations. Neuhof et al. [33] study the uncertainty of a global SHAP FA $\phi^{SHAP}(f)$ and construct confidence intervals for feature ranks based on multiple local attributions $\left(\phi^{SHAP}(x_i, f)\right)_{i=1,\dots,n}$. One method that considers explanations of multiple methods is AGREE [34] which uses FAs $\left(\phi^{m_i}(x, f)\right)_{i=1,\dots,n}$ by LIME, SHAP, Integrated Gradients, and MAPLE [35] and weighs them based on their alignment with explanations for similar data instances and by comparing their feature rankings against each other. The method AGGOPT [10] pursues a similar approach by combining FAs from multiple methods based on their fidelity and robustness. The aggregation is posed as a constrained quadratic optimization problem, with

guarantees that the resulting explanations are superior to weighted averages of the individual XAI methods.

Overall, this research field widely acknowledges that explanations provided by XAI methods are subject to explanation uncertainty, and existing works address this issue by analyzing the consistency of explanations of one or multiple methods. However, the correctness of the resulting explanations with respect to the GT (which underlies the ML model, left in Figure 1) has not been systematically analyzed and therefore remains unclear.

Model Uncertainty. It has also been recognized that a variety of ML models $(f_i)_{i=1,\dots,n}$ can be obtained, each achieving comparably good empirical performance while providing different explanations $\left(\phi^m(x, f_i)\right)_{i=1,\dots,n}$ even for the same class prediction $f_i(x)$. This impact of model uncertainty and the Rashomon effect on explanations has been subject in literature ever since. Based on different explanations derived over the models from the Rashomon set, Laberge et al. [25] aim to extract correct partial feature ranking orders that are supported by a sufficiently large subset. Further, Marx et al. [29] adopt a model uncertainty-aware perspective on XAI by estimating the best-performing model within a set of ML models and deriving confidence intervals for its explanations. Fisher et al. [14] also investigate FAs across multiple ML models, deriving confidence bounds based on permutation importance scores for the Rashomon set. Similarly, the method Variable Importance Clouds [12] analyzes FAs across the Rashomon set for linear models and decision trees (DTs), using perturbation-based importance estimates to derive lower and upper bound intervals that capture variability across equally accurate models.

Overall, prior work primarily addresses model uncertainty in explanations through the Rashomon effect by analyzing the consistency of explanations across comparably well-performing models. However, it is not analyzed that models from the Rashomon set may still fail to *correctly* reflect the underlying GT. Moreover, effects of explanation uncertainty are typically not considered. In summary, existing research provides valuable insights into the reasons for inconsistent explanations provided by XAI methods and proposes techniques for inferring consistent patterns in such explanations. However, it remains an open question whether consistent explanations are also correct, or whether they merely constitute consistently incorrect explanations.

3 Background and Basic Ideas

As discussed above, a high Model GT agreement together with a high Explanation Model agreement yields a high explanation correctness (cf. Figure 1). To enable a detailed assessment of explanation correctness with respect to the GT we conceptualize the GT not just as a data-generating distribution, but as an oracle function $f^*: X \to Y$ that possesses class subspaces (areas in the feature space where the GT corresponds to one single class) and DBs. In the following, we introduce and discuss factors related to explanation uncertainty and model uncertainty, both of which encompass aleatoric and epistemic components [20], affecting explanation correctness.

3.1 Explanation Uncertainty

Many ML models f constitute black boxes, thus it is not known whether an explanation $\phi^m(x, f)$ provided by an XAI method m correctly reflects the actual model behavior, and which FAs actually cause a class change. In this regard, explanation uncertainty arises from two main factors.

First (1a), the uncertainty of explanations is affected by the ability of an XAI method to reflect the actual decision behavior of an ML model due to its methodological *structural bias*. In many cases, the procedures used to compute explanations are heuristic in nature and therefore rely on simplifying model assumptions. One example of this is the use of a surrogate model (e.g., a linear regression) to approximate the black-box model. If the underlying model f exhibits highly non-linear and complex structure in the vicinity of a given data instance x, an XAI method m that relies on a linear surrogate struggles with Explanation Model agreement for x, since $f(x') \approx \phi^m(x, f)^T \cdot (x' - x) + f(x)$ is assumed for close instances $x' \in X$. Consequently, explanation uncertainty is induced by the structural bias of an XAI method. i.e., whether or not it can correctly reflect the considered ML model regarding its model type, e.g., neural network (NN), and its capacity.

Second (1b), regardless of the structural bias of XAI methods, their correctness is also affected by their *estimation variance* resulting from the information used about the behavior of the explained model. A common procedure related to the factor of estimation variance is the generation of random samples and perturbations $Z = \{(x_i, f(x_i))\}_{i=1}^n \subset X \times Y$, as employed by many methods such as LIME, SHAP, and ROLEX. By relying on samples and perturbations, XAI methods typically operate with incomplete or indirect analysis of the ML model f they aim to explain. While sampling and perturbations by themselves do not impose specific structural assumptions on the model behavior (as described in 1a), they nevertheless are subject to sparsity, especially in high-dimensional feature spaces. Formally, complete information on the model is given by the set $Z^* = \{(x, f(x)) \subset X \times Y \mid \forall x \in X\}$ over the whole feature space X, of which Z typically is a small discrete subset. The resulting explanations are thus derived based on partial information on the model's DBs, affecting the Explanation Model agreement. Because of the randomness of samples or perturbations, explanations provided by many XAI methods can also vary across repeated computations for the same instance, leading to uncertainty. More precisely, multiple sets $Z^1, \dots, Z^K$, each containing random samples $(z^i, f(z^i)) \in Z^i$ where z^i is obtained from a probability distribution P_X over X, induce multiple explanations $\phi^m(x, f, Z^1), \dots, \phi^m(x, f, Z^K)$ (cf. Figure 1). Hence, stability or robustness is a widely recognized criterion in the evaluation of XAI methods [1, 32].

The errors stemming from the factors (1a/1b) can be formalized as follows: Under perfect information Z^*, the structural error of an FA explanation provided by XAI method m to the true FA explanation $\phi^*(x, f)$—represented by the true decision behavior of the model given by its DBs—is solely due to the structural bias of m. We denote this deviation by $\Delta_{1a} = \phi^m(x, f, Z^*) - \phi^*(x, f)$. Further, the error resulting from imperfect information $Z \subset Z^*$ used by m can be denoted by $\Delta_{1b} = \phi^m(x, f, Z) - \phi^m(x, f, Z^*)$. Thus, Δ_{1a} represents the structural bias

of the XAI method (systematic error due to model type mismatch), while Δ_{1b} represents the estimation variance (error due to finite sampling).

3.2 Model Uncertainty

For model uncertainty, we again distinguish between two main factors (2a/2b) that affect explanation correctness. We discuss both factors based on the Rashomon effect phenomenon [6, 25]. Crucially, even when different ML models fit the same data comparably well, they still may differ substantially in their behavior. Formally, let $\mathcal{F}$ denote the set of all possible models $f: X \rightarrow Y$ that can be trained on a dataset $D = \{(x_i, y_i)\}_{i=1}^{n} \subset X \times Y$. Let $L(f, D)$ represent the empirical loss of model f on D, and let $L^* = min_{f \in \mathcal{F}} L(f, D)$ denote the minimum achievable loss within $\mathcal{F}$. Following Fisher et al. [14], for a tolerance parameter $\varepsilon \geq 0$, we define the Rashomon set as $\mathcal{R}(\varepsilon, D) = \{f \in \mathcal{F} | L(f, D) \leq L^* + \varepsilon\}$ comprising all models that achieve an empirical loss within ε. Although the models in $\mathcal{R}(\varepsilon, D)$ perform very similarly according to the chosen metric, they may differ substantially in their decision behavior [18] and thus, in their learned DBs. As a result, the explanations and FAs that XAI methods provide for the same data instance can vary significantly across these ML models [25, 41].

Regarding factor (2a), the ability of a certain ML model in the Rashomon set to correctly capture the underlying GT depends on the *ML model structural bias* determined by its respective type and its capacity, e.g., the number of NN layers and neurons. Different model types exhibit different inductive biases and representational capabilities, making them more or less suitable for capturing a specific GT f^*. In the definition of $\mathcal{R}(\varepsilon, D)$, the uncertainty resulting from model structural bias is reflected by the minimum achievable loss L^*. In reality, ML models can rarely attain a perfect loss (i.e., $L^* > 0$ holds for the best model $f_{best} \in \mathcal{R}(\varepsilon, D)$), as the limitations imposed by a type and its capacity prevent them from fully expressing the underlying GT, and thus from achieving a perfect Model GT agreement. Formally, this error (in the true explanations, i.e., regardless of a specific XAI method) resulting from the model's structural bias can be denoted by $\Delta_{2a} = \phi^*(x, f_{best}) - \phi^*(x, f^*)$.

Regardless of the structural bias, the second factor (2b) arises from the *estimation variance of the ML model*. Due to their finite and sparse nature, the available (training) data can only partially represent the underlying GT f^*, especially in high-dimensional feature spaces. Hence, different ML models (even of the same type) may fit the data equally well, forming the Rashomon set $\mathcal{R}(\varepsilon, D)$. In such cases, the learning algorithm exploits the remaining degrees of freedom when placing the DBs, leading to substantially different model behavior, especially in subsets of the feature space with fewer data points. From an explainability perspective, this second factor of model uncertainty implies that multiple, equally accurate ML models $f \in \mathcal{F}$ can yield (very) different explanations and FAs [12]. Formally, this error resulting from the estimation variance of an ML model $f \in \mathcal{F}$ can be defined by the difference to the true explanation of the best possible model f_{best}, i.e., $\Delta_{2b} = \phi^*(x, f) - \phi^*(x, f_{best})$. Thus, Δ_{2a} constitutes the ML model bias (inability to fit the GT), and Δ_{2b} constitutes the model variance (instability due to finite sampling).

Overall, the correctness of an explanation resp. its errors can be conceptualized as $\phi^m(x, f, Z) - \phi^*(x, f^*) = \Delta_{1a} + \Delta_{1b} + \Delta_{2a} + \Delta_{2b}$. Simple error aggregation risks masking distinct failure modes through cancellation effects (e.g., a biased surrogate coincidentally compensating for a biased model), i.e., $\|\Delta_{1a}\|, \|\Delta_{2a}\| > 0$ but $\|\Delta_{1a} + \Delta_{2a}\| \approx 0$. To ensure reliability in high-stakes settings, we must bound the worst-case error rather than relying on chance cancellations. Applying the triangle inequality yields a rigorous upper bound: $\|\phi^m(x, f, Z) - \phi^*(x, f^*)\| = \|\sum \Delta_i\| \leq \sum \|\Delta_i\|$. Consequently, satisfying explanation correctness requires minimizing the norm of each individual error term simultaneously.

4 Experiments

In the following, we describe the experimental setup designed to systematically investigate how the identified factors (1a)–(2b) affect the correctness of explanations. Our analysis comprises different dataset types, GT definitions, ML model types, and XAI methods, enabling a comprehensive analysis of these factors. In this paper, we focus on tabular datasets as they allow for more directly comparable FA explanations than image or textual data. Table 1 provides an overview of all evaluation setups and runs conducted.

Uncertainty	**Explanation Uncertainty**		**Model Uncertainty**	
Factor	(1a)	(1b)	(2a)	(2b)
Evaluation Setup	The internal (surrogate) model of the XAI method is altered to fit the underlying model. Runs are performed for ROLEX Regression, ROLEX Tree, Kernel SHAP, Tree SHAP, and Deep SHAP	The number of samples used by the XAI method is scaled. Runs are performed for 100, 1,000, and 10,000 samples for the XAI method.	The model type of the proxy GT and the trained model are altered. Runs are performed for real GT: GT-NN, GT-RF, for proxy GT: NN-NN, RF-NN, NN-RF, RF-RF	The number of samples used for the model training is scaled. Runs are performed for 2,000, 10,000, 100,000, 1,000,000 samples
XAI Methods	ROLEX, SHAP	LIME, ROLEX, SHAP, AGREE, AGGOPT	LIME, ROLEX, SHAP, AGREE, AGGOPT, EXPLORE, POIC	LIME, ROLEX, SHAP, AGREE, AGGOPT, EXPLORE, POIC
Metrics	Top 5 Kendall τ, Top 5 Intersection, Δ5%-ExpAgr	Top 5 Kendall τ, Top 5 Intersection, Δ5%-ExpAgr, Number of Unique Features	Top 5 Kendall τ, Top 5 Intersection, Δ5%-ExpAgr	Top 5 Kendall τ, Top 5 Intersection, Δ5%-ExpAgr

Table 1: Experiments Overview

4.1 Ideal Setting

Our experimental design follows an approach beginning with an *ideal setting*, in which the effects of factors (1a)–(2b) on explanation correctness are mitigated as much as possible, serving as a meaningful reference point. We establish this ideal setting as a prerequisite condition. Passing it is a necessary condition though not sufficient; XAI methods failing in this ideal setting, where both GT and models are known and controlled, cannot be trusted in uncontrolled real-

world settings. Moreover, this design enables ceteris paribus analyses, which are essential for sensitivity assessment and upper bound analysis. More precisely, in the ideal setting $\|\Delta_{1a}\| \approx 0 \wedge \|\Delta_{1b}\| \approx 0 \wedge \|\Delta_{2a}\| \approx 0 \wedge \|\Delta_{2b}\| \approx 0$ must hold. Starting from this, we systematically scale individual components to isolate and analyze the effect of each factor.

Focusing on the factors (1a)–(1b) (i.e., Δ_{1a}, Δ_{1b}), and thus explanation uncertainty, we must determine the actual, true FA explanations for a given explained ML model f and a test instance x. This true FA explanations must represent the true decision behavior of the ML model given by its DBs. Thus, we define the true FA explanation for x as the vector $\overrightarrow{x_{db} - x}$ pointing to the closest point x_{db} on a DB of f, i.e., $f(x) \neq f(x_{db}), \forall x' \in X: \|x' - x\| \leq \|x_{db} - x\| \Rightarrow f(x) = f(x')$ holds. When DBs are known, this notion is commonly regarded as the true explanation $\phi^*(x, f)$ in literature [21, 39, 44], and can also be used to validate XAI methods such as LIME and SHAP [15, 39], also cf. Appendix C. This true FAs can then be compared to and evaluated against the FAs provided by XAI methods. To measure this Explanation Model agreement, the faithfulness of FAs is typically assessed in literature expressing whether the relevant features and their attributions are correctly determined. We use three accuracy measures which are described in detail below: (Top k) Intersection, (Top k) Kendall τ and Δ5%-Explanation Agreement (ExpAgr) [1, 31, 38]. As both the type of the ML model being explained and the XAI method with, e.g., the type of the (surrogate) model it uses, are known, we can determine Δ_{1a} based on the XAI structural bias, i.e., the ability of the XAI method to reflect the true decision behavior of an ML model. Regarding the estimation variance of the XAI method in the ideal setting (factor (1b), Δ_{1b}), we can choose between an exhaustive but computationally expensive sampling of the neighborhood of a test instance to identify the closest DBs and the corresponding true FAs, or determining them via exact piecewise linear reconstruction of the ML model [8, 17, 23]. Since piecewise linear reconstruction extracts the exact DBs of piecewise linear models and thus inherently satisfies $\Delta_{1a} \approx 0$ and $\Delta_{1b} \approx 0$, we chose this approach to eliminate explanation uncertainty (perfect Explanation Model agreement in Figure 1), leading to $\Delta_{1a} \to 0 \wedge \Delta_{1b} \to 0$, providing the first basis for correct explanations.

Focusing on the factors (2a)–(2b) (i.e., Δ_{2a}, Δ_{2b}), and thus model uncertainty, training ML models that perfectly mimic the GT model f^* together with its DBs (which are typically unknown), poses a challenge widely acknowledged in literature [1, 21, 37, 45]. To overcome this, we instantiate the GT by an oracle function $f^*: X \to Y$ in such settings where the actual GT is unknown. Specifically, we train a 'proxy GT' model regarding it as the GT for all subsequent error calculations ($\|\phi - \phi^*_{proxy}\|$) (for details cf. Appendix B.2). In our experiments, we use an NN and a random forest (RF) as proxy GT model types. Using a proxy GT is methodologically essential to strictly decouple structural bias (Δ_{2a}) from variance (Δ_{2b}), which are inseparable in real-world data due to a mix of noise, lack of data, and model bias. Our controlled proxy GT allows us to strictly isolate the structural error Δ_{2a}. Regarding the factors (2a)–(2b) (i.e., Δ_{2a}, Δ_{2b}), we are now able to control both, the (dis)agreement of the GT model type versus the ML model type and the amount of information (i.e., instances) to train the ML model based on instances generated by the GT proxy model. To guarantee a high Model GT agreement and thus

$\Delta_{2a} \approx 0 \wedge \Delta_{2b} \approx 0$, we employ two accuracy measures: First, a local variant of the common accuracy as ratio of correctly classified samples near the test instances by an ML model. Second, the 'DB accuracy' metric measures for test instances whether the classes adjoining their closest DBs regarding the ML model being explained match those regarding the GT model (for more details and definitions of the metrics, cf. Appendix D). Given this, for the ideal setting, we first align the model types of the models being explained with the type of the underlying GT proxy model (e.g., both the GT and the models are NNs). This alignment ensures compatible model structure and expressiveness (factor 2a) relative to the GT model. Second, we train the models on a large dataset of up to 1,000,000 instances to ensure a low level of variance across the ML models (factor 2b), thereby minimizing model uncertainty. As a result of both $\Delta_{2a} \approx 0 \wedge \Delta_{2b} \approx 0$, the Rashomon set contains models, each with perfect DB accuracy and local accuracy scores of 1. In this ideal setting, model uncertainty is therefore eliminated, providing the second basis for assessing correct explanations.

On this basis, we determine all true FAs for the GT proxy models and all ML models in the Rashomon set in the ideal setting. In our experiments, we start from this and analyze each single factor (1a)–(2b) as outlined in Table 1.

4.2 Datasets

We select a diverse set of tabular datasets (cf. Table 2) covering real-world high-stakes domains (*Fetal Health, Compas, Prosper*), synthetic data (*Probabilistic*, following a generation procedure as described in [27]), and domain-law based data (*Pendulum, Inflation*) (cf. Appendix A for details on all datasets). For real-world and synthetic datasets, we employ a proxy GT (NN or RF) to strictly isolate error sources. For Pendulum and Inflation, the GT is defined by explicit physical or economic laws, allowing analyses without ML-induced proxy GTs. This combination ensures robustness across unknown and known data-generating processes. Overall, we are aware that these GT settings represent a more controlled and structured GT than any unknown data-generating process of real-world problems. However, as stated, if XAI methods fail to provide correct explanations even in these controlled (partly simpler) settings, their use in real-world settings would be questionable.

Dataset	**Fetal Health**	**Prosper**	**Compas**	**Probabilistic**	**Pendulum**	**Inflation**
Domain	Healthcare	Finance	Law	Synthetical	Physics	Economics
#Features	20	56	15	12	9	8
#Classes	3	3	3	3	2	3
Data Origin	Real World	Real World	Real World	Synthetical	Synthetical	Synthetical
GT Type	Proxy	Proxy	Proxy	Proxy	Real GT	Real GT

Table 2: Overview of Datasets used in our Experiments

4.3 Models, XAI Methods and Metrics

To analyze model uncertainty, we train a set of models by repeatedly bootstrapping 80% from a training set generated via a DB-focused sampling strategy and labeled according to the respective

GT (more details in Appendix B.1), which induces variability across ML models while maintaining similar performance. We train 200 models in this manner and evaluate only the top 30 models regarding DB accuracy as Rashomon set in our experiments. For the DB accuracy, we use a set of 100 test instances from the original dataset that were not used for GT or model training.

To assess the correctness of explanations, we apply a diverse set of XAI methods comprising both well-established FA methods and methods that are specialized to address uncertainty of explanations. We consider the following XAI methods, all of which were discussed in the related work section: LIME, ROLEX, SHAP, EXPLORE, AGREE, AGGOPT, and the Partial Order in Chaos (POIC) method [25], which is applicable for RFs. For each of the 30 models in the Rashomon set and the 100 test instances, we generate explanations using each of the considered XAI methods.

We evaluate correctness using three metrics (details in Appendix D.3): *Top-k Intersection* (measuring feature overlap), *Top-k Kendall* τ (measuring rank correlation of the most important features), and Δ*5%-ExpAgr* (a strict binary metric indicating if FA values deviate no more than ±2.5% from the GT). We set $k = 5$.

5 Results and Discussion

In this section, we discuss our experimental results based on the error decomposition introduced in Section 3. We analyze the individual impact of each factor of explanation uncertainty (Δ_{1a}, Δ_{1b}) and model uncertainty (Δ_{2a}, Δ_{2b}) on the correctness of FAs. This allows us to clarify the link between explanation consistency and correctness.

We begin our analysis with the ideal setting, which minimizes all error terms ($\Delta_i \approx 0$) to serve as a prerequisite condition. Precisely, in this setting, any deviation from the GT can be attributed solely to the inherent limitations of the XAI method itself.

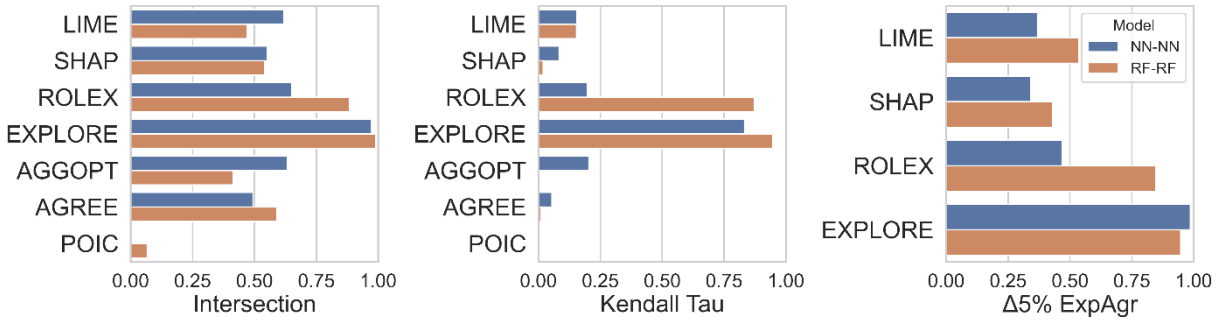


Figure 2: Explanation Correctness Results in the Ideal Setting

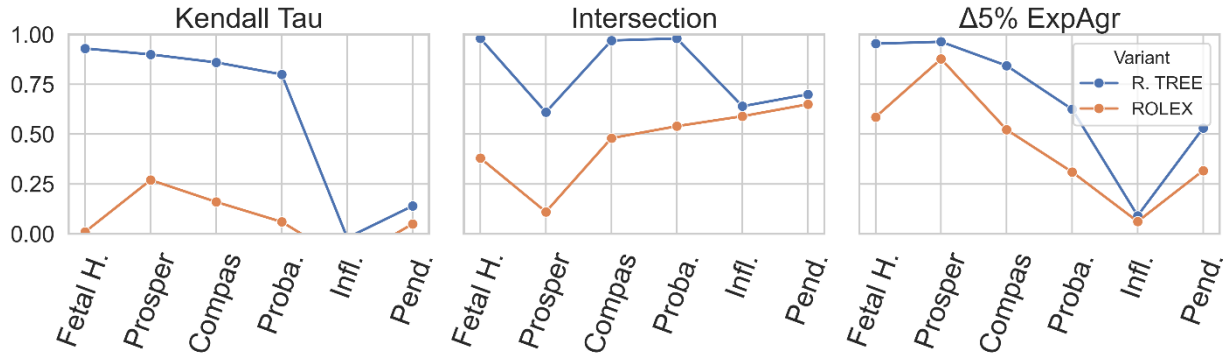


Figure 3: Explanation Correctness Results for Factor 1a

Figure 2 shows that EXPLORE achieves near-perfect scores (Intersection of 0.98, Kendall τ of 0.89, and Δ5%-ExpAgr of 0.97). This confirms that when explanation uncertainty is eliminated by design ($\Delta_{1a} = 0$ via functional extraction, Δ_{1b} via exact DBs) and model uncertainty is controlled ($\Delta_{2a}, \Delta_{2b} \approx 0$), correct explanations are indeed attainable. This serves as a crucial sanity check: deviations observed in subsequent experiments can thus be strictly attributed to each isolated factor Δ_i.

Importantly, the errors stemming from each factor can also be explicitly calculated for individual instances. For instance, the explanation for a test instance x_0 and a model f_{best} from the ideal setting of the Fetal Health dataset can be compared to the explanation obtained from a model f which was trained on a reduced (by 90%) set of the training data. Using the ℓ^∞-norm, we observe a maximum attibution deviation of $\|\Delta_{2b}\|_\infty = \max|\phi^*(f, x_0) - \phi^*(f_{best}, x_0)| = 0.12$ for x_0 that can be solely attributed to factor (2b).

Next, we discuss ROLEX and its variant ROLEX Tree, which employs surrogate DTs to better align with the tree-based structure of RFs as underlying ML models. In this case, the resulting explanation correctness remains high, with an average Intersection of 0.96, a Kendall τ of 0.80, and a Δ5%-ExpAgr of 0.85. Given the ideal setting ($\Delta_{2a}, \Delta_{2b} \approx 0$), the performance gap to all remaining XAI methods (cf. below) is driven by $\Delta_{1a} \to 0$: the tree-based surrogate locally aligns with the underlying RF model, whereas linear surrogates (LIME, SHAP, standard ROLEX) suffer from a mismatch in expressiveness. The remaining performance gap of ROLEX Tree to EXPLORE is due to the fact that ROLEX uses sampling to generate its surrogate, negatively impacting Δ_{1b}.

All other XAI methods struggle significantly with respect to correctness, even in the ideal setting: While the average Intersection of 0.49 remains acceptable, the Kendall τ of 0.06 and the Δ5%-ExpAgr of 0.42 degrade substantially (cf. Appendix E for the complete results). Since $\Delta_{1b}, \Delta_{2a}, \Delta_{2b} \approx 0$ holds in this setting, these errors are driven almost exclusively by Δ_{1a}. This validates that the structural bias of XAI methods relying on linear surrogates (e.g., LIME) is insufficient to capture the *local* DBs of models like (partly simpler) NNs or RFs correctly, even when provided with optimal conditions. As in reality such conditions are never met, explanation correctness may degrade even further in real XAI applications, where the GT is unknown and does not follow a model type behavior. This shows that explanation uncertainty (given by the

factors Δ_{1a}, Δ_{1b}) critically prevents many XAI methods, even those that aim to address uncertainty explicitly, from achieving their goals of providing correct explanations.

5.1 Factor 1a: XAI Structural Bias

Starting from the ideal setting, we first examine the effect of factor (1a) concerning the structural bias of the XAI method. To isolate and validate the impact of factor (1a), we evaluate ROLEX Tree against the original regression-based ROLEX variant. The results shown in Figure 3 demonstrate that, when explaining RFs, this tree-based variant achieves substantially higher explanation correctness than the linear-surrogate default, with average Intersection, Kendall τ, and Δ5%-ExpAgr scores increasing by 0.58, 0.68, and 0.27 respectively (for the detailed results per dataset and for the analysis of SHAP variants as described in Table 1, cf. Appendix E). This significant gap implies that explanation correctness is strictly capped by the structural bias of the XAI method. As the number of samples are equal for both ROLEX variants, the error Δ_{1a} becomes the dominant term when the surrogate model type (e.g., linear regression) structurally contradicts the explained model type (e.g., RF). We denote this as a *methodological mismatch*. Crucially, our results show that aggregation-based methods (AGGOPT, AGREE) cannot compensate for this intrinsic deficit. This reveals a fundamental limitation of aggregation methods: Aggregation may reduce variance (Δ_{1b}) but solidifies structural bias (Δ_{1a}). Combining multiple flawed approximations yields a stable but consistently incorrect explanation. Since they aggregate explanations that all share the same structural bias (high Δ_{1a}), they merely stabilize an incorrect approximation rather than correcting it. Consequently, increasing methodological expressiveness—as demonstrated by the specialized ROLEX tree—is a prerequisite for correctness that cannot be bypassed by combining structurally flawed explanations.

Overall, these findings indicate that factor (1a) substantially influences explanation correctness and, more importantly, raise important questions regarding the applicability of surrogate-based explanation methods to more complex model types. In particular, increasing methodological expressiveness often comes at the cost of interpretability, highlighting a known fundamental trade-off. However, when somewhat more complex black-box models are employed, the required level of methodological expressiveness cannot be achieved by an interpretable XAI method.

5.2 Factor 1b: XAI Estimation Variance

Next, we analyze how the estimation variance of the XAI method affects explanation correctness. Specifically, we evaluate LIME, ROLEX, SHAP, AGREE, and AGGOPT under varying amounts of provided samples and perturbations. Prior work has established that a sufficiently large number of samples is crucial for explanation consistency [4, 46]. To assess whether this effect translates to explanation correctness, we examine the results shown in Figure 4.

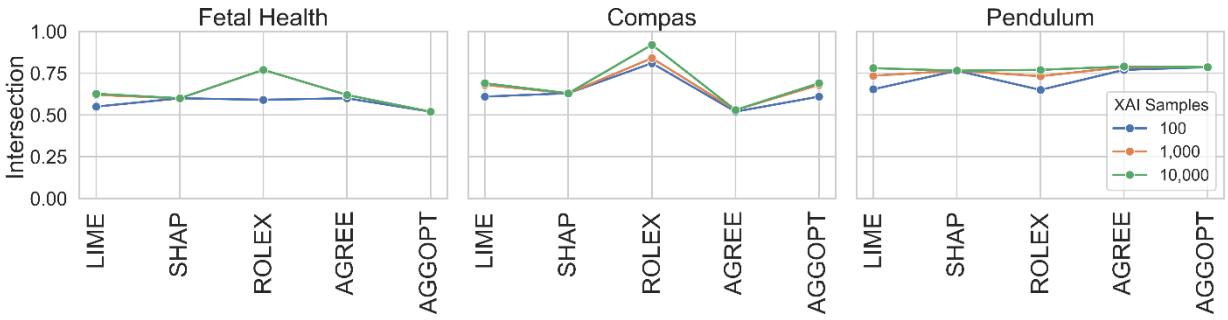


Figure 4: Explanation Correctness Results for Factor 1b

Regarding correctness, the methods achieve an Intersection of 0.52, a Kendall τ of 0.07, and a Δ5%-ExpAgr of 0.36 on average for 100 samples, which only slightly change by 0.08, -0.01, and 0.03 respectively for 10,000 samples. These findings suggest that while $\Delta_{1b} \rightarrow 0$ (via increased sampling) improves consistency, it does not rectify the structural errors introduced by Δ_{1a}. We observe a phenomenon we term the 'Consistency Trap': With increasing sample size, explanations converge to a stable value (low variance due to Δ_{1b}), leading users to trust them. However, as shown by stagnant correctness metrics, they often converge to a systematically incorrect explanation driven by the unaddressed structural bias (Δ_{1a}). This proves that consistency is a necessary but insufficient condition for correctness. Relying solely on the stability of an explanation across repeated runs ($\Delta_{1b} \rightarrow 0$) masks the potential magnitude of the structural error Δ_{1a}, creating a false sense of security in high-stakes applications.

5.3 Factor 2a: ML Model Structural Bias

Focusing on model uncertainty, we now analyze the agreement of model types (2a) between the GT model and the ML model being explained. The corresponding results for Δ5%-ExpAgr are presented in Figure 5 (for the complete results, cf. Appendix E).

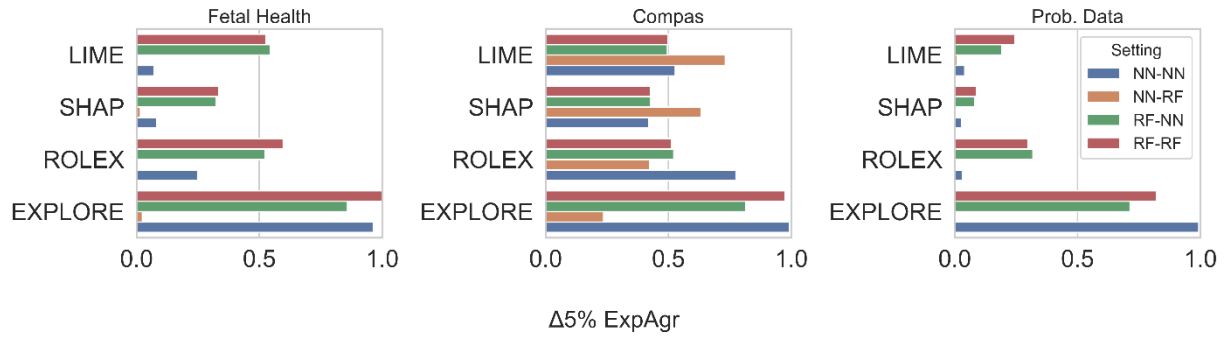


Figure 5: Explanation Correctness Results for Factor 2a

In cases where the GT and ML model types agree (NN-NN and RF-RF, i.e., $\Delta_{2a}, \Delta_{2b} \approx 0$ holds and the XAI methods are only exposed to a certain degree of explanation uncertainty (Δ_{1a}, Δ_{1b}) an average Intersection of 0.54, a Kendall τ of 0.28, and a Δ5%-ExpAgr of 0.58 is achieved. In contrast, model type disagreement (i.e., only Δ_{2a} changes) substantially decreases results by 0.10, 0.08, and 0.12 respectively. Interestingly, we observe a strong asymmetry: RF-NN explanations are nearly twice as accurate as in NN-RF (0.58 vs. 0.34 Δ5%-ExpAgr), while their Intersection and Kendall τ scores remain comparable. This indicates that while both capture the

general DB direction, NNs—due to fewer structural constraints and thus higher expressiveness—better approximate the local DBs of an RF GT. Conversely, RFs are constrained to axis-aligned splits, leading to a high structural error Δ_{2a} when the GT is complex (as with NNs). This asymmetry reveals that the structural bias of the ML model Δ_{2a} is a decisive factor. Crucially existing methods that aggregate multiple explanations typically do so either from a single model (e.g., AGREE, AGGOPT) or from multiple models of the same type (e.g., POIC), and therefore do not account for the impact of differences in model expressivity on explanation correctness.

Overall, our findings have severe implications for 'model-agnostic' XAI and in particular approaches aggregating multiple 'model-agnostic' explanations. While these XAI methods claim to explain *any* ML model, they cannot cope with the ML model's inability to capture the GT. If the chosen model class lacks the expressiveness to represent the underlying causal structure ($\Delta_{2a} \gg 0$), even a perfect explanation of the model ($\Delta_{1a/b} = 0$) will fail to describe the true feature importances of the real-world phenomenon. Thus, the choice of the ML model places an upper bound on the achievable correctness of any subsequent explanation. This is particularly concerning because in real-world XAI applications, users typically rely on explanations without being aware of how strongly they depend on the chosen model class. Explanations are often used to interpret or justify a given ML model's behavior, regardless of how well it reflects the GT which can even lead to explanations that contradict established domain knowledge, e.g., in healthcare [22].

5.4 Factor 2b: ML Model Estimation Variance

Finally, we examine factor (2b) regarding the estimation variance of ML models by systematically scaling the size of the training dataset. The results are illustrated in Figure 6.

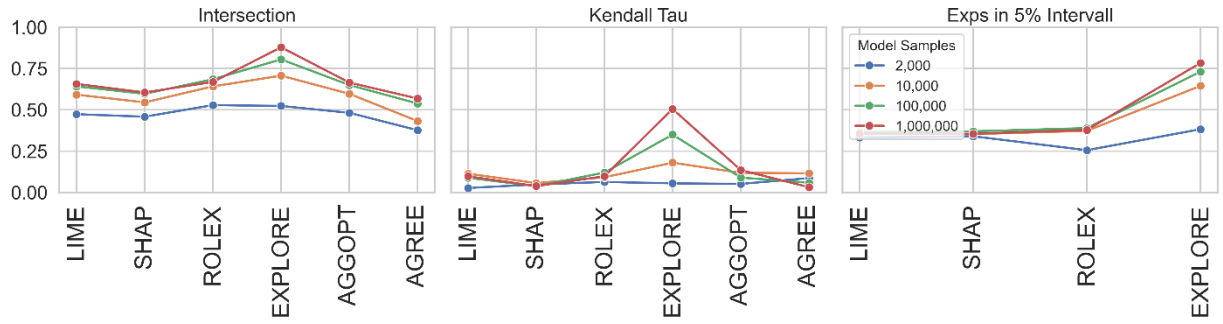


Figure 6: Explanation Correctness Results for Factor 2b

Upon reducing the number of data samples from 10,000 to 2,000 (i.e., for $\Delta_{2b} \gg 0$), for EXPLORE (with $\Delta_{1a}, \Delta_{1b}, \Delta_{2a} = 0$ holding) the correctness is severely compromised with a substantial decrease in the average Δ5%-ExpAgr from 0.68 to 0.49. This pronounced decline indicates that this reduction in training data induces substantial changes in the ML models' local decision behavior, which are accurately reflected by EXPLORE. This is highly relevant for real-world applications of ML and XAI, where training data are often sparse and hard to obtain. Interestingly, LIME, ROLEX, and SHAP retain largely stable metrics (decreasing only by 0.06, 0.00, and 0.01 respectively) despite the severe degradation of the underlying ML model due to

data scarcity (for $\Delta_{2b} \gg 0$). While this stability is often interpreted as an asset in XAI evaluation, our results expose it as a lack of sensitivity. These methods fail to reflect the shift in the model's DBs, effectively hallucinating stability where there is none. This reveals a critical 'Stability-Sensitivity Trade-off': XAI methods often mask model uncertainty, whereas correct explanations must inevitably fluctuate when the underlying model becomes unstable due to data scarcity. This impact of limited training data persists for methods such as AGGOPT, AGREE, and POIC which explicitly aim to address model uncertainty. Despite their focus, their average Intersection and Kendall τ results drop only by 0.03 and 0.00 respectively, showing similar behavior as other, non-specialized XAI methods.

6 Conclusion

In this work, we challenged the prevailing assumption that consistent explanations imply correctness. By formally decomposing the explanation error into explanation uncertainty (Δ_1) and model uncertainty (Δ_2), we demonstrated that correctness is strictly bound by the sum of structural biases and estimation variances: $\|\phi - \phi^*\| \leq \sum\|\Delta_{bias}\| + \sum\|\Delta_{variance}\|$. Our extensive experimental analysis across diverse datasets and GT settings leads to three critical conclusions:

First, current XAI suffers from a ‘Consistency Trap’. Researchers and practitioners often equate the reduction of estimation variance ($\Delta_{1b} \to 0$, e.g., by increasing samples) with correctness. We show that without addressing the structural bias of the XAI method (Δ_{1a}), this merely results in consistently incorrect explanations. A linear surrogate explaining a non-linear phenomenon will converge to a biased estimate, no matter how many samples are used.

Second, we postulate the need for 'Methodological Alignment'. Our results regarding tree-based surrogates demonstrate that the structural bias (Δ_{1a}) can only be minimized if the inductive bias of the XAI method matches the model class. For instance, ‘model-agnostic’ approaches relying on linear surrogates are inherently flawed for more complex non-linear models, necessitating a shift towards model-specific or structurally adaptive XAI methods.

Third, we identify a critical ‘Stability-Sensitivity Trade-off’. While stability is generally a desired property, we show that widely used methods like LIME, SHAP, and in particular XAI methods specialized on model uncertainty achieve stability at the cost of sensitivity. When the underlying ML model degrades due to data scarcity (Δ_{2b}), correct explanations *should* fluctuate to reflect model instability. Methods that remain ‘stable’ in these conditions essentially hallucinate reliability, masking model uncertainty.

Fourth, the ML Model structural bias (Δ_{2a}) sets the hard upper bound for correctness. We demonstrated that ‘model-agnostic’ XAI is a fallacy when the model class cannot capture the GT. Even an ideal XAI method ($\Delta_1 = 0$) cannot provide a correct explanation of the real-world phenomenon if the underlying ML model is structurally deficient.

Regarding implications and future work, the presented findings necessitate a shift in XAI evaluation: moving from variance-based metrics (consistency/stability) to bias-aware metrics (considering GT). While our study focused on tabular data and specific model classes (RF, NN) to

ensure strict comparability, the identified structural trade-offs are fundamental to approximation theory and likely extend to other domains. Future work should address the development of 'bias-aware' XAI methods that can quantify their own structural limitations and warn users when the explanation model lacks the expressiveness to capture the underlying DB.

7 References

[1] Chirag Agarwal, Satyapriya Krishna, Eshika Saxena, Martin Pawelczyk, Nari Johnson, Isha Puri, Marinka Zitnik, and Himabindu Lakkaraju. 2022. OpenXAI: Towards a transparent evaluation of model explanations. In *Advances in Neural Information Processing Systems*.

[2] Shamima Ahmed, Muneer M. Alshater, Anis E. Ammari, and Helmi Hammami. 2022. Artificial intelligence and machine learning in finance: A bibliometric review. *Research in International Business and Finance* 61 (2022), 101646. DOI: https://doi.org/10.1016/j.ribaf.2022.101646.

[3] David Alvarez-Melis and Tommi Jaakkola. 2018. On the robustness of interpretability methods. In *Proceedings of the ICML Workshop on Human Interpretability in Machine Learning*, 66–71.

[4] David Alvarez-Melis and Tommi Jaakkola. 2018. Towards robust interpretability with self-explaining neural networks. In *Advances in Neural Information Processing Systems*.

[5] Shahin Atakishiyev, Mohammad Salameh, Hengshuai Yao, and Randy Goebel. 2024. Explainable artificial intelligence for autonomous driving: A comprehensive overview and field guide for future research directions. *IEEE Access* 12 (2024), 101603–101625. DOI: https://doi.org/10.1109/ACCESS.2024.3431437.

[6] Leo Breiman. 2001. Random Forests. *Machine Learning* 45, 1 (2001), 5–32. DOI: https://doi.org/10.1023/A:1010933404324.

[7] Mustafa Cavus and Przemysław Biecek. 2025. Investigating the impact of balancing, filtering, and complexity on predictive multiplicity: A data-centric perspective. *Information Fusion* 123 (2025), 103243. DOI: https://doi.org/10.1016/j.inffus.2025.103243.

[8] Lingyang Chu, Xia Hu, Juhua Hu, Lanjun Wang, and Jian Pei. 2018. Exact and consistent interpretation for piecewise linear neural networks. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, New York, NY, 1244–1253.

[9] Alexander D'Amour, Katherine Heller, Dan Moldovan, Ben Adlam, Babak Alipanahi, Alex Beutel, Christina Chen, Jonathan Deaton, Jacob Eisenstein, Matthew D. Hoffman, Farhad Hormozdiari, Neil Houlsby, Shaobo Hou, Ghassen Jerfel, Alan Karthikesalingam, Mario Lucic, Yian Ma, Cory McLean, Diana Mincu, Akinori Mitani, Andrea Montanari, Zachary Nado, Vivek Natarajan, Christopher Nielson, Thomas F. Osborne,

Rajiv Raman, Kim Ramasamy, Rory Sayres, Jessica Schrouff, Martin Seneviratne, Shannon Sequeira, Harini Suresh, Victor Veitch, Max Vladymyrov, Xuezhi Wang, Kellie Webster, Steve Yadlowsky, Taedong Yun, Xiaohua Zhai, and d. Sculley. 2022. Underspecification presents challenges for credibility in modern machine learning. *Journal of Machine Learning Research* 23, 226 (2022), 1–61.

[10] Thomas Decker, Ananta R. Bhattarai, Jindong Gu, Volker Tresp, and Florian Buettner. 2024. *Provably better explanations with optimized aggregation of feature attributions*. arXiv:2406.05090 [cs.LG].

[11] Ann-Kathrin Dombrowski, Maximillian Alber, Christopher Anders, Marcel Ackermann, Klaus-Robert Müller, and Pan Kessel. 2019. Explanations can be manipulated and geometry is to blame. In *Advances in Neural Information Processing Systems*, 13589–13600.

[12] Jiayun Dong and Cynthia Rudin. 2020. Exploring the cloud of variable importance for the set of all good models. *Nature Machine Intelligence* 2, 12 (2020), 810–824. DOI: https://doi.org/10.1038/s42256-020-00264-0.

[13] EU AI Act. 2025. *The EU Artificial Intelligence Act* (2025). Retrieved January 14th, 2026 from https://artificialintelligenceact.eu/.

[14] Aaron Fisher, Cynthia Rudin, and Francesca Dominici. 2019. All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously. *Journal of Machine Learning Research* 20, 177 (2019), 1–81.

[15] Riccardo Guidotti. 2021. Evaluating local explanation methods on ground truth. *Artificial Intelligence* 291 (2021), 103428. DOI: https://doi.org/10.1016/j.artint.2020.103428.

[16] Tianjian Guo, Indranil R. Bardhan, Ying Ding, and Shichang Zhang. 2025. An explainable artificial intelligence approach using graph learning to predict intensive care unit length of stay. *Information Systems Research* 36, 3 (2025), 1478–1501.

[17] Bernd Heinrich, Thomas Krapf, and Paul Miethaner. 2024. EXPLORE: A novel method for local explanations. In *Proceedings of the 45th International Conference on Information Systems*.

[18] Hsiang Hsu and Flavio Calmon. 2022. Rashomon capacity: A metric for predictive multiplicity in classification. In *Advances in Neural Information Processing Systems*, 28988–29000.

[19] Xuanxiang Huang and Joao Marques-Silva. 2024. On the failings of Shapley values for explainability. *International Journal of Approximate Reasoning* 171 (2024), 109112.

[20] Eyke Hüllermeier and Willem Waegeman. 2021. Aleatoric and epistemic uncertainty in machine learning: an introduction to concepts and methods. *Machine Learning* 110, 3 (2021), 457–506. DOI: https://doi.org/10.1007/s10994-021-05946-3.

[21] Yunzhe Jia, James Bailey, Kotagiri Ramamohanarao, Christopher Leckie, and Michael E. Houle. 2019. Improving the quality of explanations with local embedding perturbations. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, New York, NY, 875–884.

[22] Buomsoo Kim, Karthik Srinivasan, Sung H. Kong, Jung H. Kim, Chan S. Shin, and Sudha Ram. 2023. ROLEX: A novel method for interpretable machine learning using robust local explanations. *MIS Quarterly* 47, 3 (2023), 1303–1332. DOI: https://doi.org/10.25300/MISQ/2022/17141.

[23] Thomas Krapf, Michael Hagn, Paul Miethaner, Alexander Schiller, Lucas Luttner, and Bernd Heinrich. 2024. Piecewise linear transformation – Propagating aleatoric uncertainty in neural networks. In *Proceedings of the 38th AAAI Conference on Artificial Intelligence*, 20456–20464.

[24] Satyapriya Krishna, Tessa Han, Alex Gu, Steven Wu, Shahin Jabbari, and Himabindu Lakkaraju. 2022. *The disagreement problem in explainable machine learning: A practitioner's perspective*. arXiv:2202.01602 [cs.LG].

[25] Gabriel Laberge, Yann Pequignot, Alexandre Mathieu, Foutse Khomh, and Mario Marchand. 2023. Partial order in chaos: Consensus on feature attributions in the Rashomon set. *Journal of Machine Learning Research* 24, 364 (2023), 1–50.

[26] Thibault Laugel, Xavier Renard, Marie Lesot, Christophe Marsala, and Marcin Detyniecki. 2018. Defining locality for surrogates in post-hoc interpretablity. In *Proceedings of the ICML Workshop on Human Interpretability in Machine Learning*, 47–53.

[27] Yang Liu, Sujay Khandagale, Colin White, and Willie Neiswanger. 2021. Synthetic benchmarks for scientific research in explainable machine learning. In *35th Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

[28] Scott M. Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*, 30, 4768–4777.

[29] Charles Marx, Youngsuk Park, Hilaf Hasson, Yuyang Wang, Stefano Ermon, and Luke Huan. 2023. But are you sure? An uncertainty-aware perspective on Explainable AI. In *Proceedings of the 26th International Conference on Artificial Intelligence and Statistics*. Proceedings of Machine Learning Research. PMLR, 7375–7391.

[30] Nachaat Mohamed. 2023. Current trends in AI and ML for cybersecurity: A state-of-the-art survey. *Cogent Engineering* 10, 2 (2023), 2272358. DOI: https://doi.org/10.1080/23311916.2023.2272358.

[31] Catarina Moreira, Yu-Liang Chou, Chihcheng Hsieh, Chun Ouyang, João Pereira, and Joaquim Jorge. 2025. Benchmarking instance-centric counterfactual algorithms for XAI: From white box to black box. *ACM Comput. Surv.* 57, 6 (2025). DOI: https://doi.org/10.1145/3672553.

[32] Meike Nauta, Jan Trienes, Shreyasi Pathak, Elisa Nguyen, Michelle Peters, Yasmin Schmitt, Jörg Schlötterer, Maurice van Keulen, and Christin Seifert. 2023. From anecdotal evidence to quantitative evaluation methods: A systematic review on evaluating explainable AI. *ACM Comput. Surv.* 55, 13s (2023), 1–42.

[33] Bitya Neuhof and Yuval Benjamini. 2024. Confident feature ranking. In *Proceedings of the 27th International Conference on Artificial Intelligence and Statistics*. Proceedings of Machine Learning Research. PMLR, 1468–1476.

[34] Craig Pirie, Nirmalie Wiratunga, Anjana Wijekoon, and Carlos F. Moreno-Garcia. 2023. AGREE: A feature attribution aggregation framework to address explainer disagreements with alignment metrics. In *CEUR Workshop Proceedings*.

[35] Gregory Plumb, Denali Molitor, and Ameet S. Talwalkar. 2018. Model agnostic supervised local explanations. In *Advances in Neural Information Processing Systems*, 31.

[36] Stephan Raaijmakers. 2019. Artificial intelligence for law enforcement: Challenges and opportunities. *IEEE Security & Privacy* 17, 5 (2019), 74–77. DOI: https://doi.org/10.1109/MSEC.2019.2925649.

[37] Kaivalya Rawal, Zihao Fu, Eoin Delaney, and Chris Russell. 2025. Evaluating model explanations without ground truth. In *Proceedings of the 2025 ACM Conference on Fairness, Accountability, and Transparency*. FAccT '25. ACM, New York, NY, 3400–3411. DOI: https://doi.org/10.1145/3715275.3732219.

[38] Marco T. Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "Why should I trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '16. ACM, New York, NY, 1135–1144. DOI: https://doi.org/10.1145/2939672.2939778.

[39] Andrew S. Ross, Michael C. Hughes, and Finale Doshi-Velez. 2017. Right for the right reasons: Training differentiable models by constraining their explanations. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*. IJCAI'17. AAAI Press, 2662–2670.

[40] Jonas Schulz, Raul Santos-Rodriguez, and Rafael Poyiadzi. 2022. Uncertainty quantification of surrogate explanations: An ordinal consensus approach. In *Proceedings of the Northern Lights Deep Learning Workshop.* DOI: https://doi.org/10.7557/18.6294.

[41] Torgyn Shaikhina, Umang Bhatt, Roxanne Zhang, Georgatzis. Konstantinos, Alice Xiang, and Adrian Weller. 2021. Effects of uncertainty on the quality of feature

importance explanations. In *Proceedings of the AAAI Workshop on Explainable Agency in Articial Intelligence*.

[42] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning*. Proceedings of Machine Learning Research, 3319–3328.

[43] G. Visani, E. Bagli, F. Chesani, A. Poluzzi, and D. Capuzzo. 2022. Statistical stability indices for LIME: Obtaining reliable explanations for machine learning models. *Journal of the Operational Research Society* 73, 1 (2022), 91–101.

[44] Zifan Wang, Matt Fredrikson, and Anupam Datta. 2022. Robust models Are more interpretable because attributions look normal. In *Proceedings of the 39th International Conference on Machine Learning*. Proceedings of Machine Learning Research. PMLR, 22625–22651.

[45] Fan Yang, Mengnan Du, and Xia Hu. 2019. *Evaluating explanation without ground truth in interpretable machine learning*. arXiv:1907.06831 [cs.LG].

[46] Han Yuan, Mingxuan Liu, Lican Kang, Chenkui Miao, and Ying Wu. 2023. *An empirical study of the effect of background data size on the stability of SHapley Additive exPlanations (SHAP) for deep learning models*. arXiv:2204.11351 [cs.LG].

[47] Yilun Zhou, Serena Booth, Marco T. Ribeiro, and Julie Shah. 2022. Do feature attribution methods correctly attribute features? *Proceedings of the AAAI Conference on Artificial Intelligence* 36, 9 (2022), 9623–9633. DOI: https://doi.org/10.1609/aaai.v36i9.21196.

[48] Zhengze Zhou, Giles Hooker, and Fei Wang. 2021. S-LIME: Stabilized-LIME for model explanation. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. ACM, New York, NY.

8 Appendix

8.1 Appendix A: Datasets

For our experiments, we deliberately select a diverse set of tabular datasets to assess the observed effects across a wide range of domains, levels of problem complexity, real-world and synthetic settings, and GT types (cf. Table 2). First, we consider real-world datasets from high-stakes domains for which XAI methods are particularly relevant and frequently applied. We employ the Fetal Health dataset from the healthcare domain, containing cardiotocography measurements used to classify fetal health conditions into normal, suspect, and pathological categories. Second, we use the Compas criminal recidivism dataset, which comprises demographic and criminal history information to predict the risk of reoffending across a scale of three outcome classes. Third, we include the Prosper dataset including information on loan applicants, based on which the default risk of peer-to-peer loan applications is predicted. Further, we consider the fully synthetic

'Probabilistic' dataset, which provides a controlled setting with a known data-generating distribution, free from unknown confounding factors or data quality issues that typically arise in real-world data. Following [27], the dataset is generated by sampling from a Gaussian mixture model with three mixture components, where each component defines a class label (see details below).

For these datasets, the GT is not (fully) known and we therefore rely on an ML model as a proxy GT in these cases. To broaden our evaluation beyond such proxy-based GTs, we additionally consider datasets governed by well-established laws from physics and economics. In these settings, the GT is explicitly defined and functionally describable, eliminating the need for a proxy GT and enabling a complementary analysis under known, non-ML-induced GTs. For this, we consider the pendulum law and create a dataset capturing the current state of a pendulum, with binary labels indicating whether the system comes to rest within a predefined time horizon (see details below). Further, we use an economics-based dataset consisting of inflation, interest rates, and other macroeconomic indicators of a country at a given point in time. Labels are assigned according to the Taylor rule, which indicates whether an expansionary or contractionary monetary policy stance is implied.

Overall, we are aware that these GT settings represent a more controlled and structured GT than any unknown data-generating process of real-world problems. However, we argue that if XAI methods fail to provide correct explanations even in these controlled (partly simpler) settings, their performance can be assumed to be at least as limited in further real-world scenarios.

We applied the following preprocessing steps to all datasets: We removed all date-related or identifier variables, as well as features with excessive missingness. All features of the original datasets are scaled to the unit interval $[0,1]$ using min–max normalization.

For the Fetal Health dataset (available at https://www.kaggle.com/datasets/andrewmvd/fetal-health-classification), we applied a second preprocessing step by selecting features to improve distributional stability of their value ranges. Specifically, the feature 'histogram_tendency' was remapped to fixed representative values (0.45, 0.5, 0.55) and the two features 'prolongued_decelerations' and 'histogram_number_of_zeros' with long-tails in their distribution were linearly rescaled to ensure comparable magnitudes. Finally, 'severe_decelerations' was identified as unstable and therefore was removed, resulting in a cleaned feature set that was used in all subsequent experiments.

For the Prosper dataset (available at https://www.kaggle.com/datasets/henryokam/prosper-loan-data), the target variable 'LoanDefault' was constructed from the original loan status and estimated loss, yielding a three-class outcome. Remaining categorical features were transformed using ordinal encoding, missing values were imputed via k-nearest-neighbor imputation ($k = 5$).

For the Compas dataset (available at https://www.kaggle.com/datasets/danofer/compass), we derived a three-class target variable from the values of 'violence_risk' by discretizing the original violence decile score into low-, medium-, and high-risk categories. Categorical attributes were

encoded using numeric category codes, missing values were imputed via k-nearest-neighbor imputation ($k = 10$).

The synthetic Probabilistic dataset was generated in two steps: (i) sampling the feature vectors from a probabilistic generative model, and (ii) assigning class labels based on the generative component of the features. More precisely, the first step was conducted as follows: Let d denote the feature dimension and K the number of mixture components. Each data point $x \in \mathbb{R}^d$ is drawn from a Gaussian mixture model

$$p(x) = \frac{1}{K}\sum_{k=1}^{K} \mathcal{N}(x|\mu_k \Sigma_k)$$

where $\mu_k \in \mathbb{R}^d$ and $\Sigma_k \in \mathbb{R}^d$ are the mean vector and covariance matrix of the k-th component, respectively.

In our experiments, all mixture components share the same covariance structure, $\forall k: \Sigma_k = \Sigma$ with $\Sigma = (1-\rho)I_d + \rho 11^{\mathrm{T}}$ where I_d is the d-dimensional identity matrix, 1 is the vector of ones, and $\rho \in [0, 1)$ controls the correlation between any pair of distinct features. Hence, each feature has unit variance, and every pair of features has correlation ρ. The means $\{\mu_k\}_{k=1}^{K}$are constructed to be well separated but not orthogonal. First, random directions $v_k \in \mathbb{R}^d$ are sampled $v_k \sim \mathcal{N}(0, I_d)$ and normalized. Then, each mean vector is obtained as $\mu_k = \alpha v_k + \varepsilon_k$, where, $\alpha > 0$ is a scaling factor and $\varepsilon_k \sim \mathcal{N}(0, \sigma_\mu^2 I_d)$ is an isotropic jitter term that increases overlap between mixture components and prevents trivial class separation.

To generate a dataset of size N, we repeat the following steps for each sample $i = 1, \dots, N$:

1. Draw a mixture index $k_i \sim Uniform\{1, \dots, K\}$
2. Draw a feature vector $x_i \sim \mathcal{N}(\mu_k, \Sigma)$
3. Assign the class label $y_i = k_i$

Thus, the resulting dataset $\{(x_i, y_i)\}_{i=1}^{N}$ corresponds to a balanced multi-class classification problem generated by a known Gaussian mixture model GT.

The Pendulum dataset is a synthetically generated binary classification dataset based on the numerical simulation of a damped physical pendulum. Each data point represents a distinct dynamical system configuration and is labeled according to whether the system returns to equilibrium within a fixed time horizon.

The pendulum dynamics are governed by the second-order ordinary differential equation

$$\ddot{\theta}(t) + b_{eff}\dot{\theta}(t) + \frac{g}{L}\sin\big(\theta(t)\big) = 0$$

where $\theta(t)$ denotes the angular displacement, L the pendulum length, g the gravitational constant, and b_{eff} an effective damping coefficient. The effective damping is defined as

$$b_{eff} = b + 0.01\rho_{air} + f_{pivot}$$

combining intrinsic damping, air density effects and pivot friction. The system is numerically integrated over a finite time interval $[0, T]$, starting from initial conditions (θ_0, ω_0), where ω_0 denotes the initial angular velocity. Each sample is defined by the following nine continuous input variables, independently drawn from uniform distributions over predefined ranges: pendulum length $L \in [0.5, 2.0]$, pendulum mass $m \in [0.1, 5.0]$, initial angular displacement $\theta_0 \in [-\frac{\pi}{4}, \frac{\pi}{4}]$, initial angular velocity $\omega_0 \in [-1.0, 1.0]$, damping coefficient $b \in [0.01, 0.2]$, gravitational acceleration $g \in [9.7, 9.9]$, air density $\rho_{air} \in [1.0, 1.3]$, pivot friction $f_{pivot} \in [0.0, 0.5]$, simulation time horizon $T \in [5.0, 15.0]$.

Each simulated trajectory is assigned a binary label based on whether the pendulum returns sufficiently close to equilibrium at the final time T. Specifically, an instance is labeled as stable if $|\theta(T) < \varepsilon_\theta|$ and $|\omega(T) < \varepsilon_\omega|$, where fixed thresholds $\varepsilon_\theta = 0.1$ and $\varepsilon_\omega = 0.05$ define angular and velocity tolerances. All other trajectories are labeled as non-stable. The dataset is generated in a class-balanced manner, with an equal number of samples per class obtained.

The Inflation dataset is a synthetically generated multi-class classification dataset inspired by a stylized Taylor-rule–based monetary policy framework. Each data point represents a hypothetical macroeconomic scenario characterized by inflation dynamics, output levels, and policy parameters, and is assigned to one of three policy regimes.

For each sample, the following continuous variables are independently drawn from uniform distributions over predefined ranges: inflation rate $\pi \in [0, 10]$, economic output $y \in [80, 120]$, real interest rate $r \in [0, 5]$, inflation target $\pi^* \in [1.5, 2.5]$, natural real rate $r^* \in [0.5, 2.0]$, potential output $y^* \in [95, 105]$. From these variables, two derived quantities are computed: $GDP\ gap = y - y^*$ and a policy interest rate determined by the Taylor-Rule,

$$i = r + \pi + 0.5(\pi - \pi^*) + 0.5 * \frac{y - y^*}{y^*} * 100,$$

which is subsequently clipped to the interval [0, 20]. The final feature vector thus consists of eight attributes:

$$(\pi, y, r, \pi^*, y^*, r^*, GDP\ gap, i)$$

Each instance is assigned to one of three policy classes based on the deviation of the computed policy rate i from the expected nominal rate $r + \pi$: Aggressive policy (class 1) if $i - (r + \pi) > 2$, passive policy (class 2) if $i < (r + \pi)$, normal policy (class 0) otherwise.

For the datasets with real GT, GT explanations are constructed by explicitly leveraging the known physical or analytical data generation process underlying each dataset. Given a normalized input instance $x \in [0,1]^d$, we first map it back to the original parameter space using min–max denormalization, yielding $\tilde{x} \in X$. The class label $\tilde{y}$ is then obtained by evaluating the governing system model, i.e., by numerically solving the underlying physical dynamics or analytical policy equations, depending on the dataset. To characterize the local DB around x, we compute a counterfactual instance $\tilde{x}^{cf}$ by solving a constrained optimization problem using differential

evolution. The objective minimizes the Euclidean distance in normalized feature space while enforcing a change in the predicted class through a penalty-based constraint

$$\min_{\tilde{x}' \in X} \|norm(\tilde{x}') - x\| + \lambda * \mathbb{1}(f(\tilde{x}') = \tilde{y})$$

where $f(\cdot)$ denotes the dataset-specific classifier, $norm(\cdot)$ the min-max-normalization operator and λ a fixed penalty constant. The resulting counterfactual $\tilde{x}^{cf}$ represents the closest point crossing the true GT DB induced by the physical system. Its normalized representation x^{cf} defines the nearest boundary point. The GT explanation is finally defined as the vector $\overrightarrow{x^{cf} - x}$. The pair of original and counterfactual labels uniquely identifies the local DB type (also cf. D.1).

8.2 Appendix B: Model Configuration and Training

8.2.1 Appendix B.1: DB-focused Oversampling

To generate synthetic training data that faithfully reflects the decision structure of a GT proxy ML model, we employ a DB-focused sampling strategy. This technique constructs new samples in the vicinity of the model's DBs and labels them using the GT proxy model itself. Importantly, the procedure uses only a randomly selected subset of the original training data and explicitly excludes test data used in our experiments. We use this technique for all GT models. Concretely, we do this by applying the following steps. First, let $D = \{(x_i, y_i)\}_{i=1}^{N}$ denote the original dataset with C classes. A small balanced dataset subset D_s is obtained by randomly sampling n data points from D:

$$D_s = \bigcup_{c=1}^{C} \{(x_i, y_i) \mid y_i = c, \;\; i = 1, \dots, n)\}$$

This ensures class-balanced coverage of the underlying data distribution. To oversample the Pendulum and Inflation datasets, we train a model $\tilde{f}$ that achieves perfect prediction accuracy of 1 on D. As model type of $\tilde{f}$ we either use an NN, if later on there will be an NN trained on the oversampled data, or an RF, if an RF is trained on the oversampled data.

In a second step we calculate the global DBs of the GT model. For NN models, we do so by computing an exact piecewise linear reconstruction of the NN model [8, 17, 23]. Similarly, for RFs, we compute the intersections of the piecewise-constant regions induced by the individual DTs, thereby obtaining the piecewise linear structure of the RF.

Next, we determine the closest point x_{DB} on each DB for each instance $x_i \in D_s$. This yields a set of boundary points $= \{x_{DB_k}\}_{k=1}^{N_{DB}}$. To obtain dense samples near the DBs, each boundary point $x_{DB} \in B$ is locally perturbed, adding data points

$$x_{new} = x_{DB} + \varepsilon, \;\; \varepsilon \sim Uniform(-\delta, \delta)^d,$$

with feature dimension d and $\delta = 0.1$. The number of perturbations per boundary point is chosen adaptively:

$$n_{perturb} = \left\lceil \frac{10^6}{|B|} \right\rceil * \kappa,$$

with $\kappa = 2$ for the Prosper dataset and $\kappa = 10$ for all other datasets. This ensures approximately uniform coverage of all DBs while generating a minimum of 1,000,000 new data points.

All generated samples are labeled using the GT model f^* as $y_{new} = f^*(x)$. This guarantees that the synthetic dataset is fully consistent with the GT's decision behavior, rather than the empirical data distribution.

Let N_c denote the number of generated samples for class c. To prevent class imbalance, the number of samples per class is capped at

$$N_c^{max} = \begin{cases} min_{c'} N_{c'} & default, \\ 3 * min_{c'} N_{c'} & for\ Compas. \end{cases}$$

Finally, the dataset is resampled to a predefined target size N_{target}, yielding the final over-sampled dataset

$$D_{OS} = \{(x_i^{OS}, y_i^{OS})\}_{i=1}^{N_{target}}.$$

8.2.2 Appendix B.2: GT Models

For our GT models we use the following settings. All RF models use $n_{estimators} = 5$ and a maximum depth of 5, class-balanced training, and a fixed random seed to ensure reproducibility. To account for differences in dataset complexity, we distinguish between two groups. For lower complexity datasets (Fetal Health, Probabilistic), the fraction of features considered at each split is set to 0.75 with disabled bootstrap sampling. Higher complexity datasets (Compas, Prosper) use 0.8 as fraction of features considered for each split with bootstrap sampling enabled. In both cases, a minimum leaf size of 5% of the training data is enforced to regularize the trees.

All NNs are implemented as fully connected multilayer perceptrons with two hidden layers of 5 neurons each. ReLU activations and the Adam optimizer are used throughout, with training performed for at most 100 iterations. Early stopping and data shuffling are enabled to improve generalization and prevent overfitting. A fixed random seed is used to ensure reproducibility across runs.

For the Rashomon set in each experimental run, models are retrained on the generated datasets D_{OS} using random 80/20 train-test splits to ensure variability across repetitions.

RFs follow the same architecture as the GT models, using $n_{estimators} = 5$ and a maximum depth of 5. For lower complexity datasets, the fraction of features considered at each split is set to 0.75, and to 0.8 for higher complexity datasets. To regularize the trees the minimum leaf size is set to 200 data points.

NNs are implemented as fully connected multilayer perceptrons with ReLU activations and the Adam optimizer. Most datasets again use two hidden layers of size 5 each. For the Prosper dataset

we use slightly more neurons in the first layer (10), to better capture the 56 data features. Training is capped at 100 epochs with early stopping and data shuffling enabled.

8.3 Appendix C: XAI Method Configurations

All XAI methods are applied using their respective standard configurations as provided by the original implementations, unless stated otherwise.

Our choice of the GT explanation is further supported by prior work that explicitly defines local explanation GT via the geometry of DBs [15, 39]. In particular, Guidotti [15] proposes an evaluation framework in which the GT explanation for a given instance is derived from the closest point on the DB, yielding a FA vector that captures the perturbation required to change the model's prediction. Within this framework, local explanation methods, including LIME and SHAP, are quantitatively evaluated by measuring their alignment with this boundary-based GT explanation. This formulation provides a principled and model-agnostic reference for local explanations and directly motivates our use of the normalized vector toward the nearest DB as GT. To ensure a fair comparison of all FA methods against our GT explanation, both explanation vectors and GT vectors are scaled to equal length 1 in all experiments.

SHAP explanations are computed using the Kernel SHAP variant with default settings. Kernel SHAP is a model-agnostic approximation method based on weighted linear regression in the space of feature coalitions, providing estimates of Shapley values for arbitrary black-box models. This choice is motivated by methodological consistency across model types. In contrast, Tree SHAP and Deep SHAP are model-specific optimizations: Tree SHAP exploits the structure of DTs to compute exact Shapley values in polynomial time, while Deep SHAP leverages the compositional structure of deep NNs to efficiently approximate Shapley values via backpropagation. In multiclass settings, we generate SHAP-based FAs by aggregating the absolute values of the SHAP outputs across all classes, which is a commonly used approach to obtain one single FA vector per instance. Since this procedure yields only unsigned (positive) FA values, we ensure a fair comparison by evaluating all SHAP-related metrics with respect to the absolute values of the corresponding GT FA explanation.

ROLEX distinguishes between linear and non-linear DBs when multiple classes are present in the neighborhood of a reference point x_{center}. For ROLEX Linear, we employ the standard configuration by reframing the multi-class problem as a one-vs-rest classification task and using a logistic regression model as the local surrogate. In addition, we consider a variant denoted ROLEX Tree, in which DTs are used as surrogate models while preserving the original multi-class formulation. This extension enables ROLEX to capture non-linear DBs more faithfully and is particularly well suited for explaining RF models.

AGREE is applied using its default configuration. It operates by aggregating explanations generated by multiple local explanation methods. In this work, AGREE combines explanations from LIME, SHAP, and MAPLE in order be compatible with RFs and NNs.

AGGOPT is also used with standard settings. Similar to AGREE, it aggregates local explanations but formulates the aggregation as an optimization problem. In our experiments, AGGOPT combines explanations obtained from LIME and SHAP.

POIC is applied using its standard configuration. As the method is only applicable to linear models and RFs, it cannot be evaluated for NNs in our experimental setting.

8.4 Appendix D: Metrics

8.4.1. Appendix D.1: DB Accuracy

To evaluate how well a retrained model reflects the decision behavior of a proxy GT model, we introduce the 'DB accuracy' metric. This metric evaluates whether both models associate a given input instance with the same relevant DB.

Let $f: \mathbb{R}^d \to \{1, \dots, C\}$ denote a classifier and let $s_f: \mathbb{R}^d \to \mathbb{R}^C$ be its associated scoring function (e.g., yielding the Softmax outputs for NNs). The set of DBs of f can be defined as

$$B_f = \bigcup_{c \neq c'} \{x \in \mathbb{R}^d \mid s_f(x)_c = s_f(x)_{c'} \wedge \forall k \neq c: s_f(x)_c \geq s_f(x)_k\}.$$

For a given input instance $x \in \mathbb{R}^d$, the nearest DB point with respect to model f is defined as

$$x_f^* = \underset{z \in B_f}{\operatorname{argmin}} \|x - z\|.$$

Let $c(x) = \underset{k}{\operatorname{argmax}}\, s_f(x)_k$ be the predicted class of x. We define the DB type of x under model f as the ordered class pair

$$\tau_f(x) = \big(c(x), c'(x)\big),$$

where $c'(x) = \underset{k \neq c(x)}{\operatorname{argmax}}\, s_f(x_f^*)_k$ denotes the adjoining class across the nearest DB. Intuitively, $\tau_f(x)$ identifies the class assigned to x and the class that would be reached first when moving from x in the direction of the closest DB.

Given a test set X_{test}, a GT model f^*, and a retrained model f, the DB accuracy is defined as

$$DB\ accuracy = \frac{1}{|X_{test}|} \sum_{x \in X_{test}} \mathbb{1}\big[\tau_{f^*}(x) = \tau_f(x)\big],$$

where $\mathbb{1}$ denotes the indicator function. DB accuracy measures agreement at the level of local decision structure, rather than predictive correctness. A model achieves high DB accuracy if, for most inputs, it not only predicts the same class as the GT model but also exhibits the same nearest competing class and thus reproduces the same local DB structure.

8.4.2 Appendix D.2: Local Accuracy

To further quantify the local fidelity of a model with respect to the GT model, we also consider the 'Local Accuracy' metric around each instance of the test dataset. The procedure is as follows:

For a given instance $x \in X_{test}$, we compute the Euclidean distances to all points in the training dataset X_{train}. Let $d_{max} = max_{x' \in X_{train}} \|x - x'\|$ be the maximum distance. The radius of the local neighborhood is then defined as $r_{local} = 0.2 * d_{max}$, following the approach of ROLEX [22]. After determining the local neighborhood, we uniformly sample a set of N points within a d-dimensional hyperball of radius r_{local} centered at x. The sampled points are clipped to lie within the valid input range $[0,1]^d$ of our min-max standardized datasets. Further we obtain the Local Accuracy of a model based on its local predictions as follows: Let $\hat{y}_{GT}$ and $\hat{y}_{model}$ denote the predictions of the GT model f^* and the model f being evaluated, respectively, on the sampled points. The Local Accuracy in the neighborhood of instance x is defined as

$$Local\ Accuracy(x) = \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}\left(\hat{y}_{GT}^{i} = \hat{y}_{model}^{i}\right),$$

representing the ratio of samples in the local neighborhood where the model and the GT model agree. This procedure is applied for each instance in X_{test}. Local Accuracy measures how well a model approximates the reference model in the immediate vicinity of an instance, rather than globally across the dataset. Combined with the DB accuracy, this provides a meaningful metric to assess the model behavior of a model around a given instance with respect to the GT.

8.4.3 Appendix D.3: Evaluation Metrics

For each instance, we generate the FAs $\left(\phi(x, f_i)\right)_{i=1,\dots,n}$ over all models f_i from the Rashomon set and for each XAI method. We denote $\phi(x, f_i)_k$ as the Top k features in the FA explanation for model f_i, i.e., the set of the k features with the highest absolute values in the FA vector. The Top 5 Number of Unique Features (NUF_k) across $\left(\phi(x, f_i)\right)_{i=1,\dots,n}$ is defined as the number of unique features that appear in any of the Top k sets $\phi(x, f_i)_k$:

$$NUF_k(\left\{\left(\phi(x, f_i)\right)_{i=1,\dots,n}\right\}) = \left|\bigcup_{i=1}^{n} \phi(x, f_i)_k\right|$$

This metric aims at the diversity of features appearing in the explanations of single instance across all models of the Rashomon set. A much higher NUF_k value in comparison to k indicates a low consistency regarding the most relevant features provided by the explanations $\left(\phi(x, f_i)\right)_{i=1,\dots,n}$.

Similarly, the faithfulness of local explanations is also often evaluated regarding the Top k features, as it is more important for users to accurately recognize and rank, e.g., the Top 5 features than less important ones ranked between, e.g., 20 and 30. Therefore, we employ the following two well-known criteria for faithfulness: Top k Intersection and Top k Kendall τ. For a feature vector $\phi(x, f)$ and the GT feature vector $\phi^*(x, f^*)$ both faithfulness criteria are defined as follows

$$Intersection_k(\phi^*(x, f^*), \phi(x, f)) = \frac{|\phi^*(x, f^*)_k \cap \phi(x, f)_k|}{k}$$

$$\tau_k\big(\phi^*(x,f^*),\phi^*(x,f^*)\big) = \frac{2}{k\cdot(k-1)}\sum_{i<j} sgn(\phi^*(x,f^*)_{k,i} - \phi^*(x,f^*)_{k,j}) \cdot sgn(\phi(x,f)_{k,i} - \phi(x,f)_{k,j}),$$

where $\phi^*(x,f^*)_k$ and $\phi(x,f)_k$ again denote the first k features of $\phi^*(x,f^*)$ and $\phi(x,f)$ respectively. Note this definition of the Top k Kendall τ only compares the correct order of the Top k features between the two FAs. With the Intersection we can check if both FAs identify the same relevant features and with Top k Kendall τ we can further assess if $\phi(x,f)$ ranks the Top k features according to $\phi^*(x,f^*)$. Thus, the Kendall τ is a stricter criterion than Intersection since $\phi(x,f)$ not only needs to identify the relevant features, but also rank them in the correct order.

Δ5%-ExpAgr: This metric provides a further, strict binary evaluation metric of FA explanation. It returns 1 if all attributions of each feature lie within a ±2.5% deviation of their GT attribution values (typically of $\phi^*(x,f^*)$), and 0 otherwise. Formally:

$$\Delta 5\% - \mathrm{ExpAgr}(\phi^*,\phi) = \begin{cases} 1, & if\ |\phi_i^* - \phi_i| \le 0.025 \quad \forall i \in \{1,\dots,d\}, \\ 0, & otherwise. \end{cases}$$

A value of 1 for an explanation ϕ indicates that the FA explanation perfectly matches the GT ϕ^* across all features within the ±2.5% tolerance. A value of 0 indicates that at least one feature's attribution exceeds the a ±2.5% deviation. Therefore, this metric enforces a very strict agreement criterion, making it the strictest faithfulness measure for FA explanations considered in this work.

To illustrate the ExpAgr criterion and possible distributions of explanations observed in our empirical experiments, Figure 7 visualizes a kernel density estimate (KDE) of the FA explanations' distribution obtained from Rashomon models for a given input instance. The black points denote individual FA explanations produced for different RF models within the Rashomon set, while the red arrow represents the actual explanation vector of the NN proxy GT. The shaded region corresponds to the 95% probability mass level set of the KDE, capturing the high-density region of plausible explanations. The figure highlights that, although the explanations generated by the Rashomon models form a compact and highly consistent cluster, this cluster does not coincide with the actual GT explanation. Consequently, the GT explanation lies outside both the 95% probability mass region and the 0.025 tolerance range of the metric, indicating a systematic deviation rather than random variation and yielding a Δ5%- ExpAgr metric value of 0 for each explanation. This observation emphasizes that high consistency or stability among explanations does not imply correctness with respect to the underlying GT. The ExpAgr metric explicitly captures this discrepancy by requiring strict, feature-wise agreement with the GT explanation, thereby revealing explanation errors that would remain undetected by distributional or similarity-based metrics. This example hence demonstrates that explanation consistency within a Rashomon set may reinforce a shared bias rather than reflect the true decision behavior of the GT.

3.1 Paper 5: Are Consistent Explanations Also Correct? The Impact of Explanation and Model Uncertainty

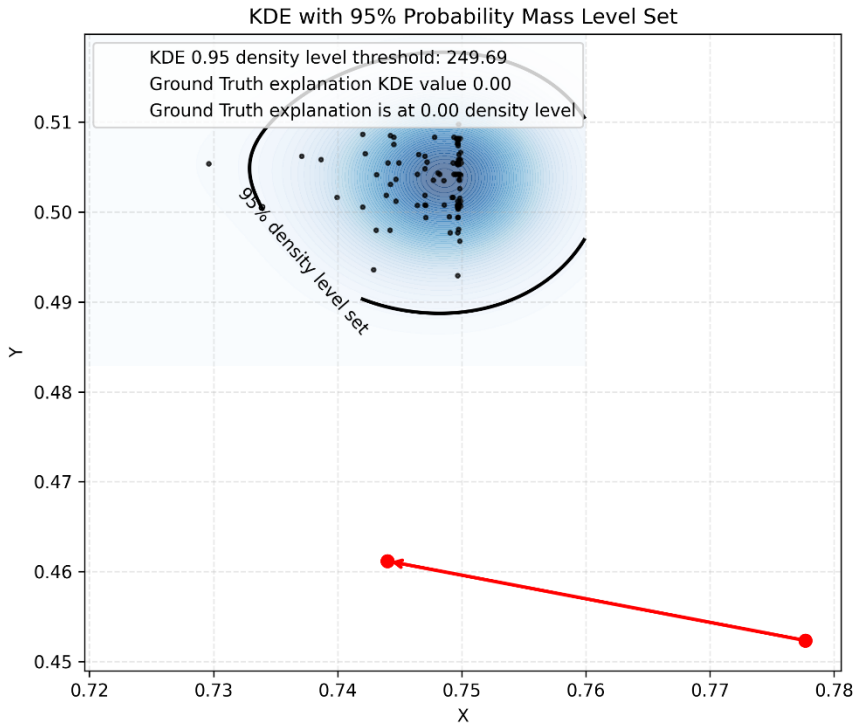


Figure 7: Distribution of Explanations of RF Models (black points) versus True Explanation by NN Proxy GT (red)

8.5 Appendix E: Complete Results

Dataset	Evaluation Setup	XAI Method	Top 5 Kendall τ	Top 5 Intersection	Δ5%-ExpAgr
Probabilistic	RF-RF	Kernel SHAP	-0.18	0.78	2.60
		Tree SHAP	-0.17	0.78	2.59
		ROLEX Linear	0.06	0.54	9.31
		ROLEX Tree	0.80	0.98	18.75
	NN-NN	Kernel SHAP	0.08	0.60	0.78
		Deep SHAP	0.04	0.62	0.67
Compas	RF-RF	Kernel SHAP	0.12	0.59	12.8
		Tree SHAP	0.23	0.55	12.8
		ROLEX Linear	0.16	0.48	15.67
		ROLEX Tree	0.86	0.97	25.31
	NN-NN	Kernel SHAP	0.27	0.62	12.86
		Deep SHAP	0.30	0.59	11.32
Fetal Health	RF-RF	Kernel SHAP	0.08	0.37	10.11
		Tree SHAP	0.08	0.37	10.11
		ROLEX Linear	0.01	0.38	17.58
		ROLEX Tree	0.93	0.98	28.62
	NN-NN	Kernel SHAP	0.18	0.60	2.67
		Deep SHAP	0.12	0.53	2.45
Prosper	RF-RF	Kernel SHAP	0.03	0.40	26.00
		Tree SHAP	0.03	0.40	26.00
		ROLEX Linear	0.27	0.11	26.32
		ROLEX Tree	0.90	0.61	28.91
	NN-NN	Kernel SHAP	-0.04	0.45	25.2
		Deep SHAP	-0.03	0.35	25.6
Pendulum	GT-RF	Kernel SHAP	0.00	0.75	18.52
		Tree SHAP	0.00	0.75	18.52
		ROLEX Linear	0.05	0.65	9.52
		ROLEX Tree	0.14	0.70	15.91
	GT-NN	Kernel SHAP	0.02	0.77	18.45
		Deep SHAP	0.01	0.74	17.13
Inflation	GT-RF	Kernel SHAP	-0.20	0.58	2.70
		Tree SHAP	-0.21	0.59	2.70
		ROLEX Linear	-0.14	0.59	1.85
		ROLEX Tree	-0.02	0.64	2.73
	GT-NN	Kernel SHAP	-0.11	0.65	5.67
		Deep SHAP	-0.13	0.65	4.87

Table 3: Results for Analysis of Factor 1a (XAI Structural Bias)

Number of Samples used by XAI Method		**100**				**1,000**				**10,000**			
Dataset	**XAI Method**	# Unique Features	Top 5 Kendall τ	Top 5 Intersection	Δ5%-ExpAgr	# Unique Features	Top 5 Kendall τ	Top 5 Intersection	Δ5%-ExpAgr	# Unique Features	Top 5 Kendall τ	Top 5 Intersection	Δ5%-ExpAgr
Probabilistic	LIME	10.43	0.05	0.66	1.21	7.19	0.18	0.70	2.55	5.81	0.22	0.69	2.61
	SHAP	6.71	0.08	0.60	0	5.71	0.09	0.60	0	5.76	0.09	0.60	0.78
	ROLEX	9.84	0.05	0.67	0.96	8.25	-0.05	0.80	7.36	7.86	0.35	0.81	7.56
	AGREE	8.22	0.08	0.57	-	7.70	0.09	0.58	-	7.71	0.09	0.58	-
	AGGOPT	6.46	0.19	0.69	-	6.38	0.20	0.70	-	6.44	0.18	0.69	-
Compas	LIME	14.37	0.37	0.61	14.13	11.11	0.47	0.68	15.73	8.40	0.43	0.69	15.91
	SHAP	6.92	0.28	0.63	6.48	5.79	0.27	0.63	6.58	5.79	0.27	0.63	6.6
	ROLEX	11.44	0.50	0.81	17.75	10.13	0.60	0.84	23.43	7.71	0.62	0.92	24.79
	AGREE	7.42	0.21	0.52	-	6.41	0.19	0.53	-	6.21	0.19	0.53	-
	AGGOPT	14.37	0.37	0.61	-	11.11	0.47	0.68	-	8.40	0.43	0.69	-
Fetal Health	LIME	12.76	0.01	0.55	1.98	8.25	0.02	0.62	2.17	6.44	-0.01	0.63	2.25
	SHAP	7.15	0.01	0.60	0	5.85	-0.00	0.60	0	5.83	0.01	0.60	2.6
	ROLEX	11.60	0.05	0.59	1.5	9.09	0.20	0.77	7.28	8.67	0.20	0.77	7.28
	AGREE	9.99	0.01	0.60	-	8.81	0.03	0.62	-	8.65	0.02	0.62	-
	AGGOPT	9.68	-0.10	0.52	-	9.80	-0.10	0.52	-	9.79	-0.09	0.52	-
Prosper	LIME	31.70	-0.15	0.24	24.9	28.59	0.02	0.40	25.07	23.31	0.20	0.57	25.03
	SHAP	8.79	-0.4	0.44	24.9	7.58	-0.04	0.45	24.9	7.46	-0.01	0.49	24.9
	ROLEX	25.65	-0.09	0.25	22.33	33.04	-0.05	0.33	24.51	7.47	0.03	0.11	26.62
	AGREE	23.95	-0.15	0.24	-	22.58	-0.15	0.26	-	26.18	-0.09	0.27	-
	AGGOPT	25.87	0.15	0.50	-	25.64	0.14	0.49	-	25.81	0.15	0.50	-
Pendulum	LIME	9	0.01	0.65	15.81	8.96	0.01	0.74	15.9	8.4	-0.00	0.78	15.83
	SHAP	7.14	-0.01	0.77	15.1	6.84	-0.02	0.77	14.92	6.84	-0.02	0.77	14.92
	ROLEX	8.68	0.04	0.65	6.75	7.86	0.01	0.73	5.88	7.10	-0.03	0.77	5.71
	AGREE	8.01	0.03	0.77	-	7.49	0.03	0.79	-	7.4	0.03	0.79	-
	AGGOPT	8.61	-0.02	0.79	-	8.53	-0.03	0.79	-	8.51	-0.02	0.79	-
Inflation	LIME	7.98	-0.02	0.67	3.56	6.12	-0.03	0.69	3.78	5.72	-0.06	0.70	4
	SHAP	5.71	-0.11	0.65	2.82	5.27	-0.11	0.65	2.7	5.27	-0.11	0.65	2.7
	ROLEX	7.52	0.02	0.59	2.9	6.77	-0.19	0.62	2.44	6.00	-0.25	0.60	2.49
	AGREE	6.36	-0.04	0.63	-	6.10	-0.06	0.63	-	6.00	-0.06	0.63	-
	AGGOPT	6.14	-0.06	0.68	-	5.92	-0.05	0.68	-	5.92	-0.05	0.68	-

Table 4: Results for Analysis of Factor 1b (XAI Estimation Variance) for NN-NN Setup

Number of Samples used by XAI Method		**100**				**1,000**				**10,000**			
Dataset	**XAI Method**	# Unique Features	Top 5 Kendall τ	Top 5 Intersection	Δ5%-ExpAgr	# Unique Features	Top 5 Kendall τ	Top 5 Intersection	Δ5%-ExpAgr	# Unique Features	Top 5 Kendall τ	Top 5 Intersection	Δ5%-ExpAgr
Probabilistic	LIME	11.71	0.08	0.57	3.46	10.46	0.16	0.66	7.37	7.72	0.21	0.70	7.72
	SHAP	5.47	-0.17	0.78	0.61	5.35	-0.17	0.78	0.6	5.32	-0.17	0.78	0.6
	ROLEX	11.16	0.07	0.52	4.25	11.36	0.05	0.55	8.93	11.49	0.06	0.55	10.65
	AGREE	10.23	0.01	0.59	-	9.91	0.01	0.59	-	9.84	0.01	0.59	-
	AGGOPT	6.96	0.08	0.47	-	6.95	0.08	0.47	-	7.18	0.08	0.47	-
Compas	LIME	14.57	0.16	0.53	13.56	14.36	0.24	0.57	14.68	14.13	0.24	0.55	14.84
	SHAP	13.15	0.13	0.60	7.96	12.57	0.11	0.60	8.18	12.61	0.12	0.59	8.17
	ROLEX	13.98	0.11	0.42	11.82	13.89	0.19	0.49	14.98	13.73	0.16	0.48	15.67
	AGREE	8.88	0.32	0.73	-	8.94	0.32	0.73	-	8.93	0.32	0.73	-
	AGGOPT	9.41	0.47	0.72	-	9.11	0.46	0.71	-	9.08	0.46	0.72	-
Fetal Health	LIME	17.20	-0.01	0.33	11.69	15.44	0.07	0.39	15.82	12.20	0.13	0.39	16.17
	SHAP	7.50	0.08	0.37	5.33	6.64	0.09	0.38	5.4	7.43	0.08	0.37	10.11
	ROLEX	15.24	-0.01	0.39	13.59	16.24	0.02	0.38	17.9	16.28	0.02	0.38	17.58
	AGREE	10.80	0.02	0.74	-	8.82	-0.00	0.75	-	8.49	-0.01	0.75	-
	AGGOPT	9.68	-0.10	0.52	-	9.80	-0.10	0.52	-	9.79	-0.09	0.52	-
Prosper	LIME	28.81	0.14	0.29	26.43	28.65	0.16	0.30	26.26	28.09	0.15	0.29	26.12
	SHAP	10.40	0.03	0.41	25.5	9.93	0.01	0.40	25.5	9.82	0.01	0.40	25.5
	ROLEX	24.98	0.15	0.22	23.81	36.22	0.11	0.27	26.43	38.77	0.16	0.31	26.64
	AGREE	28.06	-0.14	0.30	-	20.68	0.01	0.30	-	17.02	0.01	0.30	-
	AGGOPT	27.45	-0.11	0.23	-	27.33	-0.12	0.23	-	27.50	-0.13	0.23	-
Pendulum	LIME	9	0.001	0.63	15.65	9	0.01	0.71	15.98	7.56	0.00	0.76	15.93
	SHAP	7.69	-0.00	0.75	15.33	6.6	-0.01	0.75	15.17	6.6	-0.01	0.75	15.17
	ROLEX	9	0.05	0.62	10.44	8.96	0.05	0.65	9.67	8.81	0.05	0.67	9.38
	AGREE	7.85	0.04	0.78	-	6.77	0.03	0.80	-	6.48	0.03	0.80	-
	AGGOPT	8.25	-0.01	0.76	-	8.25	-0.01	0.76	-	8.31	0.00	0.75	-
Inflation	LIME	7.96	-0.16	0.60	3.21	7.70	-0.21	0.58	3.11	5.78	-0.25	0.59	3.06
	SHAP	5.98	-0.21	0.59	2.7	5.23	-0.20	0.58	2.7	5.23	-0.20	0.59	2.7
	ROLEX	7.71	-0.13	0.60	1.99	7.35	-0.17	0.60	1.77	7.42	-0.14	0.59	1.85
	AGREE	7.72	-0.25	0.59	-	7.08	-0.28	0.59	-	6.79	-0.29	0.59	-
	AGGOPT	6.15	-0.24	0.59	-	6.04	-0.24	0.59	-	6.01	-0.26	0.59	-

Table 5: Results for Analysis of Factor 1b (XAI Estimation Variance) for RF-RF Setup

Dataset	XAI Method	NN-NN			RF-RF			NN-RF			RF-NN		
		Top 5 Kendall τ	Top 5 Intersection	Δ5%-ExpAgr	Top 5 Kendall τ	Top 5 Intersection	Δ5%-ExpAgr	Top 5 Kendall τ	Top 5 Intersection	Δ5%-ExpAgr	Top 5 Kendall τ	Top 5 Intersection	Δ5%-ExpAgr
Probabilistic	DB Accuracy		1.00			1.00			0.75			0.97	
	Local Accuracy		1.00			1.00			0.93			0.99	
	LIME	0.05	0.66	1.21	0.16	0.66	7.37	-0.07	0.71	0.3	0.26	0.69	5.76
	SHAP	0.08	0.60	0.81	-0.17	0.78	2.67	-0.03	0.70	0.01	-0.13	0.75	2.42
	ROLEX	0.05	0.67	0.96	0.05	0.55	8.93	0.06	0.57	0.15	0.04	0.54	9.53
	EXPLORE	0.96	0.99	29.8	0.86	0.98	24.66	0.03	0.52	0	0.14	0.64	21.4
	AGREE	0.09	0.58	-	0.01	0.59	-	-0.02	0.43	-	0.03	0.45	-
	AGGOPT	0.20	0.70	-	0.08	0.47	-	-0.06	0.71	-	0.05	0.48	-
	POIC	-	-	-	0	0	-	0	0	-	-	-	-
Compas	DB Accuracy		1.00			0.99			0.78			0.96	
	Local Accuracy		0.99			1.00			0.76			0.96	
	LIME	0.44	0.69	15.83	0.25	0.55	14.93	0.59	0.63	21.93	0.24	0.56	14.87
	SHAP	0.28	0.63	12.63	0.13	0.60	12.79	0.54	0.62	19.03	0.21	0.58	12.84
	ROLEX	0.61	0.83	23.24	0.20	0.49	15.38	0.36	0.56	12.71	0.22	0.50	15.63
	EXPLORE	0.83	0.98	29.77	0.97	0.99	29.25	0.07	0.42	7.08	0.34	0.57	24.48
	AGREE	0.20	0.52	-	0.03	0.73	-	0.15	0.55	-	0.03	0.31	-
	AGGOPT	0.47	0.71	-	0.17	0.43	-	0.59	0.64	-	0.17	0.42	-
	POIC	-	-	-	0	0	-	0	0	-	-	-	-
Fetal Health	DB Accuracy		1.00			1.00			0.57			0.95	
	Local Accuracy		1.00			0.99			0.70			0.94	
	LIME	0.00	0.63	2.18	0.07	0.39	15.82	0.15	0.61	0.00	0.23	0.39	16.37
	SHAP	0.01	0.53	2.45	0.09	0.38	10.02	-0.01	0.56	0.47	0.10	0.37	9.71
	ROLEX	0.18	0.77	7.46	0.02	0.38	17.9	0.02	0.43	0.08	0.00	0.38	15.7
	EXPLORE	0.76	0.99	28.91	0.97	0.99	30	-0.13	0.35	0.70	-0.01	0.38	25.78
	AGREE	0.02	0.61	-	0.00	0.75	-	0.04	0.38	-	-0.04	0.46	-
	AGGOPT	0.02	0.63	-	-0.1	0.52	-	0.13	0.60	-	-0.17	0.43	-
	POIC	-	-	-	0	0	-	0	0	-	-	-	-
Prosper	DB Accuracy		1.00			1.00			0.94			0.87	
	Local Accuracy		0.97			1.0			0.79			0.90	
	LIME	0.14	0.50	25.07	0.14	0.29	26.16	-0.01	0.40	25.13	0.08	0.31	26.41
	SHAP	-0.04	0.45	25.2	0.03	0.40	26	0.07	0.51	25.25	0.10	0.42	25.91
	ROLEX	-0.05	0.33	24.51	0.11	0.27	26.32	-0.09	0.32	25.5	0.11	0.28	25.26
	EXPLORE	0.78	0.94	29.89	0.99	1.00	29.81	-0.36	0.21	26.76	0.03	0.34	28.52
	AGREE	-0.09	0.27	-	0.01	0.30	-	-0.18	0.18	-	-0.01	0.17	-
	AGGOPT	0.13	0.50	-	-0.12	0.23	-	0.00	0.41	-	-0.03	0.22	-
	POIC	-	-	-	-0.53	0.20	-	0.18	0.24	-	-	-	-

Pendulum	DB Accuracy		-			-			-			-	
	Local Accuracy		-			-			-			-	
	LIME	0.03	0.76	15.34	0.01	0.75	16.02	-	-	-	-	-	-
	SHAP	0.02	0.77	18.45	-0.00	0.75	18.52	-	-	-	-	-	-
	ROLEX	0.04	0.72	7.87	0.05	0.65	9.52	-	-	-	-	-	-
	EX-PLORE	-0.03	0.78	18.76	0.19	0.70	15.66	-	-	-	-	-	-
	AGREE	0.04	0.79	-	0.03	0.80	-	-	-	-	-	-	-
	AGGOPT	0.04	0.77	-	-0.01	0.75	-	-	-	-	-	-	-
	POIC	-	-	-	0	0	-	-	-	-	-	-	-
Inflation	DB Accuracy		-			-			-			-	
	Local Accuracy		-			-			-			-	
	LIME	-0.06	0.70	4	-0.25	0.59	3.06	-	-	-	-	-	-
	SHAP	-0.11	0.65	5.67	-0.20	0.58	2.7	-	-	-	-	-	-
	ROLEX	-0.25	0.60	2.49	-0.14	0.59	1.85	-	-	-	-	-	-
	EX-PLORE	-0.27	0.59	4.2	-0.09	0.63	2.7	-	-	-	-	-	-
	AGREE	-0.06	0.63	-	-0.29	0.59	-	-	-	-	-	-	-
	AGGOPT	-0.05	0.68	-	-0.22	0.59	-	-	-	-	-	-	-
	POIC	-	-	-	0	0	-	-	-	-	-	-	-

Table 6: Results for Analysis of Factor 2a (ML Model Structural Bias)

Number of Samples		**2.000**			**10.000**			**100.000**			**1.000.000**		
Dataset	**XAI Method**	Top 5 Kendall τ	Top 5 Intersection	Δ5%-ExpAgr	Top 5 Kendall τ	Top 5 Intersection	Δ5%-ExpAgr	Top 5 Kendall τ	Top 5 Intersection	Δ5%-ExpAgr	Top 5 Kendall τ	Top 5 Intersection	Δ5%-ExpAgr
Probabilistic	LIME	0.02	0.64	0.9	0.20	0.70	2.74	0.21	0.69	2.51	0.05	0.66	1.21
	SHAP	0.05	0.56	0.49	0.08	0.61	0.85	0.07	0.60	0.94	0.08	0.60	0.81
	ROLEX	0.03	0.65	0.99	0.13	0.79	6.92	0.13	0.80	7.55	0.05	0.67	0.96
	EXPLORE	0.01	0.71	2.69	0.59	0.95	27.67	0.90	0.97	29.66	0.96	0.99	29.8
	AGREE	0.00	0.40	-	0.06	0.43	-	0.11	0.55	-	0.09	0.58	-
	AGGOPT	0.05	0.64	-	0.18	0.69	-	0.18	0.69	-	0.20	0.70	-
Compas	LIME	0.44	0.61	14.28	0.57	0.78	17.23	0.44	0.70	16.24	0.43	0.69	15.91
	SHAP	0.27	0.54	11.65	0.27	0.63	12.61	0.28	0.64	13.04	0.27	0.63	11.32
	ROLEX	0.47	0.74	18.18	0.56	0.84	23.6	0.59	0.83	23.45	0.62	0.92	24.79
	EXPLORE	0.48	0.80	20.68	0.49	0.98	28.62	0.76	0.96	28.95	0.83	0.98	29.47
	AGREE	0.29	0.29	-	0.35	0.36	-	0.28	0.50	-	0.19	0.53	-
	AGGOPT	0.45	0.61	-	0.59	0.79	-	0.46	0.72	-	0.47	0.71	-
Fetal Health	LIME	-0.02	0.35	0.91	-0.03	0.60	2.24	-0.02	0.62	2.18	0.00	0.63	2.18
	SHAP	0.02	0.37	1.68	0.00	0.59	2.57	0.01	0.59	2.65	0.01	0.53	2.45
	ROLEX	0.04	0.36	0.62	0.12	0.72	5.27	0.21	0.78	6.98	0.18	0.77	7.46
	EXPLORE	0.04	0.37	1.03	0.19	0.84	14.19	0.65	0.98	24.35	0.76	0.99	28.91
	AGREE	0.01	0.32	-	-	-	-	0.02	0.56	-	0.02	0.61	-
	AGGOPT	0.02	0.36	-	0.00	0.63	-	0.02	0.63	-	0.02	0.63	-
Prosper	LIME	-0.03	0.12	24.98	0.18	0.35	25.07	0.15	0.47	25.02	0.14	0.50	25.07
	SHAP	0.05	0.08	24.9	0.08	0.23	25.03	-0.05	0.35	25.2	-0.04	0.45	25.2
	ROLEX	0.00	0.23	17.46	-0.06	0.31	22.42	-0.05	0.33	24.42	-0.05	0.33	24.51
	EXPLORE	0.04	0.10	25.04	0.07	0.31	26.18	0.08	0.55	26.08	0.78	0.94	29.89
	AGREE	0.08	0.04	-	0.05	0.16	-	-0.10	0.22	-	-0.09	0.27	-
	AGGOPT	0.02	0.16	-	0.15	0.35	-	0.13	0.47	-	0.13	0.50	-
Pendulum	LIME	-0.01	0.52	15.16	-0.01	0.52	15.16	-0.02	0.77	15.94	0.03	0.76	15.34
	SHAP	-0.01	0.56	15.69	-0.01	0.56	15.69	-0.01	0.76	17.73	0.02	0.77	18.45
	ROLEX	-0.00	0.57	6.22	-0.00	0.57	6.22	0.04	0.75	5.2	0.04	0.72	7.87
	EXPLORE	0.01	0.57	15.23	0.01	0.57	15.23	-0.02	0.78	18.33	-0.03	0.78	18.76
	AGREE	0.01	0.59	-	0.01	0.59	-	0.04	0.78	-	0.04	0.79	-
	AGGOPT	-0.00	0.52	-	-0.00	0.52	-	-0.03	0.79	-	0.04	0.77	-
Inflation	LIME	-0.24	0.60	3.66	-0.23	0.60	3.58	-0.23	0.60	3.65	-0.06	0.70	4
	SHAP	-0.09	0.64	6.75	-0.08	0.65	6.78	-0.07	0.64	7.01	-0.11	0.65	5.67
	ROLEX	-0.16	0.62	2.52	-0.20	0.62	2.62	-0.20	0.62	2.48	-0.25	0.60	2.49
	EXPLORE	-0.25	0.59	4.18	-0.27	0.59	4.2	-0.27	0.59	4.2	-0.27	0.59	4.2
	AGREE	0.13	0.62	-	0.11	0.62	-	0.00	0.62	-	-0.06	0.63	-
	AGGOPT	-0.23	0.60	-	-0.21	0.60	-	-0.22	0.60	-	-0.05	0.68	-

Table 7: Results for Analysis of Factor 2b (ML Model Estimation Variance) for NN-NN Setup

Number of Samples		**2.000**			**10.000**			**100.000**			**1.000.000**		
Dataset	**XAI Method**	Top 5 Kendall τ	Top 5 Intersection	Δ5%-ExpAgr	Top 5 Kendall τ	Top 5 Intersection	Δ5%-ExpAgr	Top 5 Kendall τ	Top 5 Intersection	Δ5%-ExpAgr	Top 5 Kendall τ	Top 5 Intersection	Δ5%-ExpAgr
Probabilistic	LIME	0.10	0.61	7.71	0.23	0.65	6.81	0.20	0.70	7.81	0.16	0.66	7.37
	SHAP	0.11	0.63	2.97	-0.09	0.74	3.02	-0.15	0.78	2.91	-0.17	0.78	2.67
	ROLEX	0.04	0.55	5.22	0.03	0.54	9.38	0.06	0.55	9.46	0.05	0.55	8.93
	EXPLORE	0.63	0.95	16.68	0.80	0.98	23.03	0.86	0.98	25.22	0.86	0.98	24.66
	AGREE	-0.01	0.33	-	0.25	0.39	-	0.11	0.41	-	0.01	0.59	-
	AGGOPT	0.10	0.42	-	0.08	0.44	-	0.12	0.43	-	0.08	0.47	-
	POIC	0.10	0.38	-	0	0	-	0	0	-			
Compas	LIME	0.17	0.55	16.71	0.24	0.57	14.79	0.24	0.56	14.86	0.25	0.55	14.93
	SHAP	-0.07	0.81	15.99	0.07	0.66	15.00	0.15	0.60	11.91	0.12	0.59	12.8
	ROLEX	0.10	0.49	10.64	0.20	0.52	16.08	0.18	0.50	14.43	0.16	0.48	15.67
	EXPLORE	0.62	0.91	18.84	0.86	0.98	23.76	0.93	0.99	27.68	0.96	0.99	29.25
	AGREE	0.00	0.34	-	0.04	0.38	-	0.2	0.45	-	0.03	0.73	-
	AGGOPT	0.22	0.42	-	0.17	0.41	-	0.17	0.43	-	0.17	0.43	-
	POIC	0.13	0.31	-	0.02	0.26	-	0	0	-	0	0	-
Fetal Health	LIME	0.30	0.28	13.03	-0.02	0.39	17.42	0.10	0.39	16.84	0.07	0.39	15.82
	SHAP	0.31	0.30	10.48	0.11	0.38	9.97	0.07	0.37	10.1	0.09	0.38	10.02
	ROLEX	0.32	0.35	16.85	0.03	0.38	17.42	0.00	0.38	17.58	0.02	0.38	17.9
	EXPLORE	0.90	0.94	23.25	0.93	0.98	28.89	0.96	0.99	30	0.97	0.99	30
	AGREE	0.50	0.51	-	0.01	0.52	-	-0.02	0.74	-	-0.00	0.75	-
	AGGOPT	0.42	0.40	-	0.01	0.52	-	-0.07	0.52	-	-0.1	0.52	-
	POIC	0	0	-	0	0	-	0	0	-	0	0	-
Prosper	LIME	0.17	0.35	25.95	0.03	0.30	26.25	0.13	0.29	26.17	0.14	0.29	26.16
	SHAP	0.02	0.50	25.93	0.04	0.40	26.19	0.07	0.41	26.02	0.03	0.40	26
	ROLEX	0.14	0.27	26.14	0.01	0.27	26.34	0.12	0.27	26.5	0.11	0.27	26.32
	EXPLORE	0.96	0.99	28.95	0.98	1.00	29.6	0.99	1.00	29.88	0.99	1.00	29.81
	AGREE	-0.25	0.21	-	0.01	0.21	-	0.00	0.24	-	0.01	0.30	-
	AGGOPT	-0.17	0.23	-	-0.03	0.23	-	-0.13	0.23	-	-0.12	0.23	-
	POIC	-0.12	0.06	-	0.14	0.19	-	-0.28	0.20	-	-0.53	0.20	-
Pendulum	LIME	0.03	0.65	15.39	0.03	0.65	15.39	0.002	0.73	16.04	0.01	0.75	16.02
	SHAP	-0.02	0.69	17.11	-0.02	0.69	17.11	-0.01	0.74	18.24	-0.00	0.75	18.52
	ROLEX	0.00	0.60	6.17	0.00	0.60	6.17	0.02	0.65	7.46	0.05	0.65	9.52
	EXPLORE	0.11	0.71	16.11	0.11	0.71	16.11	0.12	0.71	16.13	0.19	0.70	15.66
	AGREE	0.03	0.71	-	0.03	0.71	-	0.05	0.78	-	0.03	0.80	-
	AGGOPT	0.02	0.64	-	0.02	0.64	-	-0.02	0.73	-	-0.01	0.75	-
	POIC	0.13	0.71	-	0.10	0.70	-	0.08	0.70	-	0	0	-
Inflation	LIME	-0.17	0.62	2.71	-0.05	0.59	2.71	-0.25	0.59	2.86	-0.25	0.59	3.06
	SHAP	-0.18	0.60	2.72	-0.06	0.59	2.73	-0.23	0.59	2.7	-0.20	0.58	2.7
	ROLEX	-0.18	0.60	1.97	-0.15	0.61	2.04	-0.19	0.60	1.82	-0.14	0.59	1.85
	EXPLORE	-0.04	0.63	2.79	-0.06	0.63	2.69	-0.09	0.63	2.7	-0.09	0.63	2.7
	AGREE	0.01	0.63	-	-0.15	0.59	-	-0.15	0.60	-	-0.29	0.59	-
	AGGOPT	-0.18	0.62	-	-0.05	0.59	-	-0.26	0.59	-	-0.22	0.59	-
	POIC	0.06	0.64	-	0.01	0.63	-	0	0	-	0	0	-

Table 8: Results for Analysis of Factor 2b (ML Model Estimation Variance) for RF-RF Setup

3.2 Paper 6: EXPLORE: A Novel Method for Local Explanations

Current Status	Full Citation
This paper is **accepted** and **published** in the *Proceedings of the 45th International Conference on Information Systems*	Heinrich, B., Krapf, T., & Miethaner, P. (2024). EXPLORE: A novel method for local explanations. In *Proceedings of the 45th International Conference on Information Systems*, Bangkok, Thailand. https://aisel.aisnet.org/icis2024/aiinbus/aiinbus/21

Abstract:

Artificial Intelligence (AI) and especially Machine Learning (ML) models are ubiquitous in research, business and society. However, the predictions of many ML models are often not transparent for users due to their black box nature. Therefore, several Explainable AI (XAI) methods aiming to provide local explanations for individual ML model predictions have been proposed. Importantly, existing XAI methods relying on surrogate models still have critical weaknesses regarding fidelity, robustness and sensitivity. Thus, we propose a novel method that avoids building surrogate models but instead represents the actual decision boundaries and class subspaces of ML models in a functional and definite manner. Further, we introduce two well-founded measures for the sensitivity of individual data instances regarding changes of their features values. We theoretically and empirically evaluate the fidelity and robustness of our method (on three real-world datasets) outperforming existing methods and demonstrate the validity and meaningfulness of our sensitivity measures.

Keywords: Explainable artificial intelligence, XAI, local explanation, sensitivity

Please note that the overall formatting of the paper has been adjusted to the remainder of the dissertation.

The paper as published by AIS is available at: https://aisel.aisnet.org/icis2024/aiinbus/aiinbus/21

1 Introduction

Artificial Intelligence (AI) and its applications are pervasive in business and society, and their importance and impact on everyday life continues to rise. Although the transparency of AI models can lead to side effects (cf., e.g., Bauer & Gill, 2024), it is indispensable to explain the predictions and decisions of Machine Learning (ML) models for a responsible use of AI (Brasse et al., 2023). The importance of this topic has been recognized and intensively discussed, which becomes evident by the AI Act recently approved by the EU parliament (EU AI Act, 2024). The AI Act regulates the use of AI by categorizing it into risk classes and imposing requirements towards their transparency, accuracy/fidelity, and robustness. This is particularly relevant for AI models that are applied in the fields of education, healthcare, or other critical domains. To address these critical requirements, the research field of Explainable AI (XAI) aims to mitigate the risks by improving the explainability of black box ML models (Brasse et al., 2023; Ribeiro et al., 2016). Among other valuable contributions of XAI, a plethora of methods have been proposed to make black box models and their predictions more transparent, since they play a vital role in ML-augmented decision-making, especially in critical domains such as medical diagnostics (Kim et al., 2023).

In this paper, we aim to contribute to the body of knowledge dealing with transparency and explanations of individual class predictions (instances) of ML models, and thus local explainability. Moreover, we focus on model-agnostic XAI methods, which are not specialized to explain one particular type of ML model. Literature offers a large body of knowledge for model-agnostic local XAI methods (i.e., methods that provide local explanations), for which LIME (Ribeiro et al., 2016), LEAP (Jia et al., 2019), and ROLEX (recently published by Kim et al. (2023) in their MISQ paper) are important examples. Like many other model-agnostic local XAI methods, they generate simpler surrogate models which aim to locally explain the neighborhood of a considered instance. Despite the strengths of these methods and the interpretable outcomes they provide for black box models, they still suffer from methodological issues associated with the generation of surrogate models. Although being aware that surrogates are intended to simplify the local decision boundaries (DBs) of a ML model, their generation using approximations, random samples, and other stochastic elements leads to significant shortcomings. For example, capturing the structure of the DBs and class subspaces (CSs) even locally often poses a substantial problem, especially when multiple DBs occur (cf. details in Section Related Work). As a result, existing XAI methods exhibit both limited accuracy/fidelity and robustness in the sense that they determine false predictions for instances compared to the ML model to be explained, crucially leading to inaccurate explanations. Moreover, existing XAI methods are not able to validly assess the sensitivity of a model prediction of the instance considered. Consequently, even if a minor change in the instance's feature values completely changes the model prediction, this would not be recognized. Striving to address these limitations, we propose EXPLORE (EXPlaining machine learning models by LOcal REconstruction), a novel local XAI method which does not rely on surrogate models or stochastic elements but explicitly focuses on the reconstruction of all DBs and CSs in the neighborhood of the instance to be explained. Our (meta) method is model-

agnostic and its instantiation is demonstrated for classification NNs as typical black box ML models in depth. Thus, the research questions that we aim to address are the following:

RQ1 *How can the DBs and CSs of a ML model be locally reconstructed in a functional and definite manner, thus making them fully transparent?*

RQ2 *Based on RQ1, how can the sensitivity of a classification decision be rigorously analyzed?*

Our contributions are twofold: We (1) propose EXPLORE, a novel post hoc local explanation *method* that uses piecewise linear functions to locally reconstruct the DBs and CSs of ML classification models in a functional and definite manner. We (2) present two novel *measures* allowing for an exact analysis of the sensitivity of ML decisions. Based on the functional and definite reconstruction of the DBs and CSs of the ML model, our method significantly outperforms existing local XAI methods regarding fidelity and robustness. We provide formal proofs of the fidelity and robustness of EXPLORE and substantiate these theoretical guarantees with empirical evidence on three real-world datasets from the domains of healthcare, loan approval, and criminal recidivism. Moreover, we demonstrate the validity and meaningfulness of our sensitivity measures for selected instances from the healthcare dataset. These contributions come with several implications: First the DBs and CSs of a classification ML model are made fully transparent, providing the basis for accurate explanations and visualizations of the feature space in the neighborhood of a considered instance. Moreover, although in this paper we instantiate our meta method in depth only for NNs, it is model-agnostic and our theoretical foundations also apply for other ML classification models such as random forests. Finally, EXPLORE also provides the basis for the valid assessment of the sensitivity of ML predictions which is vital for the transparent application of ML models especially in high-risk domains.

The remainder of the paper is structured as follows. In the following section, we position our work in literature, especially related to existing post hoc local XAI methods and derive the research gap. Next, we present the theoretical foundations and the basic ideas for both our method and the sensitivity measures. We then introduce the meta method of EXPLORE and instantiate it for feedforward NNs. On this basis, we present our two sensitivity measures for ML model predictions. Thereafter, we evaluate our method as well as the sensitivity measures and discuss our results and main findings. Finally, we conclude by reflecting on limitations and providing an outlook on future research.

2 Background and Related Work

A central goal of XAI is to provide transparent and accurate explanations for the predictions of ML models. In general, ML models aim to extract and generalize patterns from a given sample of data (i.e., the training data) in their learning procedure. This should allow them to make (at best) accurate predictions for new, unseen data instances from the same feature space based on these patterns. In the case of a classification task, a ML model $f: X \to Y$ is learned which assigns an output value $y = f(x)$ to any instance $x \in X$. Usually, the output value $y \in Y$ is either given by a class itself or by a value directly corresponding to a class, e.g., the Softmax vector of a NN

prediction (Goodfellow et al., 2016). In this sense, the ML model divides the feature space into several CSs, each consisting of predictions that are assigned to the same class by the ML model. The boundaries between CSs are the DBs and can be linear or non-linear, depending on the classification task and ML model used (Bishop, 2006; Kim et al., 2023). In general, there exist symbolic and subsymbolic ML models. Symbolic ML models such as, e.g., decision trees or variants of regressions, incorporate DBs with a higher degree of transparency, because the model prediction can be reconstructed directly based on the feature values of the instance and the learned model parameters. For example, in a decision tree, an instance is assigned to one path if a feature value exceeds a learned threshold, and to another path if it falls below that threshold. By examining the feature values and associated parameters, this decision can be directly recognized. Traversing all nodes of the tree reveals the model's class decisions, making its DBs and CSs transparent. However, many well-established ML models such as NNs or support vector machines are subsymbolic, i.e., their DBs and CSs are not transparent from the learned model parameters. This makes these ML models black boxes, as decisions cannot be directly recognized (Bishop, 2006).

For reviewing both existing local XAI methods and criteria to assess their performance, as well as for our discussions throughout the rest of the paper, we briefly present a running example. We introduce the real-world Fetal Health dataset (Ayres-de-Campos et al., 2000) for this purpose, on which we trained a typical (black box) feedforward NN (details cf. Section Evaluation). We have chosen this dataset from the healthcare domain because it emphasizes the importance of an accurate and robust explanation. The prediction of the NN should support the diagnosis of a fetus's health and comprises the classes "NORMAL" (the fetus is healthy), "SUSPECT" (the fetus might not be healthy and further medical checks are required) and "PATHOLOGICAL" (the fetus's health is critical and immediate medical action is required) based on the following five data features: The first feature is the fetus's *Baseline Heart Rate*. Two further features capture possible abnormalities of the heart rate, namely the number of *Prolonged Decelerations* (per 1,000 seconds), and the relative fraction of time in which the heart rate exhibits *Abnormal Short-Term Variability* (i.e., strong fluctuations in the heart rate). Moreover, the difference between the lowest and the highest fetal heart rate (called *Heart Rate Histogram Width*) and the number of *Uterine Contractions* experienced by the mother (again per 1,000 seconds) are considered. The classification task corresponds to a highly critical decision situation, aiming to support a diagnosis of a fetus's health for which an explanation of each ML prediction should be transparent and accurate.

When reviewing local XAI methods in more depth, it is important to understand how their performance has to be evaluated from a methodological perspective. One central and well-known criterion for performance is fidelity or accuracy (Kim et al., 2023; Laugel et al., 2018). It expresses the extent to which the decisions of the surrogate model generated by a XAI method locally match the decisions of the ML model being explained. Precisely, measures such as the local fidelity score (local faithfulness) or the local DB-aware fidelity score (LDA, cf. Kim et al., 2023) have to be revealed here which, for a number of instances, assess whether the classes assigned by the surrogate model match the classes assigned by the ML model. If this match holds

true for all instances, the local (DB-aware) fidelity would be perfect (i.e., 100% or 1). Even in a local environment this fidelity can fall well below 100%, which is critical because then the surrogate model evidently fails to provide an accurate explanation of the ML model, as it suggests false class assignments for multiple instances. This means that if, for example, the ML model predicts Class "SUSPECT" for a fetus, while the surrogate model predicts Class "NORMAL", the XAI method would try to explain the (false) class. Further important criteria of performance are robustness or reproducibility (Alvarez-Melis & Jaakkola, 2018; Yeh et al., 2019), which aim to assess how stable or similar the explanations of a XAI method are for (very) similar instances or even the same instance. More precisely, one has to measure the degree to which the explanations of similar instances are consistent (e.g., Dombrowski et al., 2019). One reason to assess these criteria is the fact that many XAI methods often rely on stochastic elements when generating their explanations (e.g., LIME utilizes random samples to train the surrogate model). Crucially, different surrogate models and thus explanations for the same instance (e.g., the same fetus is diagnosed several times) or very similar instances pose a significant problem for the robustness and reproducibility of the XAI method. A third important criterion is sensitivity, which measures how sensitive a ML model's decision is to a change in the instance's feature values (Ribeiro et al., 2016; van Stein et al., 2022). Thereby, a close neighborhood of an instance reflecting possible changes in its feature values is examined regarding the ML model's class assignment. The degree to which a XAI method and its generated surrogate models accurately reveal the sensitivity of an instance is crucial for transparency as it indicates, e.g., how 'close' a class decision was made. In our running example, such a sensitivity analysis could reveal that only a slight increase in the heart rate of a fetus instance would result in a different model prediction from "SUSPECT" to "PATHOLOGICAL". Ideally, also the exact amount of this change should be exactly determined, e.g., that an increase of at least 5 heart beats per minute would change the model's class prediction. Note that sensitivity is different from the robustness criterion discussed before, since robustness measures the (unwanted) change in the explanation of the *XAI method* for similar instances (or even the same instance), while sensitivity aims to measure the effects of (deliberately) introduced changes of feature values on the *ML model* decision. Being able to accurately assess the sensitivity of the class prediction for an instance is highly relevant for a XAI method, since it not only provides information on what changes in feature values lead to a different class assignment, but also sheds light upon the importance of each feature to the model decision.

Based on these performance measures, we can now discuss well-known local XAI methods and review their methodological strengths and weaknesses. To make the model decision of a certain instance transparent, XAI methods propose to approximate the black box ML model locally (i.e., in the neighborhood of a considered instance) with a simpler, symbolic surrogate model (e.g., a linear regression or decision tree). This is typically achieved by first generating artificial sample instances close to the instance to be explained and obtaining the prediction of the black-box ML model for each sample. Based on these samples and their predictions as training data, the surrogate model is trained to approximate the ML model including its DBs in the neighborhood of an instance. After training, this surrogate is used to explain the ML model decision, for example, in the case of a linear regressor by interpreting the feature coefficients. The most well-known

representative of such local XAI approaches is LIME (Local Interpretable Model-agnostic Explanations; Ribeiro et al., 2016) as a model-agnostic method, with a multitude of further approaches building upon it, such as LS (Local Surrogates; Laugel et al., 2018) and LEAP (Local Embedding Aided Perturbation; Jia et al., 2019). In particular, Kim et al. (2023) has to be noted, as they clearly outperform previous local XAI approaches in terms of accuracy/fidelity with their method ROLEX (RObust Local EXplanations).

Despite the strengths of the mentioned approaches, such as (partly) providing good surrogates and thus explanations for selected instances, there are also methodological limitations. A key limitation is that actual DBs are not or not correctly approximated during the generation of the surrogate models. Partly, this is due to the well-known limitations of sampling approaches in higher dimensions. More crucially, local XAI methods and their surrogate models often have difficulties to capture the structure of DBs even locally, especially when there are multiple DBs (which is a common case). Such problems are critical because they often lead to inaccurate explanations for these instances. Based on our analysis of 426 test instances (cf. details in Section Evaluation), with LIME respective ROLEX as many as 215 (>50%) instances respective 75 (>17%) instances in the Fetal Health test dataset are affected by such problems, resulting in their fidelity being severely compromised with often less than 60%. This means that the surrogate model fails to accurately explain the ML model, therefore potentially leading to seriously wrong assessments of the health of the concerned fetus instances. Related to ROLEX, an example for such a case is visualized in Figure 1 with respect to the two features *Uterine Contractions* and *Heart Rate Histogram Width*. The left side of Figure 1 displays the local explanation for the Instance I by ROLEX: For the generation of the surrogate model (dotted line in left side of Figure 1), the closest Training Instance T with a different class than I is identified in a first step. Thereafter, a surrogate model (e.g., a Ridge Regression) is trained based on random samples drawn from a sphere whose center (lying between I and T) and radius is determined via an optimization process. This sphere, illustrated as the circle in Figure 1, aims to encompass the DB lying between I and T (which belong to different classes) and thus pose a suitable ground set for the training samples of the surrogate model. Crucially, this procedure disregards all regions around I that do not lie in the direction of T and could potentially contain other DBs in the neighborhood. Indeed, this occurs in the case of the fetus instance considered, as the right side of Figure 1 showing the actual DBs of the ML model demonstrates. Thereby, several DBs (db_1-db_4 in Figure 1) lie in the neighborhood of I (dotted rectangle in Figure 1) yet all but db_1 are completely neglected by ROLEX. For this reason, the generated surrogate model misclassifies a substantial number of samples drawn from the neighborhood of I, leading to limited accuracy. Furthermore, ROLEX completely neglects the DBs (db_2 and db_3) and CS corresponding to Class "PATHOLOGICAL". Note that usually due to the black box nature of subsymbolic ML models, it is not even known that the actual DBs db_2-db_4 exist at all (which is made transparent by our method proposed in the following). Thus, the explanation provided by ROLEX would also not indicate that the ML prediction for the health status of the fetus I changes to "PATHOLOGICAL" because of only a small decrease of the *Uterine Contractions*. This emphasizes that a valid sensitivity

analysis is hardly possible. Overall, surrogate models – although generated with a local focus – often do not locally represent the actual DBs and CSs of the ML model, and cannot provide valid information on the sensitivity of the instance's classification.

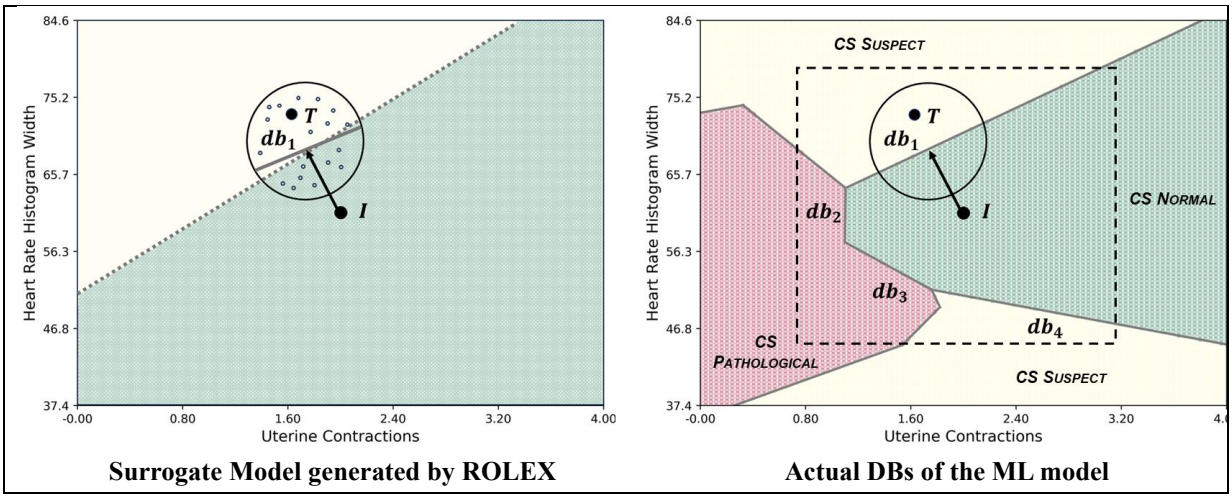


Figure 1: DBs Close to a Fetus Instance (Fetal Health Example)

To conclude, existing local XAI methods, while contributing to a transparent and accurate explanation of ML predictions, suffer from methodological issues, e.g., due to their nature of generating approximative surrogate models or their use of sampling approaches and other stochastic elements. Thus, they exhibit both a limited fidelity and robustness (cf. also Section Evaluation). Moreover, as discussed above, a sensitivity analysis is hardly feasible. With these limitations in mind, we aim to address the research gap for a transparent and accurate local explanation, which is independent from stochastic elements, sampling approaches, or the shortcomings of surrogate models. This allows to not only address fidelity and robustness, but also to rigorously analyze the sensitivity of the model's decision.

3 Theoretical Foundations and Basic Ideas

The paper has two objectives: (1) making the DBs and CSs of a ML classification model fully transparent and (2) determining the sensitivity of a ML classification decision. To address objective (1), instead of generating surrogate models associated with methodological limitations, we propose to represent the DBs and CSs of the ML model in a functional and definite manner, in particular for subsymbolic black box models such as NNs. This leads us to the first proposition of our approach, which we will examine and prove for the case of NNs in the next section.

PROPOSITION 1: DBs and CSs of the ML model to be explained.

1a. The DBs of a ML model can be represented in a functional and definite (well-defined) manner using piecewise linear functions. For ML models that induce piecewise linear DBs (e.g., NNs with ReLU activation function, decision trees, random forests), this corresponds to an exact reconstruction of the DBs, while for ML models with non-piecewise linear DBs (e.g., NNs with sigmoid activation functions), this corresponds to an (arbitrarily exact) approximate reconstruction.

1b. By composing the reconstructed DBs of a ML model represented by piecewise linear functions, the CSs of the ML model are also determined in a functional and definite manner.

If Proposition 1 holds, meaning that the DBs and CSs of a ML model to be explained can be reconstructed in a functional and definite manner, then all instances can be traced exactly and deterministically in the feature space. Creating a good or bad surrogate model is no longer necessary; instead, the DBs and CSs of the ML model are made fully transparent by their functional representation. This has significant advantages as now all DBs close to a considered instance are captured, meaning that no DBs are ignored, or wrong (non-nearest) DBs are selected. Referring to the Fetal Health dataset, Figure 2 (left) shows the *actual* DBs and CSs of the ML model for the two selected Features *Uterine Contractions* and *Abnormal Short-Term Variability* in the intervals [0;12.0] and [8.2;68.3]. For this visualization, no surrogate models or averaging of features values (e.g., Partial Dependency Plots use such simplifications and thus do not visualize the actual DBs) would be necessary. Moreover, the Instances I_1 and I_2 are exemplified. Thereby, it becomes evident that Instance I_1 is not only classified as "NORMAL" (part of CS NORMAL) but is also close to several actual DBs of the ML Model ($db_1 - db_4$ are exemplified in Figure 2). This determination of the position of Instance I_1 and thus the transparency provided can be utilized for further analysis (cf. below, sensitivity). In contrast, Instance I_2 lies deep within a CS PATHOLOGICAL and thus further away from DBs. Additionally, if DBs and CSs can be determined deterministically, then these representations are robust and reproducible, even upon repetition, which directly contributes to the criterion of robustness.

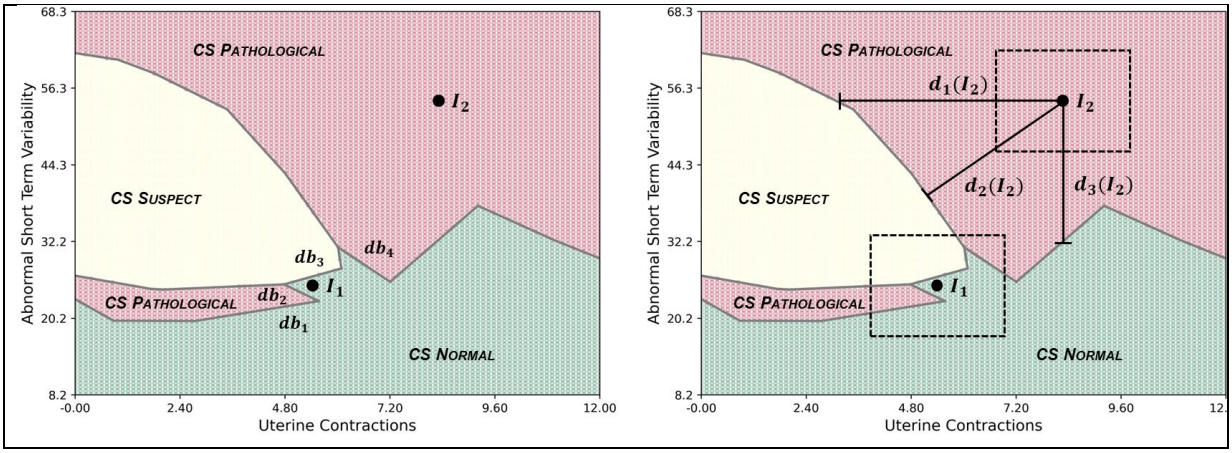


Figure 2. Reconstructing a ML Model (Fetal Health Example)

In general, Proposition 1 aims to enhance the transparency of an individual ML model decision including all its relevant DBs and CSs. We argue that by functionally defining the DBs and CSs, we can not only recognize why an instance is assigned to a class by a ML model, but also understand the role of features and their values for this assignment. Moreover, we can also determine how the feature values of an instance need to be altered to achieve a different class assignment. For that, it is necessary to analyze the proximity of an instance allowing us to precisely examine the sensitivity of the model decision to changes in its feature values. To analyze the sensitivity of instances, we rely on the concept of the neighborhood of an instance which reflects the

potential extent of changes in individual feature values. Figure 2 (right) illustrates exemplary *possible* neighborhoods of both Instances I_1 and I_2 (dotted rectangles).

To determine the effects of altering the instance's feature values in the neighborhood, we propose to analyze whether and how the ML model decision (class assignment) would change and thus, how sensitive the ML model decision is. Precisely, we aim to determine the probabilities of class assignments when the feature values of an instance are altered within this neighborhood. In Figure 2, the ML model decision for Instance I_1 is Class "NORMAL". However, the neighborhood of Instance I_1 also comprises subspaces of the classes "SUSPECT" and "PATHOLOGICAL" to a significant extent. This would be reflected in the probabilities of class assignments for Instance I_1, which are, e.g., (0.464; 0.232; 0.304) for the classes "NORMAL", "SUSPECT" and "PATHOLOGICAL". This shows that while the probability for the predicted Class "NORMAL" is the highest, the sum of the probabilities for the other two classes is higher overall. This transparently discloses the high sensitivity of the ML model decision (i.e., Class "NORMAL"), as even a small decrease of the values of Feature *Uterine Contractions* respective a small increase or decrease of the values of Feature *Abnormal Short-Term Variability* would result in a different model decision. For assessing the prediction of the ML model, for example, within a decision support context, this is of high importance as even a slight change in the feature values reveals that the ML model is more likely to assign the Fetus Instance I_1 to one of the other two serious classes rather than the (unproblematic) Class "NORMAL". Moreover, it is possible for the neighborhood of an instance to not contain any DBs (cf. Instance I_2 in Figure 2). To address transparency and sensitivity in such a scenario as well, we propose to determine the distances and directions of the instance to the next DBs. Methodically, this represents the maximum possible change in the feature values of an instance before the current ML model decision changes. Figure 2 illustrates this for the Instance I_2, where the distances to the nearest DB $d_2(I_2)$ as well as the next DBs $d_1(I_2)$ and $d_3(I_2)$ regarding both features are illustrated. Based on this discussion, we formulate Proposition 2 for sensitivity.

PROPOSITION 2: Sensitivity of a ML model decision (regarding a considered instance).

2a. When altering the feature values of an instance, which defines the neighborhood of this instance, the probabilities of class assignments can be calculated in an exact and definite manner.

2b. The distances and directions of an instance to the nearest DBs can be calculated in an exact and definite manner.

If Proposition 2a holds, it is possible to precisely determine how much the change of its feature values affects the probability of the instance being assigned to each class. This is not possible for existing XAI methods, as neither DBs nor CSs of the ML model were represented in a functional and definite manner. For example, the Softmax value of an instance is frequently used as a 'probability' for its class assignment. However, methodologically, although often argued, the Softmax value neither represents a probability for a class assignment nor is it a valid measure for any sensitivity (e.g., Hendrycks & Gimpel, 2017). In contrast, analyzing the characteristics of an instance's neighborhood allows for an explanation of why a model decision is determined as it

is. Further, if Proposition 2b holds, the position of the instance related to the next DBs and CSs can be precisely determined (e.g., distance $d_2(I_2)$). However, the advantage goes beyond the mere calculation of this position. By considering the distances and directions of the instance to the nearest DBs, the contribution of each feature to the ML model decision can be rigorously analyzed. For instance, a small decrease of the values of Feature *Uterine Contractions* is crucial for the ML decision for Fetus Instance I_1 because it would then be classified as "PATHOLOGICAL". This underscores the importance of a certain change (i.e., direction) in the value of this feature (as opposed to an undirected feature importance) and is of high value for assessing a ML prediction in decision support. We now proceed by presenting our method for reconstructing the DBs and CSs as well as the sensitivity measures.

4 Explaining Machine Learning Models by Local Reconstruction

4.1 Meta Method of EXPLORE

In this section, we first describe our model-agnostic meta method of EXPLORE which we subsequently instantiate for feedforward NNs in a model-specific manner. The latter is used to prove whether Proposition 1 holds for NNs. To reconstruct the DBs and CSs of a ML model, the following steps are necessary:

Step 1. Formal definition of the ML model: In a first step, we formally define the ML model to be explained. To this end, we conceive the ML model as a mathematical function $f: X \to Y$ which maps any instance $x \in X$ in the feature space onto a target value $f(x) = y \in Y$ in the output space. The feature space X reflects the features and their values. Hence, X might be a discrete or continuous space, or a mixture thereof. Similarly, Y represents the output space of the ML model, which might also be discrete (e.g., the set of classes that a decision tree can predict) or continuous (e.g., a simplex which contains the output values of a NN prediction).

Step 2. Identification of linear regions in the feature space: As stated in Proposition 1, we aim to represent DBs by piecewise linear functions. Therefore, we need to determine in which regions of the feature space the DBs are piecewise linear (i.e., behave like a straight line) and where the break points between these linear segments of the DBs lie. In Figure 2, the piecewise linear segments $db_1 - db_4$ as well as the breakpoints between them are shown. In fact, these segments correspond to the so-called linear regions (denoted by A_u) of the ML model, on which the ML model behaves piecewise linearly. More precisely, the function f is given by an affine linear (or even constant) function f_u on each disjoint linear region $A_u \subset X$ in the feature space. As this is our basis to determine the DBs, we have to identify (locally) all linear regions of the ML model in a formal and definite manner. For ML models that naturally generate piecewise linear regions in the feature space (e.g., NNs with ReLU activation function, decision trees), this corresponds to an exact reconstruction of the ML model. Other ML models (e.g., NNs with non-piecewise linear activation functions) can be approximated with arbitrary exactness (Hornik et

al., 1989; Hu et al., 2020) by such a piecewise linear function consisting of linear regions A_u together with affine linear functions f_u.

Step 3. Determining DBs in the feature space: The subdivision of the feature space X (or a part thereof) into linear regions A_u together with their respective affine linear functions f_u provides *complete* information about f (and hence, the ML model to be explained), because the union of the linear regions covers the whole feature space. Importantly, they can be used to recognize and understand where exactly the ML model changes its (class) decision, enabling the determination of the DBs of the ML model in a functional and definite manner. More precisely, it can be analyzed which $x \in A_u \subset X$ form the DB up to which the respective functions f_u induce the same class prediction. For example, in the case of a NN, the affine linear function f_u would map all $x \in A_u$ that lie on a DB onto an output vector with at least two equal entries. This means that x lies on a DB between the classes to which the equal entries correspond.

Step 4. Building CSs in the feature space: Finally, the feature space can be subdivided into CSs, based on the DBs identified in Step 3. As a result, even though each CS corresponds to a single class, it is enclosed by potentially multiple DBs adjoining different classes. Because DBs constitute hypersurfaces in the feature space and can thus be described by (a system of) equalities, CSs can typically be represented by (a system of) their respective inequalities.

4.2 EXPLORE-Method for Feedforward Neuronal Networks

In this section, we instantiate our meta method for the relevant and complex case of feedforward NNs.

Step 1: Formal definition of the NN

In the following, we consider an arbitrary feedforward NN mapping instances with $m \in \mathbb{N}$ features onto $n \in \mathbb{N}$ classes in the output space. Thus, we denote the NN by a function $f: X \to \mathrm{Y}$ with feature space $X \subset \mathbb{R}^m$ and output space $Y \subset \mathbb{R}^n$. A NN is well-known as a concatenation of layers (Bishop, 2006; Goodfellow et al., 2016), each consisting of a linear transformation followed by a non-linear activation function (e.g., ReLU). Hence, for an instance $x_0 \in X$ the result x_{i+1} after the i-th NN layer can be computed as $x_{i+1} = g_i(W_i x_i + b_i)$. Here, W_i denotes the weight matrix, b_i the bias, and g_i the activation function (applied entry-wise) of the i-th layer. Typically, the activation function of the output layer is a Softmax function, for the other activation functions in the intermediate layers, (Leaky) ReLU, sigmoid, or tanh are common choices.

Step 2: Identification of the linear regions

One of our main ideas is to represent DBs by piecewise linear functions. Therefore, we need to find the segments of the feature space on which the DBs are linear and hence, the break points between these piecewise linear segments. This can be done by identifying the linear regions of the NN. NNs with piecewise linear activation functions (such as Parametric ReLU, Leaky ReLU, ReLU) can be exactly reconstructed by a piecewise linear function on their feature space (cf. Krapf et al., 2024; Sattelberg et al., 2023; Zhang et al., 2018). For NNs with non-piecewise linear activation functions (such as, e.g., sigmoid or tanh) we propose to approximate the activation function with a piecewise linear function. Indeed, any NN can be approximated arbitrarily well

by a NN with piecewise linear activation functions (Hu et al., 2020). Following this idea, we introduce Definition 1 allowing us to reconstruct NNs as piecewise linear functions.

Definition 1 (NNs as Piecewise Linear Functions). Let $X \subset \mathbb{R}^m, Y \subset \mathbb{R}^n$ and $f: X \to \mathrm{Y}$ be the function representing a NN with piecewise linear activation functions. Then, f has the piecewise linear form

$$f(x) = \begin{cases} T_1 x + c_1, & if\ M_1 x \leq l_1, \\ T_2 x + c_2, & if\ M_2 x \leq l_2, \\ & \dots \\ T_t x + c_t, & if\ M_t x \leq l_t \end{cases} \tag{1}$$

with $T_u \in \mathbb{R}^{n \times m}, M_u \in \mathbb{R}^{n_i \times m}, c_u \in \mathbb{R}^n, l_u \in \mathbb{R}^{n_i}$ for $t, n_u \in \mathbb{N}, 1 \leq u \leq t$.

The systems of n_u inequalities $M_u x \leq l_u$ in Term (1) divide the feature space X into t polytopes $A_u = \{x \in X \mid M_u x \leq l_u\}$, which are disjoint (in the sense that the m-dimensional Lebesgue measure of their intersection vanishes, i.e., they only have their boundaries in common) and satisfy $X = \bigcup_{u=1}^{t} A_u$. On each polytope A_u, the NN represented by f is given by an affine linear function defined by T_u and c_u. Thus, f is a piecewise linear function with respect to the linear regions A_u. As described above, besides NNs with piecewise linear activation functions, we can also deal with NNs with other (i.e., non-piecewise linear) activation functions (such as, e.g., sigmoid or tanh) by approximating the activation function with a piecewise linear function (Hu et al., 2020).

Step 3: Determining the DBs

We now leverage the piecewise linear form of the NN from Step 2 to determine its DBs. To this end, we begin by formally characterizing the points that lie on a DB in the *output space* using the (in)equalities in the following definition and prove their validity. Thereafter, we reconstruct the DBs in the *feature space* as the preimage of the output space DBs under f.

Definition 2 (Output Space DBs). Let again $X \subset \mathbb{R}^m, Y \subset \mathbb{R}^n$ and $f: X \to \mathrm{Y}$ be the function representing a NN. If $y \in Y$ satisfies $y_i - y_j = 0$ for $i \neq j$ and $(y_j =) y_i \geq y_k$ and all $k \neq i, j$ (with y_i denoting the i-th vector entry of $y \in Y \subset \mathbb{R}^n$), then y lies on an output space DB between the classes i and j. Therefore, the DBs between two (different) classes i, j in the output space Y are given by

$$db_{i,j}{}^{o} = \{y \in \mathrm{Y} \mid y_i - y_j = 0 \ \wedge \forall k \neq i, j: y_i \geq y_k\} \tag{2}$$

for all $1 \leq i \neq j \leq n$. By this definition, the symmetry $db_{i,j}{}^{o} = db_{j,i}{}^{o}$ holds.

Proof. In NNs used for classification (with at least two classes, i.e., $n \geq 2$), the prediction for an instance with output value y is determined by selecting the class with the highest corresponding vector entry of y. Therefore, the DB between the classes i and j in the output space lies where their corresponding entries are equal, i.e., where $y_i - y_j = 0$ holds. However, this equation only defines a DB where no other class k attains a higher value than y_i and y_j, i.e., where $(y_j =) y_i \geq y_k, \forall k \neq i, j$ holds. Thus, the output space DB between the classes i and j is given by Term (2). Alternatively, the output space DB can also be described by the system of (in)equalities

$$\begin{aligned} y_i - y_j &= 0, \\ y_k - y_i &\leq 0 \; for \; k \neq i, j. \end{aligned} \tag{3}$$

□

On this basis, we next propose a system of (in)equalities which represents the DBs in the *feature space*. This is a key result in this section, which we first formulate and prove for one linear region A and two classes i, j in the following lemma. Thereafter, we generalize this result in Definition 3 and characterize all DBs (between all classes) in the whole feature space.

Lemma 1 (Feature Space DBs). Let again $X \subset \mathbb{R}^m, Y \subset \mathbb{R}^n$ and $f: X \to \mathrm{Y}$ be the function representing a NN. Moreover, let A be a linear region of f (as in Definition 1), and ${db_{i,j}}^o$ be a DB in the output space between the (different) classes i and j. Then, the corresponding DB in the linear region A is characterized by the system of (in)equalities

$$\begin{aligned} v_{i,j} \cdot T \cdot x + v_{i,j} \cdot c &= 0, \\ v_{k,i} \cdot T \cdot x + v_{k,i} \cdot c &\leq 0 \end{aligned} \tag{4}$$

for $x \in A$. In this term, $v_{i,j} \in \mathbb{R}^n$ is defined as the vector with 1 in the i-th entry, -1 in the j-th entry and 0 otherwise, and T, c are defined as the affine linear mapping representing f on the linear region A.

Proof. According to Definition 1, $f(x) = Tx + c$ holds for $x \in A$. Then, according to Term (3), the DB ${db_{i,j}}^o$ in the output space can be written by $v_{i,j} \cdot y = 0, v_{k,i} \cdot y \leq 0$ for $y \in Y, k \neq i, j$. For $y = f(x) = T \cdot x + c$, this system of (in)equalities can directly be rewritten as in Term (4).

□

Note that it is possible that the set of $x \in A$ satisfying these (in)equalities might also be empty, which is the case if (and only if) the linear region A contains no DB between the classes i and j. By considering all classes and linear regions, we can now obtain the complete system of (in)equalities which represents all DBs in the whole feature space X.

Definition 3 (Feature Space DBs). The DBs in the feature space are given by the system of (in)equalities

$$\begin{aligned} v_{i,j} \cdot T_u \cdot x + v_{i,j} \cdot c_u &= 0, \\ v_{k,i} \cdot T_u \cdot x + v_{k,i} \cdot c_u &\leq 0, \\ M_u x &\leq l_u \end{aligned} \tag{5}$$

for all $1 \leq i \neq j \leq n$, $k \neq i, j$ and $1 \leq u \leq t$. According to Definition 1, instance x satisfying the inequalities $M_u x \leq l_u$ is equivalent to instance x lying in A_u. We denote the DBs, i.e., the sets of all $x \in X$ which satisfy this system of (in)equalities for the linear region A_u and classes i and j, by $db_{u,i,j}$.

Step 4: Building the CSs

The DBs as described in Term (5) can now be used to derive the CSs in the feature space, in which the NN attains one single class. Because the *equalities* in Term (5) divide the linear region into areas corresponding to the classes i and j, the CSs can be exactly represented by using their respective *in*equalities.

Definition 4 (Class Subspaces). The feature space can be divided into (convex) subspaces, in which the NN attains only class i, by the system of inequalities (including all inequalities iterating over $k \neq i$)

$$\begin{aligned} v_{k,i} \cdot T_u \cdot x + v_{k,i} \cdot c_u \leq 0, \\ M_u x \leq l_u \end{aligned} \tag{6}$$

for all linear regions $1 \leq u \leq t$ .We denote the CS consisting of the set of all $x \in X$ which satisfy these inequalities for the linear region A_u and class i, by $cs_{u,i}$.

For one single u, this system of inequalities exactly represents the CS in the linear region A_u (defined by $M_u x \leq l_u$), in which the NN predicts class i. Compared to Term (5), the class index j is now attained by the running index k in this definition, as the output value (of any $x \in cs_{u,i}$) for class i must be higher than that of *all* other classes, including j. Note that by this definition it is possible that two CSs $cs_{u_1,i}$ and $cs_{u_2,i}$ of the same class i adjoin each other if their respective linear regions A_{u_1} and A_{u_2} adjoin each other.

4.3 Measures for Sensitivity

In the previous section, we have proven Proposition 1 by establishing the piecewise linear structure of a NN and reconstructing its DBs and CSs in a functional and definite manner. We now shift our focus on analyzing the sensitivity of ML model decisions (Proposition 2), for which our functional representation of the DBs and CSs is employed. To this end, we introduced the idea of a neighborhood of an instance, which reflects the potential extent of changes in its individual feature values. We then propose two novel sensitivity measures grounded on this notion.

For an instance x we define the neighborhood Ω_x as the subset of the feature space X which contains all changes of feature values being part of the sensitivity analysis. Moreover, we also consider a probability distribution p_{x} on Ω_{x}, which describes the probabilities of feature value changes in the neighborhood. Therefore, we first introduce a probability-theoretic grounding which serves as a rigorous basis for our sensitivity measures.

Definition 5 (Probability-Theoretic Grounding). Let $x \in X$ be the considered instance and let $\Omega_{\mathrm{x}} \subset X$ be the neighborhood of x together with a probability density function p_{x}. Then the random experiment is defined by the probability space $(\Omega_{\mathrm{x}}, \mathfrak{B}(\Omega_{\mathrm{x}}), P_{\mathrm{x}})$ where $\mathfrak{B}(\Omega_{\mathrm{x}})$ denotes the Borel-σ-Algebra of Ω_{x}. For any set $A \in \mathfrak{B}(\Omega_{\mathrm{x}})$ the probability measure P_{x} is defined as

$$P_{\mathrm{x}}(A) \coloneqq \int_A p_{\mathrm{x}}(z)dz. \tag{7}$$

A natural choice for neighborhood Ω_x can be a cube around the instance x with width ε for each feature, i.e., $\Omega_x = \{x' \in X \mid |x'_i - x_i| \leq \varepsilon/2\,, \forall i = 1, \ldots, m\}$. By this definition, positive or negative feature value changes of $\varepsilon/2$ would be possible for each feature. If a uniform probability distribution is chosen for p_{x} (i.e., $p_x(x') = \varepsilon^{-m}$, $\forall x' \in \Omega_x$), then all possible changes in the neighborhood would be equally probable. In general, however, the definition of Ω_{x} and p_{x} is

fully adaptable to the context of the needed sensitivity analysis. Other choices for Ω_x could be spherical or cuboid-shaped neighborhoods, and for p_x various types of probability distributions such as Gaussian are conceivable. We now formally define our two sensitivity measures based on this theoretical grounding.

Definition 6 (Class Probability). Let $x \in X$ and $(\Omega_x, \mathfrak{B}(\Omega_x), P_x)$ be as in Definition 5. We define the class probability p_i (with respect to f) by the probability mass of the subset $\Omega_{x,i} \subset \Omega_x$ where f predicts class i.

$$p_i \coloneqq \int_{\Omega_{x,i}} p_x(z)\, dz \tag{8}$$

In the previous section we pointed out that the class decision of a NN f corresponds to the highest value in its output vector, i.e., it is given by $\operatorname{argmax} f(x)$ for $x \in X$. Using our notation of CSs, the preimage $(\operatorname{argmax} f)^{-1}(i)$ is equal the union of all CSs of class i over all linear regions $\bigcup_{u=1}^{t} cs_{u,i}$. Since we only consider the neighborhood Ω_x in Term (8), $\Omega_{x,i} = \Omega_x \cap (\operatorname{argmax} f)^{-1}(i)$ holds. Overall, p_i describes the probability of f attaining class i in the neighborhood Ω_x. Importantly, our functional descriptions of the CSs can be used to exactly calculate p_i, which we demonstrate in Lemma 2. Before, we introduce our first sensitivity measure by assembling the class probabilities of all classes.

Definition 7 (Class Probability Vector (CPV)). Let $x \in X$, $(\Omega_x, \mathfrak{B}(\Omega_x), P_x)$, and f be as in Definition 6. We define the Class Probability Vector (CPV) $m_x \coloneqq (p_1, \dots, p_n)$ with p_i denoting the class probability of class i as defined above. Therefore, the vector m_x contains the probabilities of f predicting each class in Ω_x.

By proposing the CPV in Definition 7, we can now assess the sensitivity of an instance in the sense of Proposition 2a. Now following Proposition 2b, we proceed by formally defining our second sensitivity measure given by the distances and the directions of an instance to the DBs of f.

Definition 8 (Distance and Direction to DBs). We define the distance $d_{db}(x)$ of instance $x \in X$ to the DBs of f as the minimum of its distances to all boundaries $db_{u,i,j}$ for all linear regions A_u and classes i, j:

$$d_{db}(x) = \min_{u,i,j} dist(x, db_{u,i,j}) \tag{9}$$

Following common notation, for the instance x and any set S, the distance is defined as the infimum of all distances of x to any point in the set S (i.e., $dist(x, S) = \inf\{\|x - s\| \mid s \in S\}$). Note that in the case $S = db_{u,i,j}$, this infimum is always attained at a point $x_{db,min} \in db_{u,i,j}$, since $db_{u,i,j}$ is always a closed subspace of a hyperplane. The vector $\overrightarrow{d_{db}}(x) \coloneqq x_{db,min} - x$ thus defines the direction of x to the closest point on a DB. For the distance between x and the DBs with respect to a certain feature/dimension $1 \leq e \leq n$, we define

$$d^e_{db}(x) = \min_{u,i,j} dist(x, db^{e,x}_{u,i,j}), \tag{10}$$

where $db_{u,i,j}^{e,x}$ denotes a DB of the restricted one-dimensional feature space consisting of all points on a DB whose feature values — except for feature e — are equal to the feature values of x. Formally, $db_{u,i,j}^{e,x}$ can be defined as $db_{u,i,j}^{e,x} = \{z \in db_{u,i,j} \mid \forall d \neq e: (z)_d = (x)_d\}$. Intuitively, $d_{db}^{e}(x)$ describes by how much the value of the feature e has to be changed to achieve a different class assignment by f without changing any other feature values of instance x (i.e., ceteris paribus).

Note that the distance to any DB of interest can analogously be determined by adjusting the minimizing indices in Term (9). For example, DBs that adjoin a specific class or that lie in a specific direction could be analyzed in this manner as well. Before evaluating these measures by concretely assessing the sensitivity of the class decision of instances from the Fetal Health dataset, we prove the validity of the computation of both sensitivity measures.

Lemma 2 (Validity). Let $x \in X$, $(\Omega_x, \mathfrak{B}(\Omega_x), P_x)$, and f be as in Definition 6, then the computation of both sensitivity measures is valid, in the sense that they can be calculated exactly by utilizing the functional representation of DBs and CSs of the model f (cf. Definition 3 and 4).

Proof. Since the Definitions 1-4 and Lemma 1 also hold for any subset of the feature space, we can set $X = \Omega_x$ without loss of generality. Thus, $\Omega_{x,i} = (\operatorname{argmax} f)^{-1}(i) = \bigcup_{u=1}^{t} cs_{u,i}$ holds in Term (8) and according to Definition 4, all CSs $cs_{u,i}$ are functionally represented by the systems of inequalities in Term (6), and their representation is valid. Therefore, p_i is given as the sum of the integrals over the linear regions of $X = \Omega_x$:

$$p_i = \int_{\Omega_{x,i}} p_x(z)\, dz = \sum_{u} \int_{cs_{u,i}} p_x(z)\, dz \tag{11}$$

This shows the lemma for the CPV m_x. For the distance of x to the nearest DB $d_{db}(x)$, we first consider the DBs $db_{u,i,j}$ for all linear regions A_u and classes i, j as described in Definition 3. For each DB $db_{u,i,j}$, the system of (in)equalities as in Term (5) defines a closed subset of a hyperplane in X, for which the distance $dist(x, db_{u,i,j})$ can be calculated by applying the orthogonal projection π of the associated hyperplane onto x. If the projection $\pi(x)$ lies in $db_{u,i,j}$ (which is a closed subset of the hyperplane), then $\pi(x)$ is the nearest point and $dist(x, db_{u,i,j}) = |x - \pi(x)|$ holds. If $\pi(x)$ does not lie in $db_{u,i,j}$ this procedure can be reapplied in the lower-dimensional projection space $\pi(X)$ until the projection point lies in the hyperplane, which is guaranteed if the dimension of the projection space reaches zero. As a result, the distance $dist(x, db_{u,i,j})$ can be calculated exactly for any combination of indices u, i, j, and taking the minimum yields $d_{db}(x)$. This also shows the validity of $d_{db}^{e}(x)$ because this argument also holds for the (one-dimensional) restriction of the feature space $X^e = \{z \in X \mid \forall d \neq e: (z)_d = (x)_d\}$.

□

5 Evaluation

In this section, we evaluate our method. To this end, we first briefly describe three real-world datasets that we use in our experiments and discuss the existing XAI methods against which we compare the performance of EXPLORE. More precisely, we first focus on the criterion fidelity and present the LDA fidelity measure as proposed in Kim et al. (2023). We next focus on the criterion robustness, and then on sensitivity using our measures introduced in the previous section. For all three criteria we provide empirical evidence and – if possible – theoretical guarantees, thus evaluating our approach from a methodological perspective.

5.1 Description of the Datasets and Competing XAI Methods

To evaluate our method, we used three datasets because they are related to critical decision situations in healthcare, loan approval, and criminal recidivism and thus are already known in XAI literature. If ML model predictions are used for decision support in these domains, they must be treated with great caution as they have potentially profound consequences for a person's life. The first dataset used in this evaluation is the Fetal Health dataset which we introduced in the background section. The dataset has five medical features, based on which the health of a fetus is predicted as one of the three classes "NORMAL", "SUSPECT", or "PATHOLOGICAL". We use 80 percent (i.e., 1,700) of the available 2,126 data instances for model training, the remaining 426 instances are used as test instances in our experiments. As second dataset, we use a version of the Prosper dataset (https://www.kaggle.com/datasets/henryokam/prosper-loan-data) which contains data from a loan company about previous loans and their default status (i.e., whether they defaulted or not). Based on this data, a ML model can be trained and used to predict the future status of a loan application, e.g., if it will be paid back or defaulted, and thus provide a decision support for the company if an application should be approved or not. For this dataset, we consider six features regarding the loan and the applicant, and six classes of the loan outcome. Again, we split the 3,000 data instances into 80 percent training data (2,400 instances) and 20 percent test data (600 instances). As a third dataset, we use a version of the COMPAS dataset (https://www.kaggle.com/datasets/danofer/compass) that contains information on former criminals and their risk of recidivism. A ML model can thus be trained and used to predict the likelihood and point in time of reoffending based on personal information about the former inmate. For this dataset, we consider six features and eight classes and once again split the 3,000 considered instances into training and test datasets in an 80:20 ratio. For each dataset, we trained a feedforward NN with ReLU activations which we aim to explain with EXPLORE and the two competing local XAI methods from literature considered in our empirical evaluation. First, we employ the well-known method LIME from Ribeiro et al. (2016). According to the publicly available code, the surrogate model was instantiated as a Ridge regression in this evaluation. As a second competing method, we consider ROLEX from Kim et al. (2023) which significantly outperformed existent local XAI methods. Following the authors' suggestions as well as their original code implementation (including the configuration for ROLEX), a linear Ridge regression model is used in the case of linear DBs and a non-linear decision tree model in the case of

non-linear and multiclass DBs in the experiments. For assessing EXPLORE, no configuration is necessary as it is deterministic in nature. For calculating sensitivity, we used a cube-shaped local neighborhood with a side length of 10 percent (other percentages are also possible) of the respective features total value space in positive and negative direction.

To evaluate our method, we implemented EXPLORE in Python, which was used to generate the reported results. For the three datasets considered, the runtime of EXPLORE was few seconds per data instance/local explanation and thus comparable to LIME and ROLEX.

5.2 Assessing Fidelity, Robustness and Sensitivity

To begin with fidelity, the LDA score as presented by Kim et al. (2023) is defined as the average of the local fidelity scores of all instances with at least one DB nearby. Hereby, the local fidelity score for a test instance $x_i \in X$ and its local explanation $\bar{f}_{x_i}$ for the ML model f is defined as the accuracy (cf. Kim et al., 2023)

$$LocalFid\left(\bar{f}_{x_i}, x_i\right) = Acc_{z \in Z}\left(f(z), \bar{f}_{x_i}(z)\right). \tag{12}$$

In this term, Z is a set of random samples drawn from the feature space close to x_i. As a result, the local fidelity is 1 if the local explanation $\bar{f}_{x_i}$ predicts the same class as the ML model f on all random samples. Finally, the LDA score is calculated by the average of the local fidelities of all instances x_i for which f predicts at least two different classes on the set of samples, i.e., $\exists z_1, z_2 \in Z: \operatorname{argmax} f(z_1) \neq \operatorname{argmax} f(z_2)$. If this condition holds, then there is a least one DB near x_i since different classes are predicted by f. As a result, fidelities of 'trivial' instances without any nearby DBs are excluded in the calculation of $LocalFid$. Table 1 presents the LDA scores on all three datasets, indicating that both LIME and ROLEX have scores well below 1. This exposes not only the mismatch between the ML model and the XAI surrogate model for many instances but also potentially highly problematic explanations being generated on inaccurate surrogates.

	ROLEX	LIME	EXPLORE
Fetal Health	0.8885	0.6864	1.000
Prosper	0.6813	0.6549	1.000
COMPAS	0.6237	0.2770	1.000

Table 1. LDA Scores

Our method achieves a fidelity of 1 which can be substantiated by establishing a theoretical proof that guarantees a perfect LDA score of 1: Let Z be the set of all random samples collected and $z \in Z$. Utilizing our representation of CSs, we are able to identify the (unique) CS $cs_{u,i}$ of the ML model f containing z by checking for which linear region A_u and class i the corresponding system of inequalities is satisfied by z (cf. Definition 4). Since the inequalities describing the CS are valid as a direct consequence of Lemma 1, the class predicted by f agrees with the one associated to $cs_{u,i}$, namely i. Since this holds for all samples $z \in Z$, the local fidelity of any instance and hence, the LDA score of EXPLORE must always equal 1.

To evaluate the criterion of robustness, we need to assess the stability of the XAI method's explanations by measuring the extent to which the explanations of similar instances match. Thus, we generate explanations for a certain number $J \in \mathbb{N}$ of similar artificial instances $x_{i,j} \in X$ for each test instance x_i. For the results in Table 2, we generated five (i.e., $J = 5$) artificial instances whose feature values differ by at most 3.5 percent of the respective features total value space, for each method and dataset. Following Dombrowski et al. (2019), the robustness measure is defined as the share of mismatches between all local explanations of similar instances $\bar{f}_{x_{i,j}}$ and the original explanation $\bar{f}_{x_i}$ on a set a of random samples Z:

$$Rob\left(\bar{f}_{x_i}, x_i\right) = 1 - \frac{1}{J}\sum_{j=1}^{J} Acc_{z \in Z}\left(\bar{f}_{x_i}(z), \bar{f}_{x_{i,j}}(z)\right) \tag{13}$$

In case of perfect robustness, the prediction of the explanation models $\bar{f}_{x_{i,j}}$ is consistent with the prediction of the original explanation model $\bar{f}_{x_i}$ on any random sample $z \in Z$. Then, the accuracy $Acc_{z \in Z}\left(\bar{f}_{x_i}(z), \bar{f}_{x_{i,j}}(z)\right)$ would be 1 for each explanation $\bar{f}_{x_{i,j}}$, leading to the optimal score $Rob\left(\bar{f}_{x_i}, x_i\right) = 0$ for x_i. In Table 2, our empirical analysis on all three datasets indicates that the robustness of ROLEX is limited, especially for incorrectly classified instances, while LIME (at first glance) seems to have high robustness. However, it should be noted that according to Table 1, LIME exhibits partially very poor fidelity, resulting in "robust" poor surrogate models being generated for many instances.

	Correctly classified instances			**Incorrectly classified instances**		
	ROLEX	**LIME**	**EXPLORE**	**ROLEX**	**LIME**	**EXPLORE**
Fetal Health	0.129	0.002	0.0	0.140	0.0	0.0
Prosper	0.190	0.0	0.0	0.234	0.0	0.0
COMPAS	0.117	0.065	0.0	0.175	0.068	0.0

Table 2. Robustness Scores

Our method achieves a robustness of 0 which can again be substantiated by providing a theoretical proof for the perfect robustness: Let $x_i \in X$ be the considered test instance and let $x_{i,j} \in X$ be an (artificial) instance in close proximity of x_i. By applying our proof for the perfect fidelity of EXPLORE to both explanations $\bar{f}_{x_{i,j}}$ and $\bar{f}_{x_i}$, we can deduce that the class predictions of $\bar{f}_{x_{i,j}}$ and $\bar{f}_{x_i}$ are equal to the class predictions of f on all samples $z \in Z$. This directly implies that the class predictions of $\bar{f}_{x_{i,j}}(z)$ and $\bar{f}_{x_i}(z)$ are also equal for all samples $z \in Z$. As a result, $Acc_{z \in Z}\left(\bar{f}_{x_i}(z), \bar{f}_{x_{i,j}}(z)\right) = 1$ for an arbitrary artificial instance $x_{i,j}$ and $Rob\left(\bar{f}_{x_i}, x_i\right) = 0$ follows.

Instance number	0	78
Ground truth class	Class "NORMAL"	Class "NORMAL"
ML model prediction	Class "NORMAL"	Class "SUSPECT"
Five feature values	(133.0, 0, 46.0, 69.0, 3.0)	(135.0, 0.0, 58.0, 53.0, 1.0)

Nearest point on DB		(136.6, 0.3, 45.6, 67.8, 2.8)					(134.9, 0.0, 57.8, 52.8, 1.0)				
Distance to DB (overall)		3.84					0.30				
Class probabilities (CPV)		(0.939, 0.061, 0.0)					(0.495, 0.501, 0.0)				
Feature		1	2	3	4	5	1	2	3	4	5
Feature-wise distances	Negative direction	-	-	-	-	-	0.4	-	0.3	1.6	0.7
	Positive direction	-	5.0	16.6	-	-	9.9	2.8	33.3	47.0	1.7

Table 3. Sensitivity Measures for Fetal Health Instances 0 and 78

Finally, we demonstrate our two measures for the sensitivity of a class prediction in the case of two selected instances from the Fetal Health dataset. An evaluation of LIME and ROLEX is not feasible here, as neither of these XAI methods is able to recognize the actual DBs or the CSs of the ML model. Instead, this is feasible based on the functional and definite representation of the DBs and CSs in our method. The considered instances are selected because they exhibit vastly different and interesting values regarding our sensitivity measures, which are presented in Table 3 and visualized in Figure 3 (for two feature dimensions). Instance 78 lies very close to the nearest DB (with a distance of 0.30) and has class probabilities (CPV) of (0.495, 0.501, 0.0), both indicating that the class prediction of Instance 78 is very sensitive even to slight changes in the feature values. Moreover, training instances with ground truth Class "SUSPECT" (denoted by the symbol '**x**' in Figure 3) lie in the CS "NORMAL", and vice versa, making transparent that the DBs were not learned accurately by the model. In contrast, the class prediction of Instance 0 is not very sensitive, as the clearly distinct CPV (0.939, 0.061, 0.0) and the rather high distance to the nearest DB of 3.84 suggest. Note, that the probability of 0.061 for Class "SUSPECT" comes from features that are not visualized in Figure 3.

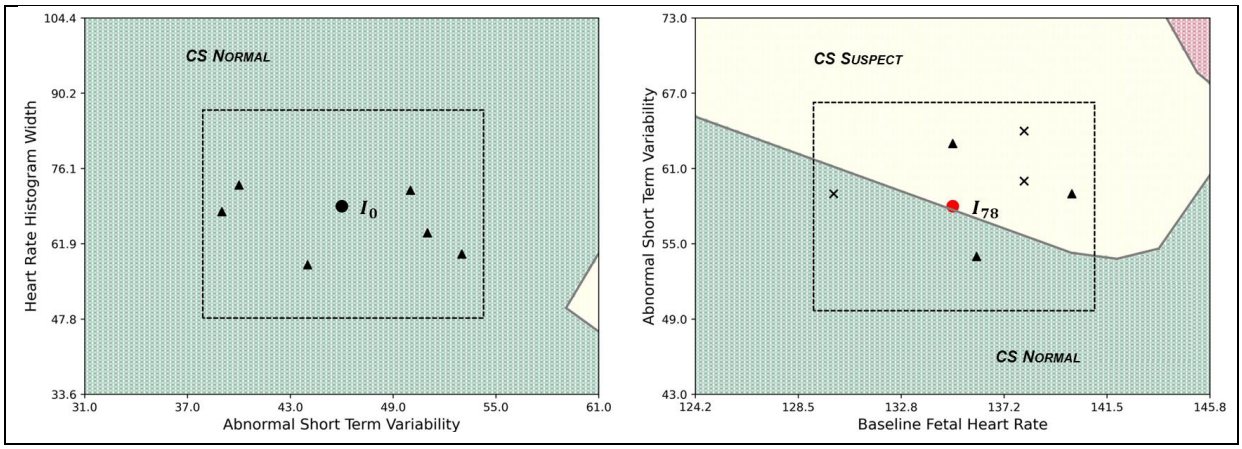


*Figure 3. Sensitivity of the two Instances 0 and 78 from the Fetal Health Dataset (Ground Truth of Selected Training Instances: Class "NORMAL": **Δ**, Class "SUSPECT": x)*

6 Discussion, Implications, Limitations and Future Research

In this paper, we proposed a novel local XAI methods as well as two sensitivity measures. Based on these two contributions, we will discuss the main results and implications of our paper.

6.1 Reconstructing DBs and CSs of a ML Model

First, we introduced EXPLORE, a method to reconstruct the DBs and the CSs of a ML model in a functional and definite manner, thus making the ML model fully transparent. We provide formal proofs for the validity of this representation and thus for Proposition 1, which ─ together with the fact that our method is deterministic and does not rely on stochastic elements ─ guarantees both, the fidelity and robustness of EXPLORE from a theoretical perspective. As a result, our method not only perfectly matches the ML model in the local neighborhood of any instance to be explained but is also robust and reproducible upon repetition. In the evaluation, we substantiate our theoretical guarantees with empirical evidence by measuring the fidelity and robustness of EXPLORE and comparing it to the competing local XAI methods LIME and ROLEX on three real-world datasets. Indeed, Table 1 shows that EXPLORE clearly outperforms the competing methods regarding fidelity, even in challenging settings such as the COMPAS dataset where ROLEX and LIME attain a LDA score of 0.623 and 0.277 respectively. Further, our analysis shows that EXPLORE provides perfect robustness for similar instances, which is substantiated by our robustness score of 0 across all datasets. Interestingly, LIME consistently achieves the lowest fidelity scores but a very high robustness (cf. Table 2). Thus, it yields robust, but inaccurate explanations for similar instances (which is highly critical). Instead, ROLEX outperforms LIME regarding fidelity, but is less robust compared to LIME and our method. More precisely, the robustness scores for 20 fetus instances in the Fetal Health dataset are even higher than 0.3 for ROLEX, indicating that the generated explanations for these and similar instances vary significantly upon repetition which is also highly critical. These theoretical and empirical results form the basis of the first contribution of our paper corresponding to our Proposition 1. By reconstructing the DBs and CSs of a ML model in the neighborhood of an instance in a formal and definite manner, we are able to accurately represent the ML model in that neighborhood. Essentially, this addresses a key task of local XAI methods to make the ML model around an instance transparent: First and foremost, generating surrogate models is not necessary. Frequently used surrogate models such as linear regression or decision trees often have difficulties to capture the actual DBs of a ML model even locally (cf. Figure 1). This is particularly evident in classification tasks with a higher number of classes such as the COMPAS dataset (8 classes), where (the surrogate model-based methods) ROLEX and LIME achieve significantly lower fidelity than on the Fetal Health dataset (3 classes). Thereby, local XAI methods often explain an instance focusing on one (usually the nearest) DB, potentially ignoring other relevant DBs in the neighborhood of the instance. In the case of the instance from the Fetal Health dataset illustrated in Figure 1, this can have severe consequences as several DBs to Class "SUSPECT" and even Class "PATHOLOGICAL" are in fact completely disregarded in the provided explanation. In contrast, our method is able to reconstruct all DBs and CSs in a neighborhood of arbitrary size. In particular, this profound information on the (local) decisions of the ML model provides a valuable basis for its adaptable and accurate visualization on the feature space as shown in the Figures 2 and 3. This indicates that our method has the potential to enhance the transparency of the ML model and the decisions it provides.

6.2 Measuring the Sensitivity of a ML Model Decision

For the second contribution, we leverage the formal and definite representation of the DBs to analyze the sensitivity of ML model predictions. To this end, we introduce two theoretically founded measures for sensitivity, prove their validity using Definition 3 and 4 and thus show Proposition 2. The first measure (CPV in Definition 7) captures the sensitivity of an instance by modeling its potential changes (regarding all or a selected set of features) with a probability distribution, and then precisely assigning the probability masses to their respective classes. This assignment harnesses the formal representation of the CSs and is feasible for arbitrary probability distributions describing feature changes. The resulting class probabilities pose a meaningful indicator for the sensitivity of the instance, because, e.g., an instance with similar or evenly distributed probabilities for different classes is very sensitive in its ML decision and should thus be handled with caution. As a second measure for the sensitivity of an instance, we introduce its Distance and Direction to DBs (Definition 8). Again, by harnessing the formal representation of the DBs, it is possible to accurately determine the distance between an instance and its nearest point lying on a DB. This provides a meaningful and interpretable indication of the sensitivity of the instance because it not only sheds light upon the minimal amount of feature value change required for a different class assignment, but also fully discloses the direction (i.e., the combination of features) in which that change needs to occur. Conversely, this distance explicitly defines the radius of a sphere around the instance in which the class prediction does not change and thus, is stable. For example, the distance between Instance 0 and its nearest point on a DB is rather high, indicating that the class decision of the ML model is obviously less sensitive to changes than Instance 78 (cf. Table 3 and Figure 3). Moreover, we also apply this measure feature-wise allowing us to isolate and analyze the sensitivity of each individual feature. For example, the influence of the feature *Baseline Heart Rate* on the class decision for Instance 78 (x-axis on the right side of Figure 3) is made transparent by the feature-wise distances of 0.4 (in negative direction) and 9.9 (in positive direction) according to Table 3. This means that even a minimal decrease of the *Baseline Heart Rate* would change the class prediction of the ML model, whereas for the same instance, a substantially higher increase of the *Baseline Heart Rate* would be required. In this manner, all features and their influence on the class decision can be analyzed.

We now briefly outline a possible workflow that users can follow to utilize the advantages of EXPLORE. First, we suggest examining the direction vector from the considered data instance to the nearest decision boundary (overall or for each class). The entries of the direction vector provide a valid feature attribution since they (by definition) indicate the necessary changes for each feature to reach the boundary and thus, the subspace of any focused class. Second, one can analyze multiple or – if needed – potentially all nearby decision boundaries in the same manner, enabling a comprehensive understanding of the local neighborhood. For example, in the case of a physician treating a patient, this reveals possibly very different directions to nearby “healthy” class subspaces each corresponding to a possible therapy strategy. However, only relevant features (in the given context) should be considered in this analysis, as there are features that are impossible to change (e.g., the age of a patient), or only an increase, decrease, or other specific

range of values is attainable. In such cases, we recommend adapting the expansion of the neighborhood accordingly, so that only attainable decision boundaries are explored. On this basis, also the most important features for a visualization of the neighborhood can be indicated. Finally, this feature analysis also sheds light on possible feature interactions, as the attribution of each feature to a class change often varies depending on whether the other features increase or decrease, and by the magnitude of their changes. This way, the relevant feature interactions for a data instance become apparent for a user.

Furthermore, the computational complexity of EXPLORE is approximately proportional to the number of linear regions that lie in the considered neighborhood of an instance. Therefore, it makes sense to focus on the direct neighborhood containing the nearby and most relevant decision boundaries (and thus, linear regions) only. This not only reduces the runtime, but also facilitates the interpretability of our method for users. Consistent with the workflow outlined above, we also suggest focusing on a reasonable (pre-)selection of relevant features to reduce the dimension of the feature space and thus the computational cost.

Our work also has limitations that provide a starting point for future research. To begin with, we instantiated our meta method only for the challenging ML model type of NNs in this paper. However, our method can also be applied to other model types such as decision trees, random forests, support vector machines, or other more complex NN architectures such as, e.g., CNNs with ReLU or NNs for text mining (Binder et al., 2022). However, it is part of future research to specify corresponding method instantiations. Further, we focused on the reconstruction of DBs and CSs in the neighborhood of an instance. Our method can also be extended to the whole feature space, resulting in a representation of all DBs and CSs learned by the model and thus, a global explanation. Moreover, our method can contribute to the research of counterfactual explanations by enabling the deterministic computation of counterfactuals based on the overall and feature-wise distances of an instance to the DBs. Thereby, counterfactuals can also be exactly determined with respect to other properties discussed in literature such as proximity to ground truth instances. Future research could also conduct user-centric studies to examine understandability of our visualizations and the proposed sensitivity measures. For example, these could include 2-/3-dimensional visualizations of the neighborhood of an instance (including its DBs and CSs). Such studies could further develop the alignment of our method with user needs and preferences, facilitating effective interaction and collaboration between users and AI.

7 References

Alvarez-Melis, D., & Jaakkola, T. (2018). On the robustness of interpretability methods. *Proceedings of the 2018 ICML Workshop on Human Interpretability in Machine Learning*, 66–71.

Ayres-de-Campos, D., Bernardes, J., Garrido, A., Marques-de-sá, J., & Pereira-Ieite, L. (2000). Sisporto 2.0: A program for automated analysis of cardiotocograms. *Journal of Maternal-Fetal Medicine*, *9*(5), 311–318. https://doi.org/10.3109/14767050009053454

Bauer, K., & Gill, A. (2024). Mirror, mirror on the wall: Algorithmic assessments, transparency, and self-fulfilling prophecies. *Information Systems Research*, *35*(1), 226–248. https://doi.org/10.1287/isre.2023.1217

Binder, M., Heinrich, B., Hopf, M., & Schiller, A. (2022). Global reconstruction of language models with linguistic rules – Explainable AI for online consumer reviews. *Electronic Markets*, *32*(4), 2123–2138. https://doi.org/10.1007/s12525-022-00612-5

Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer.

Brasse, J., Broder, H. R., Förster, M., Klier, M., & Sigler, I. (2023). Explainable artificial intelligence in information systems: A review of the status quo and future research directions. *Electronic Markets*, *33*(1), 1–30. https://doi.org/10.1007/s12525-023-00644-5

Dombrowski, A.-K., Alber, M., Anders, C., Ackermann, M., Müller, K.-R., & Kessel, P. (2019). Explanations can be manipulated and geometry is to blame. *Advances in Neural Information Processing Systems*, 13589–13600.

EU AI Act. (2024). *The EU Artificial Intelligence Act*. https://artificialintelligenceact.eu/

Goodfellow, I., Courville, A., & Bengio, Y. (2016). *Deep learning*. The MIT Press.

Hendrycks, D., & Gimpel, K. (2017). A baseline for detecting misclassified and out-of-distribution examples in neural networks. *Proceedings of the 5th International Conference on Learning Representations*.

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, *2*(5), 359–366. https://doi.org/10.1016/0893-6080(89)90020-8

Hu, X., Liu, W., Bian, J., & Pei, J. (2020). Measuring model complexity of neural networks with curve activation functions. *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1521–1531. https://doi.org/10.1145/3394486.3403203

Jia, Y., Bailey, J., Ramamohanarao, K., Leckie, C., & Houle, M. E. (2019). Improving the quality of explanations with local embedding perturbations. *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 875–884. https://doi.org/10.1145/3292500.3330930

Kim, B., Srinivasan, K., Kong, S. H., Kim, J. H., Shin, C. S., & Ram, S. (2023). ROLEX: A novel method for interpretable machine learning using robust local explanations. *MIS Quarterly*, *47*(3), 1303–1332. https://doi.org/10.25300/MISQ/2022/17141

Krapf, T., Hagn, M., Miethaner, P., Schiller, A., Luttner, L., & Heinrich, B. (2024). Piecewise linear transformation – Propagating aleatoric uncertainty in neural networks. *Proceedings of the 38th AAAI Conference on Artificial Intelligence*, 20456–20464. https://doi.org/10.1609/aaai.v38i18.30029

Laugel, T., Renard, X., Lesot, M.-J., Marsala, C., & Detyniecki, M. (2018). Defining locality for surrogates in post-hoc interpretablity. *Proceedings of the 2018 ICML Workshop on Human Interpretability in Machine Learning*, 47–53.

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why should I trust you?" Explaining the predictions of any classifier. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1135–1144. https://doi.org/10.1145/2939672.2939778

Sattelberg, B., Cavalieri, R., Kirby, M., Peterson, C., & Beveridge, R. (2023). Locally linear attributes of ReLU neural networks. *Frontiers in Artificial Intelligence*, *6*, 1–17. https://doi.org/10.3389/frai.2023.1255192

van Stein, B., Raponi, E., Sadeghi, Z., Bouman, N., van Ham, R. C., & Bäck, T. (2022). A comparison of global sensitivity analysis methods for explainable AI with an application in genomic prediction. *IEEE Access*, *10*, 103364–103381. https://doi.org/10.1109/ACCESS.2022.3210175

Yeh, C.-K., Hsieh, C.-Y., Suggala, A., Inouye, D. I., & Ravikumar, P. K. (2019). On the (in)fidelity and sensitivity of explanations. *Advances in Neural Information Processing Systems*, 10967–10978.

Zhang, L., Naitzat, G., & Lim, L.-H. (2018). Tropical geometry of deep neural networks. *Proceedings of the 35th International Conference on Machine Learning*, 5824–5832.

3.3 Paper 7: Overcoming the Black Box: Explaining Machine Learning Models with EXPLORE

Current Status	Full Citation
This paper is **under review** (since 11/2025) for publication in the journal *MIS Quarterly*	Heinrich, B., Krapf, T., & Miethaner, P. (2025). Overcoming the Black Box: Explaining Machine Learning Models with EXPLORE. *Working Paper*, University of Regensburg

Abstract:

Artificial Intelligence (AI) and especially Machine Learning (ML) models are pervasive in research, business and society. However, due to their black-box nature, it is often not transparent how the predictions of various ML models are achieved. As a result, numerous Explainable AI (XAI) methods have been proposed in the literature to provide explanations for ML model predictions. Importantly, existing XAI methods which often rely on perturbations, random sampling, and surrogate models, still exhibit critical weaknesses in terms of fidelity, robustness, and faithfulness of their explanations, negatively impacting both practical benefits and user trust. Therefore, we propose the novel XAI method EXPLORE, which aims to enhance the transparency of ML model predictions by representing the actual decision boundaries and class subspaces of ML models in a functional and definite manner—without relying on perturbations, random sampling, and surrogate models. Further, based on this method, we introduce new ways to determine feature attributions and counterfactual examples that enable local explanations for individual data instances. We theoretically and empirically evaluate the fidelity and robustness of our method, as well as the faithfulness of the provided feature attributions and counterfactual examples—using different real-world datasets from high-stakes domains—compared to existing XAI methods.

Keywords: Explainable artificial intelligence, XAI, local explanation, piecewise linear functions, feature attribution, counterfactual explanation

Please note that the overall formatting of the paper has been adjusted to the remainder of the dissertation.

1 Introduction

Artificial intelligence (AI), and in particular machine learning (ML) models are becoming ubiquitous and indispensable in research, business and society, and their importance and impact on everyday life continues to grow. Although the transparency of ML model predictions can lead to side effects (cf., e.g., Bauer & Gill, 2024) it is essential to explain their predictions for a real, responsible use (Berente et al., 2021; Brasse et al., 2023). Indeed, there have been several incidents where the opacity of AI has led to severe consequences for the people involved. One current example is the case of an AI system used by the UK government to detect welfare fraud that showed bias based on people's age, disability, marital status, and nationality, fueling calls for greater transparency (Booth, 2024). The importance of this topic has also been recognized and intensively discussed, as becomes evident by the now legally binding AI Act passed by the EU Parliament (EU AI Act, 2025) imposing requirements towards the transparency, fidelity, and robustness of ML model predictions. This is particularly relevant in finance, healthcare, legal affairs/justice, traffic, or other high-stakes domains. To address these critical requirements, the research field of Explainable AI (XAI) aims to mitigate the risks by improving the explainability of black-box ML models (Brasse et al., 2023; Ribeiro et al., 2016). Among other contributions of XAI, various methods have been proposed to make the predictions of ML models more transparent, since this plays a vital role in ML-augmented decision making, especially in critical domains such as medical diagnostics and healthcare (Guo et al., 2025; Kim et al., 2023).

In this paper, we aim to contribute to the body of knowledge dealing with the transparency and explanation of individual ML model predictions, and thus local explainability. Literature offers numerous local XAI methods, for which LIME (Ribeiro et al., 2016), SHAP (Lundberg & Lee, 2017), and ROLEX (Kim et al., 2023) are recognized examples. Like many other local XAI methods, they use perturbations and random sampling, often to generate surrogate models, aiming to analyze the local neighborhood of an instance. Despite the strengths of these methods and the outcomes they provide to explain black-box models, they still suffer from methodological issues associated with the generation of their explanation. For instance, although being aware that surrogate models are explicitly intended to simplify the local prediction behavior of an ML model, their generation using perturbations, random sampling, and other stochastic elements leads to significant shortcomings. For example, capturing the actual decision boundaries (DBs) of an ML model, which is central for determining both feature attributions and counterfactual (CF) examples, even locally often poses a substantial problem, especially when multiple DBs exist near the considered instance (cf. details in Section 3). On the one hand, this results in existing XAI methods exhibiting limited fidelity in the sense that their explanations lead to incorrect class predictions for instances compared to the ML model to be explained. On the other hand, their robustness is also compromised, which becomes highly problematic when their outcomes for the same or very similar instances are not reproducible, thereby undermining user trust in the XAI methods themselves. Moreover, applying perturbations, random sampling, and surrogates, XAI methods may encounter severe difficulties in accurately determining the relevant features and their attributions (Fernández-Loría et al., 2022; Salih et al., 2025). Given the growing importance of XAI

in fostering trust in high-stakes, ML-augmented decision making, it is crucial that XAI methods provide faithful explanations. Striving to contribute to these challenges, we introduce EXPLORE (EXPlaining machine learning models by LOcal REconstruction), a local XAI method that does not rely on perturbations, random sampling, and surrogate models, but explicitly focuses on the representation of the actual DBs and class subspaces (CSs) of an ML model in the neighborhood of the considered instance. The research questions addressed are the following:

- RQ1: How can the DBs and CSs of an ML model be locally reconstructed in a functional and definite manner, thus making them fully transparent?
- RQ2: Based on RQ1, how can feature attributions and CF examples be functionally determined to provide local explanations for individual ML model predictions?

Our contributions are twofold: We (1) propose EXPLORE, a novel post hoc local XAI method that uses piecewise linear (PL) functions to reconstruct the DBs and CSs of ML classification models in a functional and definite manner. We (2) present functionally determined feature attributions and CF examples that enable local explanations for individual instances. In this paper, the term 'functional' is used in the sense that DBs and CSs can be represented based on mathematical (in)equalities which are made explicit and can be directly computed. By 'definite', we denote that this representation is unique, meaning it is the one correct and complete characterization of the DBs and CSs for the given ML model. Based on (1) and (2), our method outperforms existing XAI methods regarding fidelity, robustness, and faithfulness. We provide formal proofs for the fidelity, robustness, and faithfulness of EXPLORE and substantiate them with empirical evidence on five real-world datasets from the domains of healthcare, loan approval, traffic safety, and criminal recidivism. Our contributions come with major implications: First, by functionally and definitely reconstructing the actual DBs and CSs of an ML model, accurate and robust outcomes and thus full transparency are *guaranteed* by EXPLORE. This transparency is of utmost importance, e.g., for the ethical and fair use of ML, as it is crucial for ensuring legal and social justice and preventing discrimination in domains such as the public sector (Kronblad et al., 2024) or finance (Hurlin et al., 2024; Li et al., 2024). Second, by providing faithful feature attributions and CFs, EXPLORE enables the accurate determination of necessary and sufficient feature changes to either remain in the present class or change to a particular class. In high-stakes domains such as healthcare, such faithful local explanations enable the identification of the truly relevant features and their accurate attribution for an ML-supported medical diagnosis, paving the way for suitable treatment and enhancing patient trust in ML models (Kim et al., 2023).

The remainder of the paper is structured as follows. First, we introduce a running example that we use for illustrations throughout this paper. We then position our work in literature, especially related to existing local XAI methods and discuss their strengths and weaknesses. We proceed by presenting the theoretical foundations and basic ideas for our method and its provided local explanations. We then introduce the meta method of EXPLORE and instantiate it for feedforward neural networks (NNs). Based on this, we present our local explanations based on EXPLORE. Thereafter, we evaluate our new method and discuss our results and main findings. We conclude by reflecting on limitations and providing an outlook on future research.

2 Running Example

For our illustrations, we briefly present a running example employing the Fetal Health dataset[1]. A low neonatal mortality is a central target in the United Nations' Sustainable Development Goals for 2030 and a key indicator of human progress over time (Unicef, 2024). Many cases of neonatal, infant, and maternal mortality are preventable, and cardiotocography is an accessible and effective option to diagnose fetal health. ML model predictions can support physicians in this context (Mehbodniya et al., 2022), but given this high-stakes domain, an explanation of each prediction should be transparent and accurate, therefore the fidelity, robustness, and faithfulness of explanations are of particular importance. The Fetal Health dataset consists of 2,126 real data instances, each containing measurements on a fetus and mother from cardiotocographic examinations. The features comprise information on the fetal heart rate, abnormalities of the fetal heart rate, and general information on the fetus and the mother (cf. Table 1). Each instance was assigned by three expert obstetricians to one of the three classes NORMAL (the fetus is healthy), SUSPECT (the fetus may not be healthy and further medical checks are required) and PATHOLOGICAL (the fetus' health is critical and immediate medical intervention is required).

Feature Category	Brief Description	Feature Examples
Fetal heart rate	Data on the fetal heart rate	Baseline fetal heart rate Heart rate histogram minimum/maximum/width Heart rate histogram tendency
Heart rate abnormalities	Data on the occurrence of unusual fetal heart rate abnormalities	Prolonged decelerations Mean value of long-term variability Abnormal short-term variability
General information	Data on the fetus and mother (not related to heart rate)	Fetal movement Uterine contractions

Table 1. Fetal Health Dataset Feature Information

On this dataset, we trained different ML models such as NNs, support vector machines (SVM), as well as random forest (RF) and AdaBoost classifiers (cf. Section 6). Following common practice, we standardized each feature to the unit interval [0,1] (min-max normalization) for model training to avoid disproportionate impact of features due to their original unit scale. Due to its widespread use, black-box nature and complexity, we focus on the NN in our following illustrations regarding the running example (i.e., the other ML models are presented in Section 6).

3 Background and Related Work

In training, ML models generally aim to extract and generalize patterns from a given sample of data (i.e., the training data). This should allow them (at best) to make accurate predictions for

[1] We provide information on the provenance of all datasets used in this paper in our Supplementary Material.

new, unseen data instances from the same feature space based on these patterns. In the case of a classification task, an ML model $f: X \rightarrow Y$ is learned that assigns an output value $y = f(x)$ to any instance $x \in X$. Usually, the output value $y \in Y$ is either given by a class itself or by a value directly corresponding to a class, e.g., the Softmax vector of an NN prediction (Goodfellow et al., 2016). In this sense, the ML model divides the feature space into several CSs, each consisting of predictions that are assigned to the same class. The boundaries between the CSs are called DBs and can be (locally) linear or non-linear depending on the ML model used (Bishop, 2006; Kim et al., 2023). Contrary to symbolic ML models such as decision trees (DT) or variants of regressions, many established ML models such as NNs or SVMs are subsymbolic, i.e., their DBs and CSs are not transparent from the learned model parameters. This makes them black boxes, necessitating XAI methods, as decisions cannot be directly recognized (Bishop, 2006).

3.1 Local Explanations and Performance Criteria

When reviewing local XAI methods in more depth, it is essential to understand how their performance has to be evaluated from a methodological perspective. Current literature discusses various performance criteria and their relevance (e.g., Agarwal et al., 2022; Nauta et al., 2023; Pawlicki et al., 2024). One central and very well-known criterion for performance is fidelity (cf., e.g., Kim et al., 2023; Laugel et al., 2018b, also called accuracy or correctness). It expresses the extent to which the predictions of an XAI method locally match the predictions of the ML model being explained. For instance, criteria such as the local fidelity score or the local DB-aware fidelity score (LDA, cf. Kim et al., 2023) assess, for a number of instances, whether the classes assigned by the XAI method match the classes assigned by the ML model. If this match holds true for all instances, the local fidelity would be perfect (i.e., 100% or 1). However, even in a local environment this fidelity can fall well below 100%. This means that if, e.g., the ML model predicts Class SUSPECT for a fetus, while the XAI method predicts Class NORMAL, it would try to explain the false class, which is highly problematic. Another important performance criterion is robustness (cf., e.g., Alvarez-Melis & Jaakkola, 2018a; Saarela & Geogieva, 2022 also referred to as stability, sensitivity, continuity, or reproducibility), which aims to assess how stable and reproducible the predictions of an XAI method are for very similar instances or even the same instance. As many XAI methods rely on perturbations and random sampling when generating their predictions, one has to assess the degree to which these predictions are robust (e.g., Dombrowski et al., 2019; Lakkaraju et al., 2020). Crucially, different predictions for the same instance, e.g., the same fetus is diagnosed several times, or very similar instances present a significant issue for the robustness of the XAI method and can greatly undermine user trust.

A third central criterion is the faithfulness of explanations provided by a local XAI method (e.g., Alvarez-Melis & Jaakkola, 2018b; Ribeiro et al., 2016, also referred to as correctness or relevance of feature attributions). In this context, a general distinction can be made between abductive and contrastive (CF) explanations, which are often referred to as the central "why or why not" questions in XAI (Herm et al., 2023; Miller, 2019). Abductive explanations focus on the question of why an instance was classified by the ML model in the way it was—i.e., which

features and feature values of the considered instance are relevant and important for the current classification. Contrastive explanations, on the other hand, focus on the question of why an instance was not assigned to a different class, or what needs to change in its features and their values to be assigned to a different class. For a faithful abductive and contrastive explanation of an XAI method, it is necessary to (a) identify the relevant features and (b) determine the attribution of the individual features (e.g., the ranking of importance) under the condition that the instance retains or alters its current class assignment. If in our running example only the features *Baseline Heart Rate* and *Abnormal Short-Term Variability* are actually relevant to change the prediction from PATHOLOGICAL to SUSPECT for a fetus, a CF explanation including (only) the features *Baseline Heart Rate* and *Uterine Contractions* would be (at least partly) not faithful, since the correct set of relevant features is not identified.

3.2 Existing XAI Methods

To examine local XAI methods and review their methodological strengths and weaknesses, we structure them based on their explanation type and based on their underlying methodological (algorithmic) strategy (cf. Table 2). In accordance with literature, we consider *CF explanations*, *feature attributions*, and *surrogate models* as main categories of explanation types of existing XAI methods (cf., e.g., Fernández-Loría et al., 2022). With EXPLORE, we aim to contribute to each category of explanation types by leveraging the information about DBs and CSs provided by its functional local reconstruction of the ML model. Regarding their methodological strategy to generate explanations, we organize XAI methods into four categories based on *perturbations*, *sampling*, *gradients*, and *ML model reconstruction*. This categorization of existing strategies follows the one proposed in the XAI survey by Mersha et al. (2024), however, we decide to distinguish more precisely between perturbation-based and sampling-based methods. Both strategies crucially utilize artificially generated data points that typically lie in close proximity to the instance being explained. We consider a method to be perturbation-based if the data points are generated by changing the (original) data instances in a systematic manner such as, e.g., changing only one single or specific subsets of feature values by a certain amount to examine whether or not the class prediction changes (e.g., SHAP); and sampling-based if the generation of the data points largely relies on random drawing from a certain probability distribution, e.g., drawing a number of samples around the instance as training data for a surrogate model.

Counterfactual Explanation. The main goal of methods providing CF explanations is to explain model predictions by showing examples of how changing certain feature values of the instance (e.g., minimally, or with respect to other requirements) alters the prediction outcome of the model, thus providing a *contrastive* explanation. On this basis, examining the direction and distance from the instance to the CF example(s) reveals for which features a positive or negative change is necessary, optional, or not needed at all. Methods for CF explanations utilize different strategies: For instance, the methods PrototypeCF (van Looveren & Klaise, 2021), FACE (Poyiadzi et al., 2020), and NICE (Brughmans et al., 2024) rely on targeted perturbations, the Growing Sphere Counterfactual (GSCF) method by Laugel et al. (2018a) is based on random sampling,

and DICE (Mothilal et al., 2020), WACHTERCF (Wachter et al., 2018), Projected Gradient Descent (PGD, Madry et al., 2018), and the Fast Gradient Method (FGM, Goodfellow et al., 2015) exploit the model gradient to determine CFs.

Such methods can achieve a higher success rate, i.e., the percentage of cases where a valid CF example is successfully identified, and can incorporate certain requirements, e.g., proximity to the instance. However, due to their reliance on heuristics based on perturbations, random sampling, and gradients, they are partially unable to identify CF examples even though they provably exist. Moreover, even when CF examples are determined, they are typically not optimal with respect to the requirements and metrics which they are determined on, such as proximity to the instance (cf. Evaluation). EXPLORE does not rely on such heuristics but instead functionally reconstructs the model and its DBs and CSs, thus being able to identify valid CF examples.

<table>
<tr><th>Explanation Type
Method based on</th><th>Counterfactual Explanation</th><th>Feature Attribution</th><th>Surrogate Model (e.g., linear models, rule sets)</th></tr>
<tr><td>Perturbations</td><td>PrototypeCF,
FACE, NICE</td><td>SHAP</td><td>MAPLE,
ANCHORS, LORE</td></tr>
<tr><td rowspan="2">Sampling</td><td rowspan="2">Growing Sphere Counterfactuals (GSCF)</td><td>SPVIM, RISE</td><td>ROLEX, LS</td></tr>
<tr><td colspan="2">LIME, S-LIME, LEAP</td></tr>
<tr><td>Gradients</td><td>DICE, WachterCF
PGD, FGM</td><td>Integrated Gradients</td><td>Explanation Vectors</td></tr>
<tr><td>ML Model Reconstruction</td><td colspan="3">EXPLORE
(exact/approximate)</td></tr>
</table>

Table 2. Outlining Selected XAI Methods

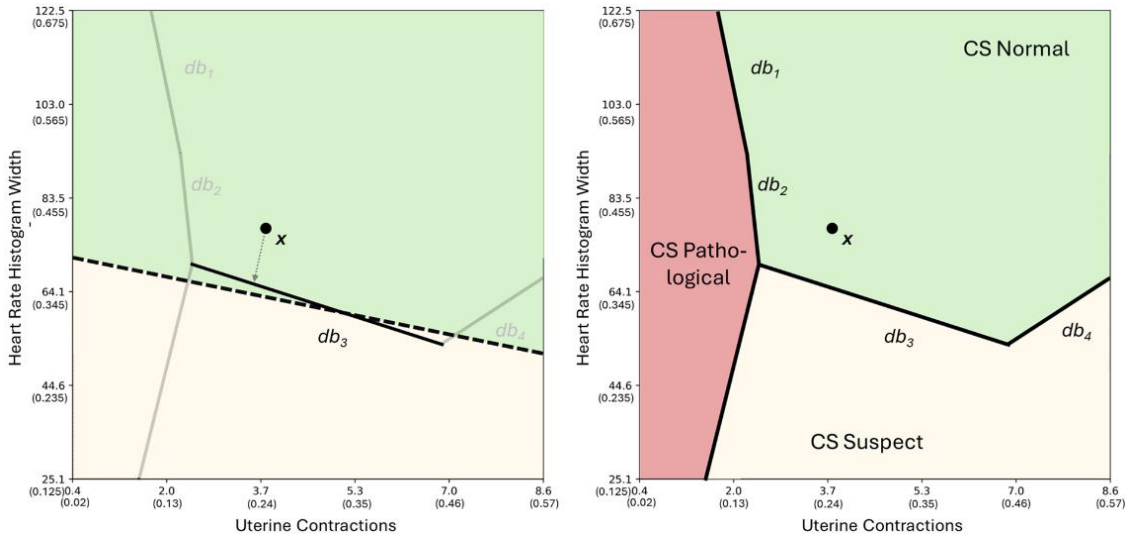
Feature Attribution/Importance. The goal of feature attribution methods is to explain one or many predictions of a complex ML model by associating a weight/value to each feature, indicating each feature's (relative or absolute) relevance and attribution to the ML model's class prediction. Methods such as SPVIM (Williamson & Feng, 2020) and RISE (Petsiuk et al., 2018) belong to this category and are mainly based on random sampling. Another very well-known example is SHAP (Lundberg & Lee, 2017) which utilizes perturbations to indicate the feature attributions for the current class assignment, thus providing an *abductive* explanation. In their Integrated Gradients method, Sundararajan et al. (2017) provide insights into the predictive behavior of NNs by deriving feature attributions based on the network's gradients. Nevertheless, it has been discussed in literature that feature attributions can often be harder to interpret (as opposed to, e.g., the feature values of CF examples) due to aggregation and averaging of multiple feature orderings and importance weights obtained by perturbations, random samples, or gradients (Fernández-Loría et al., 2022). This issue is further exacerbated by the fact that these heuristics can only incorporate a partial view of the local model behavior which critically results in a limited faithfulness of the feature attributions (cf. Section Evaluation). Against this, we aim to leverage the functional representation of DBs and CSs provided by EXPLORE to present robust feature attributions by analyzing the importance of each feature for a given model prediction.

Surrogate Model. To make a model prediction transparent, many XAI methods propose to approximate the black-box ML model locally, i.e., in the neighborhood of the considered instance, with a simpler, symbolic surrogate model (e.g., a linear regression). In some cases, such a surrogate model can also be directly associated with a feature attribution vector, thus providing a multifaceted explanation. For example, in LIME, S-LIME, and LEAP, the coefficients of the linear regression surrogate model are explicitly interpreted as feature attributions. In this sense, these methods can be evaluated with respect to both, fidelity and robustness (as surrogate model) and faithfulness (as feature attribution). However, the categories of these explanation types must still be distinguished, as not all XAI methods that provide surrogate models also provide feature attributions and vice versa. For instance, ANCHORS (Ribeiro et al., 2018) extract decision rules to mimic and thereby explain the predictions of the ML model using a set of rules (given by if-then-statements), which can be represented as a DT-based surrogate model; but an associated feature attribution vector cannot be derived directly. On the other hand, the feature attributions generated by, e.g. SHAP, do not result from an explicitly specified surrogate model.

The most well-known representative of such local XAI methods is LIME (Local Interpretable Model-agnostic Explanations; Ribeiro et al., 2016) with a multitude of further variants building upon it, such as LS (Local Surrogates; Laugel et al., 2018b), and S-LIME (Zhou et al., 2021). Thereby, the surrogate model is typically achieved by first generating perturbations or artificial sample instances close to the instance being explained and obtaining the prediction of the black-box ML model for each sample. Based on these artificial instances and their predictions as training data, the surrogate model is trained to approximate the ML model including its DBs, thus trying to locally explain the ML model. A further example for such an XAI method is MAPLE (Model Agnostic Supervised Local Explanations; Plumb et al., 2018), which utilizes an RF and gradient boosting to construct the surrogate model. In particular, Kim et al. (2023) has to be noted, as they clearly outperform previous XAI methods in terms of fidelity with their method ROLEX (RObust Local EXplanations). Finally, surrogate models are rather rarely created based on the model's gradients, as in the case of Explanation Vectors by Baehrens et al. (2010).

Despite the strengths of existing XAI approaches, such as (partly) providing good local explanations, there are also methodological weaknesses that critically compromise their fidelity and robustness. A key weakness is that actual DBs are not or not correctly recognized during the generation of explanations, even at a local level. This is partly attributed to the well-known limitations of sparse perturbations and random sampling in higher-dimensional spaces, and the critical assumption of feature independence. Both becomes particularly pronounced in case of multiple DBs, which are especially common in higher-dimensional feature spaces or multi-class problems—both frequently encountered in real-world applications. To demonstrate such issues being part of the addressed research gap, we analyzed the test instances from the Fetal Health dataset regarding surrogate model-based methods such as LIME and MAPLE. Our results indicate that in more than 41% (LIME) and 48% (MAPLE) of all instances classified as PATHOLOGICAL or SUSPECT, the actual related DBs are not or not correctly recognized. This results in substantially compromised fidelities of less than 75% (see details in Section 6) potentially leading to both

seriously incorrect predictions and explanations of the health of the respective fetus instances. Related to linear surrogate models (as employed by, e.g., LIME, MAPLE, or LS), an example for such an instance is visualized in Figure 1 with respect to the two features *Uterine Contractions* and *Heart Rate Histogram Width*. The left side of Figure 1 displays a local explanation for the Instance x as typically provided by a LIME surrogate model (e.g., a Ridge Regression, illustrated by the dotted black line in left side of Figure 1). Crucially, such surrogate models can at best approximate only one actual DB around the fetus instance x due to their linearity (e.g., DB db_3 in left side of Figure 1), thus all other actual related DBs are not made transparent.



Linear Surrogate Model Actual DBs of the ML model

Figure 1. DBs Close to a Fetus Instance (Fetal Health Example). Axes are labeled in real-world units with their respective standardized unit values in parentheses.

This issue becomes evident in the right side of Figure 1 showing the actual DBs of the ML model. Thereby, several DBs (db_1-db_4 in Figure 1) actually lie in the neighborhood of x yet all but db_3 are completely neglected. Even worse, the DBs db_1 and db_2 and the CS PATHOLOGICAL are completely disregarded. Note that usually due to the black-box nature of subsymbolic ML models, it is not even known that the actual DBs db_1, db_2, db_4 exist at all (which is here made transparent by our method EXPLORE). In particular, XAI methods using surrogate models and feature attributions would not indicate that the ML prediction for the health status of fetus x changes to PATHOLOGICAL with only a small decrease of the feature *Uterine Contractions*.

To conclude, while existing local XAI methods clearly contribute to explaining ML predictions, they also exhibit methodological weaknesses due to their reliance on perturbations, random sampling, and surrogate models, which can result in limited fidelity, robustness, and faithfulness of explanations. Our contribution to existing research is twofold: (1) Regarding surrogate models, we aim to enhance the transparency of ML model predictions through the functional representation of the actual DBs and CSs of ML models. This not only leads to improved fidelity but also ensures the robustness of EXPLORE due to its determinism. (2) EXPLORE is also able to provide faithful feature attributions, and for CFs, EXPLORE can rigorously determine whether or not a CF example exists in a given neighborhood of an instance. Even further, the reconstruction of the

model can then be utilized for a targeted selection of CF examples with respect to needed requirements (e.g., risk-tolerance; cf. Section Discussion).

4 Theoretical Foundations and Basic Ideas

The paper has two objectives: (1) Establishing the functional and definite representations of the actual DBs and CSs of an ML model and (2) providing local explanations for model predictions based on determined feature attributions and CF examples. To address Objective (1), we propose to reconstruct the actual DBs and CSs of a (black-box) ML model in a functional and definite manner. This leads us to the first proposition of our approach.

PROPOSITION 1: Actual DBs and CSs of the ML model to be explained.

1c. The DBs of an ML model can be reconstructed in a functional and definite manner using piecewise linear (PL) functions. For models inducing PL DBs (e.g., ReLU NN, linear SVM, RF, boosting techniques such as AdaBoost), this corresponds to an exact reconstruction of the DBs, while for ML models with non-PL DBs (e.g., NN with sigmoid activation functions), this corresponds to an (arbitrarily exact) approximate reconstruction. Thus, regarding EXPLORE, we distinguish between exact and approximate reconstruction in Table 2.

1d. By composing the reconstructed DBs of an ML model represented by PL functions, the CSs of the ML model can also be determined in a functional manner.

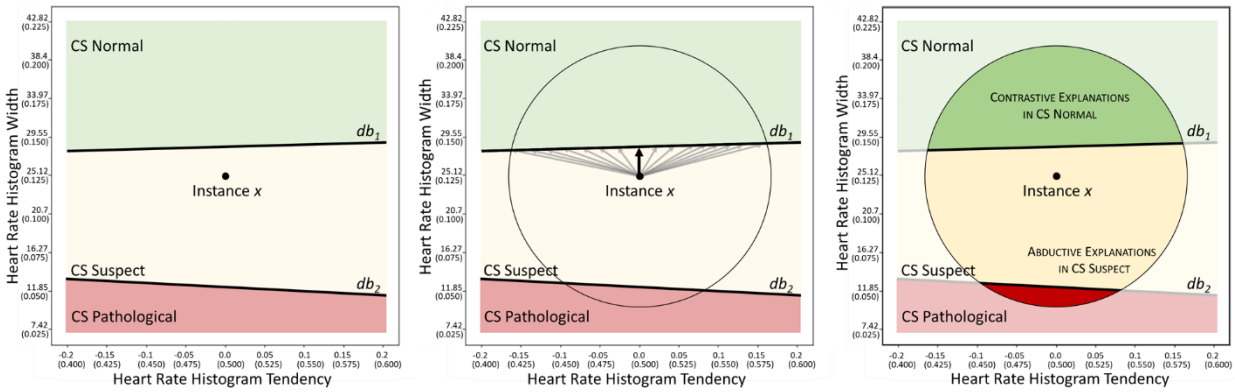


Figure 2. Exemplified DBs, CSs and Feature Attributions in a Neighborhood (Fetal Health Example)

If Proposition 1 holds, the position of any instance within the feature space can be analyzed deterministically. Generating good or bad surrogate models is no longer necessary; instead, the DBs and CSs of the ML model are made transparent by their functional representation. This offers significant advantages as all DBs close to a considered instance are now captured, ensuring that no DBs are ignored or wrong (e.g., non-nearest) DBs are determined. Referring to the Fetal Health dataset, Figure 2 (left) shows an excerpt of the actual DBs and CSs of an NN in the running example for the two selected Features *Heart Rate Histogram Tendency* and *Heart Rate Histogram Width* in the intervals $[-0.2,0.2]$ and $[7.42,42.82]$. For this visualization, no perturbations, random sampling, surrogate models or averaging of features values (e.g., Partial Dependency Plots use such simplifications and thus do not visualize the actual DBs) are necessary. The exemplary Instance x with the feature values $(0, 25.1)$ is not only classified as SUSPECT

(part of the yellow CS SUSPECT) but is also close to the two actual DBs db_1 and db_2 of the NN. Determining the position of Instance x in relation to any actual DB and CS enhances transparency while ensuring fidelity, as both the ML model and EXPLORE assign the same class to any considered instance. Moreover, this identification of DBs and CSs addresses robustness and reproducibility, ensuring stable explanations upon repetition.

Proposition 1 aims to make the nearby DBs and CSs of a considered instance transparent. We call this local proximity the neighborhood of an instance, for which we propose feature attributions and CF examples (Objective (2)) in the following. In Figure 2 (middle), a possible neighborhood is exemplified for Instance x by the sphere around x. We argue that by reconstructing the DBs and CSs in the neighborhood, we can not only reveal why or why not an instance is assigned to a class but also determine the relevant features and their attributions for the current or an alternative class assignment. Considering Instance x in Figure 2 (middle) in its neighborhood, any value changes of the two selected Features *Heart Rate Histogram Tendency* and *Heart Rate Histogram Width* that would shift x onto the DB db_1 (illustrated with gray arrows) would result in a class assignment change from SUSPECT to NORMAL, thus constituting CF / contrastive explanations (similar for DB db_2). On the other hand, any value changes of x in the Features *Histogram Tendency* and *Histogram Width* that would shift x (infinitesimally) close to but not onto a DB would result in retaining the class assignment of SUSPECT, thereby constituting abductive explanations in the neighborhood. Thus, it is emphasized that we propose to represent both abductive and contrastive explanations through possible feature value changes—or, more precisely, through the components of feature vectors. To make our basic idea clear: We always adopt the position of the considered instance, here Instance x, and specifically look at the feature space from its point of view. That is, every feature vector originating from x towards the nearest DB db_1 determines the minimum required feature value change *in this direction* to be classified as NORMAL (CF/contrastive). Conversely, every feature vector originating from x that points (infinitesimally) close to but not onto any DB in the neighborhood determines the maximum feasible feature value change *in this direction* to be still classified as SUSPECT (abductive). In general, feature vectors naturally specify through their components not only which value changes in each feature are both necessary and sufficient to assign the instance to a certain class, e.g., class NORMAL. Rather, they also fully determine the interactions of the features to reach or not to reach a *particular DB*. We call each of these feasible feature vectors a *vector-based feature attribution* VFA_{DB} of x, either to remain in the present class (abductive) or to change to a certain class (CF/contrastive). Since feature vectors represent feature attributions, we assign EXPLORE to feature attributions in Table 2, and—if a feature vector points to another class—to CF examples.

The VFA_{DB} vectors only constitute a part of the abductive and contrastive explanations in a neighborhood. Obviously, there are further possible changes in an instance's feature values (and thus additional feature vectors) that lead to either retaining the current class or changing to a certain other class. More precisely, these are all vectors within the neighborhood that originate from the considered instance and point into the CS of the present class (abductive) or of a certain

other class (CF/contrastive). This is illustrated in Figure 2 (right): Each thought vector originating from x and pointing to any position within one of the colored CSs (in the sphere) represents a feature value change of x to either remain in the current class or change to one of the other two classes. Again, to make our basic idea clear: Instead of DBs (hyperplanes in feature space), we are now focusing on CSs (volumes in feature space). Specifically, we examine feasible feature value changes (feature vectors) that enable reaching any point within the *CS of a certain class*. We call each of these a *vector-based feature attribution* VFA_{CS} of an instance, now explicitly referring to a CS and not a DB. Obviously, each VFA_{CS} constitutes an abductive or CF/contrastive explanation. The following Table 3 summarizes this discussion.

	Abductive Explanation	**Contrastive Explanation**
Decision Boundary	Each VFA_{DB}^{i} shows the *maximum feasible change* to the feature values—in direction of a DB—to remain in the present class i	Each VFA_{DB}^{j} shows the *minimum required change* to the feature values—in direction of a DB—to alter to a certain class j
Class Subspace	Each VFA_{CS}^{i} shows the *feasible change* in the feature values—in direction of the CS—to remain in the present class i	Each VFA_{CS}^{j} shows the *feasible change* in the feature values—in direction of the CS—to alter to a certain class j

Table 3. Conceptualizing Abductive and Contrastive Explanation

To illustrate our idea, we refer to Figure 2 (middle): Each vector to db_1 represents a VFA_{DB}^{Normal} (analogous for DB db_2 below and the class PATHOLOGICAL). The minimum of Feature *Histogram Width* over these feature vectors VFA_{DB}^{Normal} can be calculated as +3.2 (for $VFA_{DB}^{Normal} = \begin{pmatrix} -0.16 \\ 3.2 \end{pmatrix}$), indicating that without changing this feature value by at least +3.2, the Feature *Histogram Tendency* can even be altered arbitrarily within the neighborhood, and the instance cannot change its class to NORMAL. Thus, in this case, the Feature *Histogram Width* is relevant and more important for changing the class to NORMAL (CF/ contrastive explanation) than the Feature *Histogram Tendency*. Moreover, when focusing on the CS NORMAL in the neighborhood (see the green-colored area in Figure 2, right), the CS-oriented feature vectors VFA_{CS}^{Normal} enable an analysis of whether the CS NORMAL substantially extends beyond db_1. This can be of great interest to a user, i.e., whether a further increase in the Feature *Histogram Tendency* beyond db_1 would immediately lead to a class other than NORMAL or not. Examining the maximum of Feature *Histogram Width* over these feature vectors VFA_{CS}^{Normal} reveals that the entire area beyond db_1 belongs to CS NORMAL, with a maximum of +15.0. This shows that the CS NORMAL extends to the edge of the neighborhood, meaning there are many possible feature attributions for changing the class of the fetus to NORMAL.

So far, we focus on a symmetrically extended neighborhood (e.g., a sphere). In reality, the relevant feature space may be constrained by the specific characteristics of the considered features. For instance, the values of a certain feature may not be changeable at all (e.g., the age of a fetus), may only be increased or decreased within certain intervals or in discrete steps, or may only be changeable together with other feature values (e.g., medication administered during treatment affecting values of several features at once). Such settings result in constraints on the

neighborhood, and feature attributions must also be determined for such user-defined cases. Based on this discussion, we formulate Proposition 2.

PROPOSITION 2: Local explanations based on relevant features and their attributions.

2a. For a given neighborhood of an instance, each VFA (one vector of features and their attributions) can be accurately determined regarding a functionally defined DB or CS, thereby providing either an abductive or a contrastive (CF) local explanation.

2b. VFA can be accurately determined for any user-defined neighborhood in the feature space.

Establishing Proposition 2 would help answering two central questions: why an instance was classified by the ML model as it was and why it was not assigned to a different class, and what feature value changes are necessary and sufficient for the instance to be assigned to a different class. To achieve this, we rely on geometric structures such as vectors (directions and distances), neighborhoods, DBs, and CSs. We now present our method for reconstructing the DBs and CSs, as well as our local abductive respective contrastive explanations.

5 Explaining ML Models by Local Reconstruction

5.1 Identifying Decision Boundaries and Class Subspaces

5.1.1 Meta Method of EXPLORE

In this section, we first present the meta method of EXPLORE which we subsequently instantiate for PL feedforward NNs. The latter is used to prove that Proposition 1 holds for feedforward NNs and can similarly be shown for other ML models such as recurrent NN, convolutional NNs (CNNs), SVM, DT, RF, and boosting techniques[2]. To reconstruct the DBs and CSs of an ML model, the following steps are necessary:

Step 1. Formal definition of the ML model: We conceive the ML model to be explained as a mathematical function $f: X \to Y$ that maps any instance $x \in X$ in the feature space onto a target value $f(x) = y \in Y$ in the output space. The feature space X reflects the features and their values. Hence, X might be a discrete or continuous space, or a mixture thereof. Similarly, Y represents the output space of the ML model, which also might be discrete (e.g., a set of classes) or continuous (e.g., a simplex containing the Softmax values of an NN prediction).

Step 2. Identification of linear regions in the feature space: As stated in Proposition 1, we aim to represent DBs by PL functions. Therefore, we need to determine in which regions of the feature space the DBs are PL (i.e., behave like a straight line) and where the break points between these linear DB segments lie. In fact, these segments correspond to the linear regions (denoted

[2] Due to length restrictions, we cannot incorporate corresponding proofs for these ML models in the manuscript. However, on request, we would be glad to provide these proofs in an appendix. In this regard, we already included these ML models in our EXPLORE Python implementation and present the respective evaluation results in Section 6.

by A_u) of the ML model, on which the ML model behaves linearly. Precisely, the function f is given by an affine linear (or even constant) function f_u on each such disjoint linear region $A_u \subset X$ in the feature space. In Figure 3 (left), the actual linear regions $A_1 - A_4$ of the ML model are illustrated by the dotted lines. The subdivision of the feature space X into linear regions A_u together with their respective affine linear functions f_u provides complete information about the ML model to be explained, as the union of the linear regions covers the whole feature space.

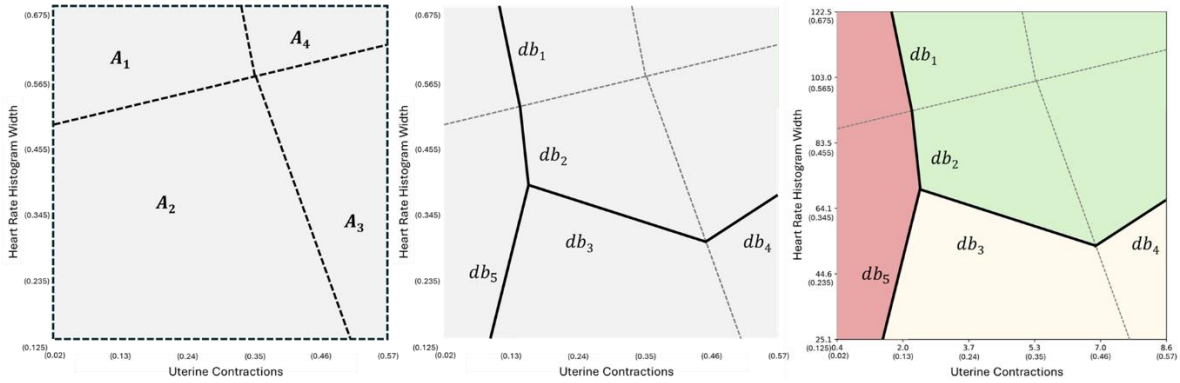


Figure 3. Steps 2-4: Linear Regions (left), DBs (middle), and CSs (right) of an ML Model (Fetal Health)

Step 3. Determining DBs in the feature space: The linear regions can now be used to identify where exactly the ML model changes its (class) decision, allowing the DBs of the ML model to be functionally determined. More precisely, it can be analyzed which points $x \in A_u \subset X$ form the DB up to which the respective functions f_u induce the same class prediction. For our example from Figure 1, the PL segments $db_1 - db_5$ as well as the breakpoints between them are shown in Figure 3 (middle). Indeed, the break points between the linear DB segments db_1 and db_2 as well as db_3 and db_4 lie on the boundary of linear regions on each of which the model is defined by a different affine linear function (Figure 3 middle). Moreover, note that a single linear function can also induce break points (within a linear region) if and only if the break point is a multiclass boundary point of three or more classes at once (cf. break point between db_2, db_3, and db_5).

Step 4. Building CSs in the feature space: Finally, the feature space can be subdivided into CSs, based on the DBs identified in Step 3 (Figure 3 right). Even though each CS corresponds to a single class, it is enclosed by potentially multiple DBs adjoining different classes. Since DBs constitute subsets of hypersurfaces in the feature space and can thus be described by (a system of) (in)equalities, CSs can be functionally represented by (a system of) their respective inequalities.

5.1.2 EXPLORE-Method for Piecewise Linear Feedforward Neural Networks

We now instantiate our meta method for the relevant and complex case of feedforward NNs with PL activation functions (such as ReLU, Leaky/Parametric ReLU, Hard tanh/sigmoid, GELU).

Step 1. Formal definition of the NN: In the following, we consider an arbitrary feedforward NN mapping instances with $m \in \mathbb{N}$ features onto $n \in \mathbb{N}$ classes. Thus, we denote the NN by a function $f: X \to Y$ with feature space $X \subset \mathbb{R}^m$ and output space $Y \subset \mathbb{R}^n$. A (feedforward) NN is

a concatenation of layers (Bishop, 2006; Goodfellow et al., 2016), each consisting of a linear transformation followed by a non-linear activation function (typically ReLU, tanh, or sigmoid).

Step 2. Identification of the linear regions: A main idea is to represent DBs by PL functions. Therefore, we need to find the segments of the feature space on which the DBs are linear and hence, the break points between these PL segments. NNs with PL activation functions can be exactly reconstructed by a PL function on their feature space (Chu et al., 2018; Krapf et al., 2024). For NNs, the linear regions are convex polytopes, which means that they are enclosed by a set of hyperplanes (cf. dotted lines and $A_1 - A_4$ in Figure 3 left). Thus, each linear region can be characterized by a system of inequalities given by a (left side) matrix M and a (right side) vector l. Following this, we introduce Definition 1 allowing us to reconstruct NNs as PL functions.

Definition 1 (NNs as PL Functions). An NN with PL activations has the PL form

$$f(x) = \begin{cases} T_1 x + c_1, & if\ M_1 x \leq l_1, \\ T_2 x + c_2, & if\ M_2 x \leq l_2, \\ & \dots \\ T_t x + c_t, & if\ M_t x \leq l_t \end{cases} \tag{1}$$

with $T_u \in \mathbb{R}^{n \times m}, M_u \in \mathbb{R}^{n_u \times m}, c_u \in \mathbb{R}^n, l_u \in \mathbb{R}^{n_u}$ for $t, n_u \in \mathbb{N}, 1 \leq u \leq t$. The systems of n_u inequalities $M_u x \leq l_u$ in Term (1) divide X into t polytopes $A_u = \{x \in X \mid M_u x \leq l_u\}$, which are disjoint (i.e., they only have their boundaries in common) and satisfy $X = \bigcup_{u=1}^{t} A_u$. On each polytope A_u, the NN represented by f is given by an affine linear function defined by T_u and c_u. Thus, f is a PL function with respect to the linear regions A_u.

Step 3. Determining the DBs: We now leverage the PL form of the NN from Step 2 to determine its DBs. To this end, first, it is necessary to represent the DBs in the *output space* using the (in)equalities in the following definition, to then reconstruct the DBs in the *feature space*.

Definition 2 (Output Space DBs). If a point $y \in Y$ in the output space has two equal entries that are also its largest entries, then y lies on an output space DB. Formally, the DBs between two (different) classes i, j in the output space Y are therefore given by

$$db_{i,j}{}^{o} = \{y \in \mathrm{Y} \mid y_i - y_j = 0 \ \wedge \forall k \neq i, j \colon y_i \geq y_k\} \tag{2}$$

for all $1 \leq i \neq j \leq n$. By this definition, the symmetry $db_{i,j}{}^{o} = db_{j,i}{}^{o}$ holds.

On the output DB between the classes $i = \text{NORMAL}$ and $j = \text{SUSPECT}$, their corresponding output values must be equal, i.e., $y_i - y_j = 0$ must hold. Then, $y_i, y_j \geq y_k$ for $k = \text{PATHOLOGICAL}$ is guaranteed for $y_i, y_j > 1/3$ since the output values of all classes add up to 1, hence a prediction for class k cannot occur. We next propose a system of (in)equalities which represents the DBs in the *feature space*. This is a key result in this section, which we first formulate for one linear region A and two classes i, j in the following lemma. Thereafter, we generalize this result in Definition 3 and characterize all DBs (between all classes) in the whole feature space.

Lemma 1 (Feature Space DBs). Let A be a linear region of f (as in Definition 1), and $db_{i,j}{}^{o}$ be a DB in the output space between the (different) classes i and j. Then, the corresponding DB in

the linear region A in the feature space can be explicitly calculated by substituting $y = T \cdot x + c$ in Term (2) for $x \in A$ and for the affine linear mapping T, c representing f on A:

$$\begin{aligned} v_{i,j} \cdot T \cdot x + v_{i,j} \cdot c \ (= v_{i,j} \cdot y) = 0, \\ v_{k,i} \cdot T \cdot x + v_{k,i} \cdot c \ (= v_{k,i} \cdot y) \leq 0 \end{aligned} \tag{3}$$

In this term, $v_{i,j} \in \mathbb{R}^n$ is defined as the vector with 1 in the i-th entry, -1 in the j-th entry and 0 otherwise, which means that $v_{i,j} \cdot y = y_i - y_j$. Note that it is possible that the set of $x \in A$ satisfying these (in)equalities might also be empty, which is the case if (and only if) the linear region A contains no DB between the classes i and j. For example, again for i =NORMAL, j =SUSPECT, and k =PATHOLOGICAL in Figure 3, db_4 is characterized by the equality $v_{i,j} \cdot T_3 \cdot x + v_{i,j} \cdot c_3 = 0$ which guarantees $y_i - y_j = 0$, the inequality $v_{k,i} \cdot T_3 \cdot x + v_{k,i} \cdot c_3 \leq 0$ which guarantees $y_{i/j} \geq y_k$, and the inequalities $M_3 x \leq l_3$ describing the linear region A_3.

Definition 3 (Feature Space DBs). By considering all pairs of classes and linear regions, the entirety of DBs of f in the feature space is given by

$$\begin{aligned} v_{i,j} \cdot T_u \cdot x + v_{i,j} \cdot c_u = 0, \\ v_{k,i} \cdot T_u \cdot x + v_{k,i} \cdot c_u \leq 0, \\ M_u x \leq l_u \end{aligned} \tag{4}$$

for all $1 \leq i \neq j \leq n$, $k \neq i, j$ and $1 \leq u \leq t$. According to Definition 1, x satisfying the inequality $M_u x \leq l_u$ is equivalent to x lying in A_u. We denote each DB, i.e., the set of all $x \in X$ satisfying this system of (in)equalities for the linear region A_u and classes i and j, by $db_{u,i,j}$. Moreover, we denote the set of all DBs by S_{db} and the set of all DBs adjoining class i by $S_{db,i}$. For simplicity of notation, we also write $x \in S_{db}$ or $x \in S_{db,i}$ for points lying on a respective DB.

Step 4. Determining the CSs: The DBs as described in Term (4) can now be used to derive the CSs in the feature space, in which the NN attains one single class. Because the *equalities* in Term (4) divide the linear region into areas corresponding to the classes i and j, the CSs (the colored spaces in $A_1 - A_4$ in Figure 3) can be exactly represented by using their respective *in*equalities.

Definition 4 (Class Subspaces). The feature space can be divided into (convex) subspaces, in which the NN attains only class i, by the system of inequalities (iterating over $k \neq i$)

$$\begin{aligned} v_{k,i} \cdot T_u \cdot x + v_{k,i} \cdot c_u \leq 0, \\ M_u x \leq l_u \end{aligned} \tag{5}$$

for all linear regions $1 \leq u \leq t$ .We denote the CS consisting of all $x \in X$ that satisfy these inequalities for the linear region A_u and class i, by $cs_{u,i}$. Further, we denote the set of all CSs of class i by $S_{cs,i}$ and the set of all CSs by S_{cs} (we again write $x \in S_{cs,i}$ for points in a CS of class i).

Definition 4 establishes Proposition 1b and thus completes Proposition 1 for PL NNs. For one u, this system of inequalities exactly represents the CS in the linear region A_u (defined by $M_u x \leq l_u$), in which the NN predicts class i. Compared to Term (4), the class index j is now attained by

the running index k in this definition, as the output value (of any $x \in cs_{u,i}$) for class i must be higher than that of *all* other classes, including j. By this definition, two CSs $cs_{u_1,i}$ and $cs_{u_2,i}$ of the same class i adjoin each other if their respective linear regions A_{u_1} and A_{u_2} adjoin each other.

5.1.3 EXPLORE-Method for Non-Piecewise Linear ML Models

The above instantiation of EXPLORE for NNs with PL activations aims to exactly reconstruct the ML model. To reconstruct non-PL ML models such as NNs with sigmoid activations, we can rely on existing techniques for an (arbitrarily exact) approximation by PL functions. These techniques are well established in the literature, where numerous works have shown that non-PL models can be approximated by PL ones for purposes of efficiency and theoretical analysis (Hinton et al., 2015; Kim et al., 2024). To this end, we present two well-known categories of techniques in this section, that allow for such an approximation. We evaluate their theoretical (cf. Supplementary Material) and empirical performance (cf. Evaluation) when instantiated within EXPLORE.

A first category of techniques aims to approximate a non-PL ML model by substituting all of its non-PL components with (one or multiple) PL components. In the case of an NN, this means that the non-PL (e.g., sigmoid) activation functions as the only non-PL components are directly substituted by PL components (e.g., ReLU). These components can be generated using training, interpolation, or other approximation techniques, spanning this category. The idea of substituting and thus approximating smooth activation functions with PL ones has been widely explored. Reddy and Gujral (2025) demonstrate efficiency gains from PL replacements of sigmoid/tanh activation functions, while Chiluveru et al. (2021) develops a PL approximation of the sigmoid and tanh function based on the Remez exchange algorithm.

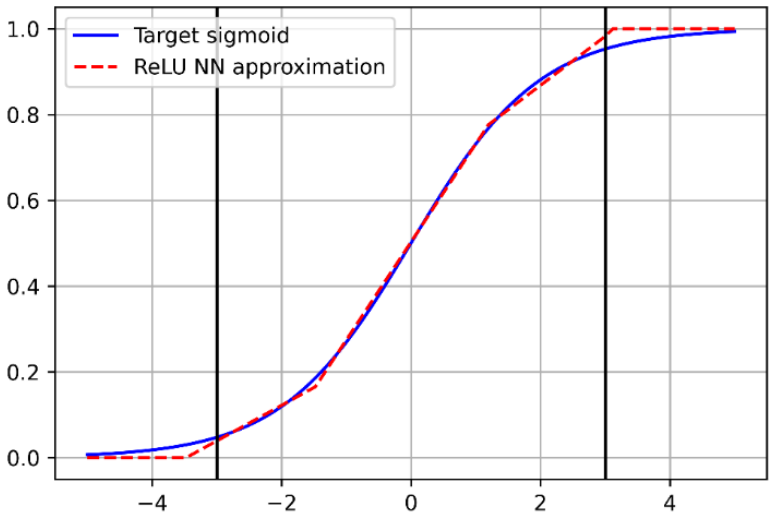


Figure 4. Functional Approximation of Sigmoid Activation Function by a ReLU NN

Utilizing this knowledge base, we propose the following instantiation for EXPLORE. To approximate a non-PL NN, we train a small ReLU NN to approximate the non-PL activation function *itself* on a defined interval of, e.g., [-3,3], and bound it to constant values (0 and 1 for sigmoid) outside this interval. We then replace each non-PL activation with this trained ReLU NN, thus

obtaining an NN with ReLU activations only, on which EXPLORE can then be directly applied. Figure 4 shows how the sigmoid function is approximated by a (PL) ReLU NN. The substitution of internal components of ML models is not restricted to non-PL NNs. For example, for SVMs it is possible to approximate non-PL kernel functions with PL ones. In such cases, EXPLORE can be directly applied to the resulting approximating PL model.

Besides this first category, the established research strand on knowledge distillation (KD) forms the basis for a second category of approximation techniques. KD pursues an input-output-oriented approach to approximate an ML model *as a whole*. Specifically, a PL model is trained on data with labels generated by the non-PL model it aims to approximate. For an NN with non-PL activation, these techniques propagate (artificial) samples through the smooth NN to train a PL NN on the resulting labels. In particular, such a technique does not rely on the internal workings of the ML model and is therefore highly generalizable to different model types. One common technique in this category trains so-called student networks to mimic teacher models (Hinton et al., 2015) by distilling smooth activations into ReLU-based ones (Kim et al., 2024).

Utilizing this knowledge base, we propose the following instantiation for EXPLORE. For any non-PL model, we first generate a larger set of labeled training data for a PL student model. To obtain a high-quality approximation, the data is generated through a sampling procedure centered around the original model's training instances, ensuring it reflects the underlying distribution—consistent with standard KD practices (Hinton et al., 2015). Using this labeled data, we train a ReLU NN to closely approximate the non-PL model. The approximation quality depends on both the size of the generated dataset and the capacity of the ReLU architecture. This approach allows us to mimic the original ML model with a PL NN that can be directly explained by EXPLORE.

Despite the methodological differences, both complementary categories of techniques share the goal of approximating a non-PL model with a PL one. The quality of this approximation becomes a central consideration when explaining ML models. In the Supplementary Material, we therefore examine the theoretical properties of these PL approximations in general. Further, we validate the empirical performance of EXPLORE when applied on PL approximations in our evaluation.

5.2 Local Explanations with EXPLORE

We now employ the functional representation of the DBs and CSs to derive feature attributions, CF examples and thus local explanations (Proposition 2). First, we introduce the concept of a local neighborhood of an instance, for which local explanations are determined.

Definition 5 (Neighborhood). For an instance $x \in X$ we define a neighborhood $\Omega_x \subset X$ as a convex subset of the feature space X with $x \in \Omega_x$.

Natural choices for a local neighborhood Ω_x can be a sphere or a cube around the instance x with radius ε (cf. Figure 5) or width ε. Other exemplary choices for Ω_x could be hemispherical or cuboid-shaped neighborhoods. For such a neighborhood, we now define the distances and directions of an instance to the DBs of an ML model f to provide information about the proximity

and position of any DB in the neighborhood. To illustrate this, we consider an Instance x from our Fetal Health dataset, together with a spherical neighborhood (solid line) in Figure 5. For illustrative purpose, we focus on the Features *Uterine Contractions* and *Mean Value of Long-Term Variability,* where the respective real-world values of x are $(1.05, 10.14)$.

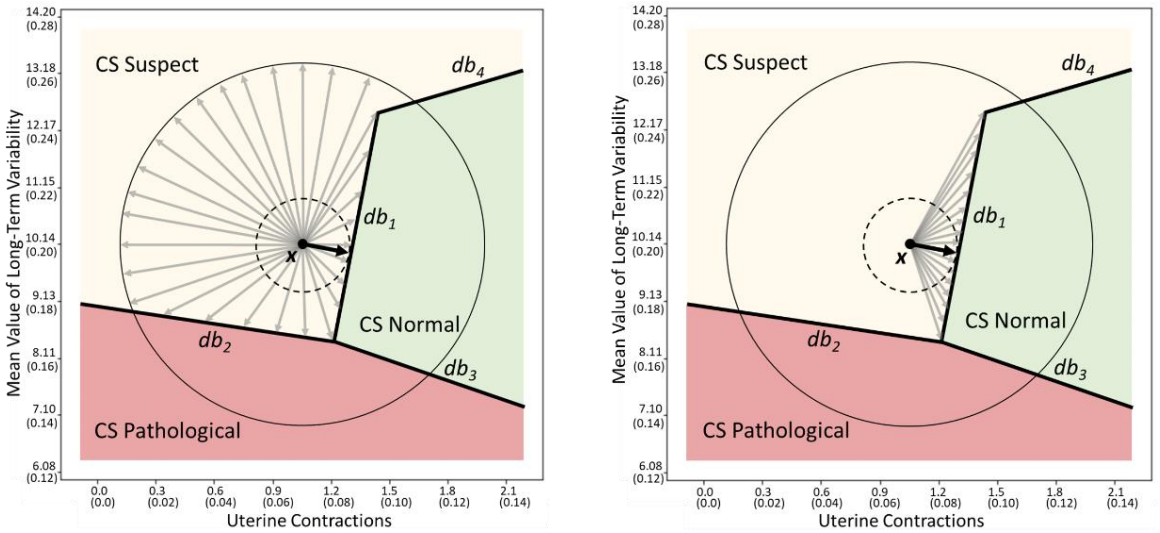


Figure 5. Conceptualizing Abductive Explanations (Class SUSPECT, left) with Insensitive Sphere (dotted sphere) and Contrastive Explanations (Class NORMAL, right) for DBs

Definition 6 (Distance to DBs and Vector). For each point x_{db} lying on a DB $db_0 \in S_{db}$, we define the direction $\overrightarrow{d_x}(x_{db})$ and distance $d_x(x_{db})$ from instance $x \in X$ to $x_{db} \in db_0$ as

$$\overrightarrow{d_x}(x_{db}) \coloneqq x_{db} - x, \tag{6}$$

$$d_x(x_{db}) = \left\|\overrightarrow{d_x}(x_{db})\right\|. \tag{7}$$

Definition 7 (Minimum Distance to DBs and Insensitive Sphere). Let $x \in X$ and $i(\leq n)$ be the class of x assigned by model f (denoted by $f(x) = i$ in the following). We define the minimum distance $d_{db,j}(x)$ of instance x to another class $j \neq i$ as the infimum of its distances to each point x_{db} lying on a DB db that adjoins class j (i.e., $x_{db} \in S_{db,j}$):

$$d_{db,j}(x) = \inf_{x_{db} \in S_{db,j}} d_x(x_{db}). \tag{8}$$

Note that in the case of any DB $db \in S_{db}$, this infimum is always attained at a point $x_{db,min}$ since any db is a closed subset of a hyperplane. Analogously to Definition 6, the vector

$$\overrightarrow{d_{db,j}}(x) \coloneqq x_{db,min} - x \tag{9}$$

defines the direction of x to the closest point on a DB towards class j. Instance x is classified as SUSPECT by the NN and its minimum distance to Class NORMAL is represented by the black-colored vector in Figure 5. For this standardized vector, $\overrightarrow{d_{db,Normal}}(x) = \begin{pmatrix} 0.017 \\ -0.003 \end{pmatrix}$ to DB db_1 (corresponding to $\begin{pmatrix} 0.25 \\ -0.16 \end{pmatrix}$ in real-world units) and $d_{db,Normal}(x) = 0.017$ holds. On this basis, the so-called Insensitive Sphere of x with respect to a certain class j can be defined as

$$IS_j(x) = \{x' \in X \mid |x' - x| < d_{db,j}(x)\}. \quad (10)$$

Since $d_{db,j}(x)$ defines the minimum distance from x to class j, $IS_j(x)$ determines the largest sphere around x in which class j is not attained by model f. By taking the overall minimum distance to any DB $d_{db}(x) = \min_{j \neq i} d_{db,j}(x)$ (over all classes $j \neq i$), we obtain the Insensitive Sphere $IS(x)$, in which no class changes can occur at all and class i is retained by f. In this sense, f is insensitive to any value changes of x within $IS(x)$. As the vector $\overrightarrow{d_{db,Normal}}(x)$ also realizes the overall minimum distance to any DB, its length defines the radius of the dotted Insensitive Sphere IS of x which is particularly important for *abductive explanations* because the class SUSPECT remains unchanged. This implies that any change in one single feature of less than 1.7 percent of its observed standardized value range, or any change of multiple features with (Euclidean) relative length of 0.017 will not result in a different class assignment.

We now present our *Vector-based Feature Attribution* (VFA). As already discussed, we adopt the position of the considered instance and observe the feature space and all DBs from its point of view. In particular, regarding VFA, we aim to analyze the instance locally by carefully focusing on the feature space *up to the first DB* in any direction.

Definition 8 (Vector-based Feature Attribution). Let $db \in S_{db,j}$ be a DB to class j in the neighborhood Ω_x of $x \in X$. Moreover, let the point $x_{db} \in db$, such that no other DB of class j lies between x and x_{db}. We define the VFA with respect to j as the direction vector towards x_{db}

$$VFA^j_{DB,x}(x_{db}) = \overrightarrow{d_x}(x_{db}), \quad (11)$$

with the value of feature $k(\leq m)$ being defined as the k-th entry of $VFA^j_{DB,x}(x_{db})$.

The vector $\overrightarrow{d_x}(x_{db})$ contains the exact changes in each feature value required to reach the point $x_{db} \in db$, naturally representing the attribution of each feature for the change towards this DB point. In particular, not only the extent but also the direction of the change (i.e., the sign of the entry) and its distance (i.e., how close the DB lies) become evident providing feature values in the sense of both abductive and contrastive / CF explanations: If x is of class i, $VFA^i_{DB,x}(x_{db})$ indicates the maximum feasible change of the feature values in the direction of point x_{db} while still maintaining the current class (abductive explanation). For any class $j \neq i$ and any point x_{db} on the DB to class j, $VFA^j_{DB,x}(x_{db})$ defines the minimum required feature value changes in this direction to change the assignment to class j (contrastive / CF explanation). For instance, each of the illustrated gray vectors in Figure 5 (left) pointing from $x = (1.05, 10.14)$ to either the first DB in this direction or the solid edge of the spherical neighborhood is an abductive $VFA^{Suspect}_{DB,x}$, yielding class retention. Further, any vector pointing to db_1 represents a contrastive $VFA^{Normal}_{DB,x}$ (right side of Figure 5), providing a change to class NORMAL. Moreover, also the relevance and attributions of features for a class retention or change can be directly derived from the VFA vectors. For Instance x, one feasible $VFA^{Normal}_{DB,x}$ is given by (0.018, 0) indicating that no change in the attribute *Mean Value of Long-Term Variability* is necessary for a change to class NORMAL.

The functional representation of the DBs from Lemma 1 allows the determination of any DB point x_{db} and thus, of any $VFA^{j}_{DB,x}$ vector. This yields Proposition 2a with respect to DBs.

Many XAI methods such as SHAP aggregate their (abductive) explanations. This can obviously be done very easily with the VFA leading to the *Set-based Feature Attribution* (SFA).

Definition 9 (Set-based Feature Attribution). For $x \in X$ and class j, we denote the set of all feasible VFA vectors with respect to j as

$$SFA^{j}_{DB,x} = \{VFA^{j}_{DB,x}(x_{db}) \mid x_{db} \in S_{db,j} \cap \Omega_{x,j}\}. \tag{12}$$

The $SFA^{j}_{DB,x}$ precisely comprises the vectors towards all points on DBs of class j that do not intersect any other DBs to class j. Crucially, $SFA^{j}_{DB,x}$ allows a careful analysis of the neighborhood and its DBs by using (set) operators such as the minimum, maximum, average, diameter or variance of feature value changes to achieve class retention or change (cf. Supplementary Material for information on different operators). Such operators enable various analyses, e.g., determining which feature value changes are (not) necessary and sufficient for class retention or change. For example, analyzing $SFA^{Normal}_{DB,x}$ for changing Instance x to Class NORMAL (cf. Figure 5 right side) shows that for the Feature *Uterine Contractions*, a positive real-world value change of at least 0.17 is necessary to attain this change. In contrast, for the Feature *Mean Value of Long-Term Variability* the $SFA^{Normal}_{DB,x}$ contains $VFA^{Normal}_{DB,x}$ vectors with possible positive changes (up to 2.33), possible negative changes (down to -1.74) and even a $VFA^{Normal}_{DB,x}$ vector where no change in this feature is necessary. Thus, in this regard, *Uterine Contractions* is more important as it must be increased by at least 0.17. Based on the computation of any VFA_{DB} vector following Lemma 1, an accurate and complete determination of SFA_{DB} is guaranteed.

Both, VFA and SFA can be extended by considering not only the DBs, but the CSs to provide additional insights into the depth and volume of CSs. We briefly outline this in the following.

Definition 10 (Vector-based Feature Attribution CS). Let $cs \in S_{cs,j}$ be a CS of class j in the neighborhood Ω_x of $x \in X$, and $\overrightarrow{d_x}(x_{cs})$ the direction vector of x towards an arbitrary point $x_{cs} \in cs$. We define the VFA_{CS} as the direction vector towards x_{cs}

$$VFA^{j}_{CS,x}(x_{cs}) = \overrightarrow{d_x}(x_{cs}), \tag{13}$$

with the value of feature $k(\leq m)$ being defined as the k-th entry of $VFA^{j}_{CS,x}(x_{cs})$.

Analogously to Definition 8, this provides both an accurate feature attribution or CF examples in the sense of Proposition 2a. For any point $x_i \in \Omega_x$ in a CS of the same class i as the Instance x, the vector $VFA^{i}_{CS,x}(x_i)$ represents an abductive explanation (see Figure 6 left; any vector from x to a point x_i in the yellow CS Suspect). The analogous holds for CF examples (see Figure 6 right). Again, all possible (abductive or contrastive) VFA_{CS} vectors are included in SFA_{CS}.

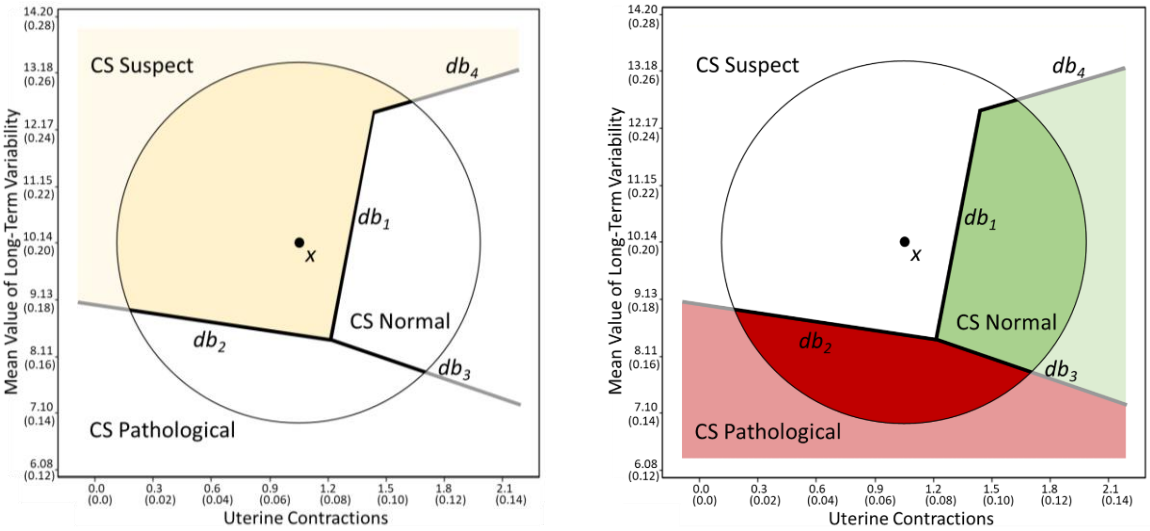


Figure 6: Conceptualizing Abductive Explanations (Class SUSPECT, left) and Contrastive Explanations with respect to CS NORMAL and CS PATHOLOGICAL (right) respectively.

Definition 11 (Set-based Feature Attribution CS). For $x \in X$ and class j, we denote the set of all feasible VFA_{CS} with respect to the subspace of class j as

$$SFA^{j}_{CS,x} = \{VFA^{j}_{CS,x}(x_{cs}) \mid x_{cs} \in S_{CS,j}\}. \tag{14}$$

The $SFA^{j}_{CS,x}$ precisely comprises the vectors towards all points in the CS of class j. Moreover, the $SFA_{CS,x}$ of each CS can again be utilized to derive results on sufficient and necessary feature value changes to reach and retain the respective CS. In our example, considering the change to CS NORMAL, $SFA^{Normal}_{CS,x}$ contains a $VFA^{Normal}_{CS,x}$ for every value in the range of [0.17, 0.90] for the Feature *Uterine Contractions*, and more specifically all vectors of the form $\binom{val}{0.0}$ for $val \in [0.3, 0.9]$, indicating the substantial depth of the CS in this dimension. Moreover, the $VFAs$ with the maximum and minimum attribution value for the second Feature *Mean Value of Long-Term Variability* are given by $\binom{0.52}{2.43}$ and $\binom{0.60}{-2.24}$ respectively. This not only shows the extension of the CS, but also that a simultaneous increase in the Feature *Uterine Contractions* is necessary to attain this CS, thus revealing the existing interaction between these features. This can support a user's decision making by providing information about the depth, volume and expansion of CSs beyond a DB. In summary, both SFA_{DB} and SFA_{CS} enable a wide range of user-specific and context-dependent options for accurate and complete analyses. Finally, we address that, in reality, the feature space may be constrained by specific characteristics of the considered features.

Definition 12 (Feature Sub Spaces as Neighborhoods). For $x \in X$ in a neighborhood Ω_x, any subset of features and any set of constraints on these features (such as minimum and maximum allowed feature values) can be selected. Given the functional representation of DBs and CSs, for each such choice, all introduced local explanations (VFA and SFA for DBs and CSs) can be accurately determined on the subset of the neighborhood that satisfies these constraints (Proposition 2b; for a formalization of restricted neighborhoods, cf. Supplementary Material).

The following section evaluates our proposed method and the proposed local explanations.

6 Evaluation

In this section, we briefly introduce the five real-world datasets that we use in our evaluation experiments and discuss existing XAI methods against which we compare the performance of Explore. Following the evaluation frameworks in Agarwal et al. (2022) and partly also Nauta et al. (2023) and Pawlicki et al. (2024), we employ the criteria fidelity, robustness, and faithfulness of local explanations. These criteria are fundamental for the (quantitative) evaluation of local XAI methods and are widely adopted in XAI literature (Nauta et al., 2023).

6.1 Describing Datasets, ML Methods, and XAI Methods

To evaluate our method, we used five datasets which are related to critical decision situations in healthcare, loan approval, criminal recidivism, and traffic safety and thus are already known in XAI literature. If ML model predictions are used for decision support in these domains, the explanations for these predictions must be accurate, robust, and provide faithful feature attributions, as they can have potentially serious consequences for an individual's life. The first dataset is the Fetal Health dataset which we already introduced as our running example. As a second dataset, we use the Prosper dataset (similarly used in, e.g., Fu et al., 2021), which contains data from the loan company "Prosper" on completed or still pending loans between 2005 and 2014. Each data instance includes information on the loan applicant, their previous loans, and their default status (i.e., whether they defaulted or not). Based on this data, ML models were trained and used to predict the future status of a loan application, i.e., whether it will be fully repaid or defaulted, and if it is defaulted, the estimated loss (on a scale of 1 to 4 from low to very high). As the third dataset, we use the well-known COMPAS dataset (also used in, e.g., Alvarez-Melis & Jaakkola, 2018b or Agarwal et al., 2022) which contains personal information and recidivism risk scores of criminal defendants from Broward County, Florida who were assessed in 2013 and 2014. As in existing studies, ML models were trained and used to predict the likelihood of (violent) reoffending, represented on a scale from 1 to 5 from very low to very high. As the fourth dataset, we use traffic data from the Highway Safety Information System (HSIS) which is a multistate database from the US Department of Transportation that contains crash, roadway inventory, and traffic volume data for a select group of US states. From this HSIS database, we consider traffic accidents that occurred in North Carolina in 2022. ML models were trained to predict crash severity and human injury based on information about road characteristics and the drivers and vehicles involved. More precisely, the classes are given by the type of human injury which can be none, possible, minor, major, or fatal. As the fifth dataset, we use a version of the well-known MIMIC-III database (also used in, e.g., Guo et al., 2025) which contains a wide range on health-related data associated with over 40,000 hospital patients who stayed in the intensive care units of the Beth Israel Deaconess Medical Center between 2001 and 2012. We follow the data preprocessing as in Harutyunyan et al. (2019) and consider the ML classification task that predicts the combination of patient mortality (whether the patient dies in the first 48 hours of hospitalization) and estimated length of stay (whether the patient stays longer than 24 hours), resulting in four possible classes. The main characteristics of the datasets are:

DATASET	#Instances	#Features	#Classes	Domain
Fetal Health	2,126	21	3	Healthcare
Prosper	113,937	47	5	Loan Approval
COMPAS	18,316	15	5	Criminal Recidivism
HSIS	218,490	19	5	Traffic Safety
MIMIC	21,139	17	4	Healthcare

Table 4. Dataset Information

To address the known class imbalance of the datasets while focusing only on real data, i.e., without generating artificial (probably biased) data instances, we employed under-sampling for all datasets so that each class occurs at most twice as often as the minority class. This is motivated by the fact that safety critical problem settings often suffer from severe class imbalance because the worst-case outcome is typically rather rare (e.g., fatal accidents in the HSIS dataset or a pathological fetus in the Fetal Health dataset) but must not be disregarded by ML models in training. We then performed feature standardization (min-max normalization to the unit interval [0,1]) and trained—based on a fivefold cross-validation for each dataset— different classifier models comprising both, PL and non-PL ML model types. For PL model types, we consider feedforward (ReLU) NN, DT, RF, SVM, and AdaBoost. For non-PL model types, we employ NNs with sigmoid and tanh activations as well as SVMs with (non-linear) polynomial and RBF kernels. For each classifier and dataset, we optimize the hyperparameter configuration with respect to balanced accuracy (for details see Supplementary Material). Using the optimal hyperparameters, we trained five models for each dataset and model type and used them all in our experiments to obtain a higher number of test instances for all datasets. For all experiments we generated explanations for up to 1,000 arbitrarily selected test data instances (i.e., 200 per fold).

As competing methods for our evaluation, we selected widely known (indicated by citations), recently published local XAI methods with publicly available code. For our experiments, we adopted their respective versions for tabular data with the recommended or predefined parameter configurations in the code repositories. First, for methods providing surrogate models and feature attributions, we employ the well-known method LIME (Ribeiro et al., 2016). Following to the publicly available code, the surrogate model was instantiated as a Ridge regression in our evaluation. Moreover, we consider S-LIME (Zhou et al., 2021), a more recent and well-published variant of LIME that aims to provide more robust explanations upon repeated use. The method LS (Laugel et al., 2018b) is also widely recognized and generates a local linear surrogate model based on a modified sampling procedure compared to LIME, whereas MAPLE (Plumb et al., 2018) utilizes a RF-based feature selection to train a local linear surrogate model. We also employ the well-known rule-based explanation method ANCHORS (Ribeiro et al., 2018) which explains the classification decision of the ML model by a set of interpretable rules. In our analyses, we consider the rule set provided by ANCHORS as a one-vs-all surrogate model which predicts the class of the data instance to be explained within the given rule set. As second rule-based method, we consider LORE (Guidotti et al., 2018) which provides a DT as an interpretable local surrogate model based on a genetic algorithm. Further, we consider ROLEX from Kim et al. (2023), a recently published advanced method based on surrogates. Following the authors' suggestions as

well as their requested original code implementation (including the configuration for ROLEX), a linear Ridge regression model is used in the case of linear DBs and a non-linear DT model in the case of non-linear, multiclass DBs in the experiments. Corresponding to literature, we evaluate all methods that provide surrogate models regarding fidelity and robustness.

For our faithfulness analysis, we consider all methods that use linear regressions (LIME, ROLEX, LS, MAPLE, S-LIME) since they naturally provide feature attributions via their coefficient vectors. We also evaluate SHAP (Lundberg & Lee, 2017), employing its model-specific implementation (*LinearExplainer* for Table 7). Since the feature attribution values provided by SHAP cannot be interpreted as a local surrogate model (in particular, it is not intended to apply them on other data instances around the instance to be explained), they cannot be evaluated regarding fidelity and robustness. Finally, we consider CF methods in our faithfulness evaluation using well-known performance metrics. We first employ the method DICE (Mothilal et al., 2020) which is known for generating multiple high-quality CFs (Moreira et al., 2025) in its three main variants KDTREE, RANDOM, and GENETIC. Further, we consider the established methods FGM (Goodfellow et al., 2015) and PGD (Madry et al., 2018) which rely on gradient-based perturbations. Finally, we consider GSCF (Laugel et al., 2018a) since it uses random sampling to identify a close spherical neighborhood in which the class prediction is expected to change.

EXPLORE is evaluated in all above introduced settings. It is either applied directly on the already PL model or on a PL approximation of the non-PL model. For evaluating EXPLORE on both non-PL sigmoid and tanh NNs, we employ an instantiation for each approximation category described in Section 5.1.3. The first instantiation involves substituting each non-PL activation function by a small, two-layer ReLU subnet. The second instantiation is grounded on KD and uses a ReLU NN student model which was trained to mimic the non-PL original NN. Due to its high flexibility, this second KD-based technique is also applied for both non-PL SVM models with polynomial and RBF kernel respectively. We provide details on the instantiations and configurations of both techniques for our evaluation in our Supplementary Material. Beyond this approximation, no further configuration was applied for assessing EXPLORE (as deterministic in nature).

6.2 Evaluating Fidelity, Robustness and Faithfulness of Explanations

We first evaluate the fidelity by quantifying the alignment of the predictions of the XAI methods to those of the ML model (e.g., Laugel et al., 2018b; Saarela & Geogieva, 2022). We adopt the demanding DB-aware LDA score as presented by Kim et al. (2023) which determines the average local fidelity scores over all instances near of at least one DB. Hereby, the local fidelity score for a test instance $x_i \in X$ and its local explanation $\bar{f}_{x_i}$ for the ML model f is defined as the accuracy

$$LocalFid\left(\bar{f}_{x_i}, x_i\right) = Acc_{z \in Z}\left(f(z), \bar{f}_{x_i}(z)\right). \tag{15}$$

In this term, Z is a set of random samples lying in the feature space close to x_i. As a result, the local fidelity is 1 if an XAI method $\bar{f}_{x_i}$ predicts the same class as the ML model f on all random samples. Finally, the LDA score is calculated by averaging the local fidelities for only those instances x_i (overall, up to 1,000 instances per dataset) for which f predicts at least two different classes on Z. If this condition holds, then there is at least one DB near x_i. As a result, 'trivial' instances without any nearby DBs are excluded. Table 5 presents the LDA scores for all five datasets. For the sigmoid and tanh NNs, the fidelity of EXPLORE using the technique replacing the non-PL activation functions is reported first, and the KD-based technique is reported second.

The results show that for the Fetal Health dataset, the LDA scores remain relatively high on average for some XAI methods, whereas the scores for the remaining datasets are well below one[3]. This not only exposes the mismatches between the predictions of the ML model and the XAI methods for many instances but also indicates highly problematic explanations, i.e., the XAI method provides an explanation for a class that the ML model did not predict for the instance.

Dataset	Method	NN	DT	RF	SVM	Ada-Boost	NN (sigmoid)	NN (tanh)	SVM (poly)	SVM (RBF)
Fetal Health	EXPLORE	1.0	1.0	1.0	1.0	1.0	0.99/0.99	0.99/0.99	0.97	0.99
	LIME	0.74	0.59	0.62	0.62	0.64	0.68	0.66	0.68	0.67
	ROLEX	0.93	0.96	0.87	0.91	0.92	0.92	0.89	0.95	0.96
	LS	0.68	0.71	0.72	0.68	0.67	0.68	0.66	0.61	0.65
	MAPLE	0.72	0.61	0.66	0.71	0.72	0.73	0.73	0.71	0.74
	ANCHORS	0.51	0.47	0.50	0.53	0.50	0.46	0.47	0.35	0.46
	S-LIME	0.74	0.59	0.62	0.62	0.64	0.67	0.66	0.72	0.71
	LORE	0.40	0.40	0.40	0.38	0.42	0.31	0.32	0.37	0.36
Prosper	EXPLORE	1.0	1.0	1.0	1.0	1.0	0.98/0.98	0.98/0.97	0.98	0.97
	LIME	0.64	0.47	0.82	0.56	0.45	0.62	0.59	0.70	0.68
	ROLEX	0.74	0.75	0.85	0.80	0.93	0.76	0.73	0.93	0.92
	LS	0.45	0.31	0.75	0.37	0.46	0.38	0.40	0.70	0.73
	MAPLE	0.46	0.36	0.48	0.54	0.71	0.47	0.47	0.65	0.64
	ANCHORS	0.46	0.47	0.25	0.56	0.50	0.57	0.56	0.44	0.41
	S-LIME	0.66	0.5	0.82	0.58	0.49	0.64	0.63	0.72	0.67
	LORE	0.54	0.50	0.69	0.60	0.70	0.52	0.57	0.52	0.52
COMPAS	EXPLORE	1.0	1.0	1.0	1.0	1.0	0.98/0.97	0.98/0.97	0.96	0.97
	LIME	0.55	0.38	0.47	0.66	0.26	0.51	0.47	0.59	0.61
	ROLEX	0.66	0.47	0.45	0.69	0.52	0.65	0.57	0.83	0.84
	LS	0.40	0.23	0.27	0.25	0.30	0.39	0.37	0.56	0.43
	MAPLE	0.59	0.35	0.44	0.63	0.36	0.56	0.53	0.69	0.70
	ANCHORS	0.55	0.54	0.58	0.51	0.57	0.58	0.56	0.50	0.49
	S-LIME	0.42	0.38	0.47	0.66	0.26	0.51	0.44	0.59	0.61
	LORE	0.52	0.39	0.48	0.61	0.28	0.37	0.34	0.42	0.42
HSIS (Traffic)	EXPLORE	1.0	1.0	1.0	1.0	1.0	0.96/0.97	0.95/0.96	0.95	0.95
	LIME	0.24	0.26	0.31	0.38	0.35	0.35	0.29	0.48	0.39
	ROLEX	0.61	0.73	0.54	0.86	0.84	0.70	0.67	0.81	0.83
	LS	0.34	0.30	0.34	0.25	0.35	0.35	0.33	0.42	0.41
	MAPLE	0.46	0.29	0.34	0.47	0.42	0.56	0.45	0.67	0.71
	ANCHORS	0.58	0.53	0.50	0.52	0.53	0.56	0.53	0.62	0.52

[3] If we additionally consider the (class) balanced local fidelity scores, which would be particularly important for our high-stakes dataset, these scores would mostly be considerably lower (e.g., for Lime and the corresponding row in the Fetal Health dataset, the scores are 0.48, 0.39, 0.44, 0.44, and 0.43).

	S-LIME	0.24	0.26	0.32	0.38	0.34	0.35	0.29	0.49	0.38
	LORE	0.48	0.37	0.49	0.74	0.54	0.60	0.53	0.61	0.57
MIMIC	EXPLORE	1.0	1.0	1.0	1.0	1.0	0.99/0.96	0.98/0.97	0.96	0.99
	LIME	0.33	0.27	0.30	0.49	0.38	0.36	0.27	0.75	0.75
	ROLEX	0.39	0.37	0.38	0.35	0.53	0.23	0.38	0.82	0.98
	LS	0.21	0.29	0.28	0.31	0.15	0.50	0.34	0.32	0.69
	MAPLE	0.28	0.26	0.26	0.49	0.30	0.27	0.31	0.51	0.74
	ANCHORS	0.55	0.50	0.49	0.56	0.51	0.59	0.62	0.50	0.50
	S-LIME	0.33	0.27	0.30	0.49	0.38	0.37	0.26	0.40	0.75
	LORE	0.43	0.32	0.38	0.41	0.59	0.19	0.35	0.52	0.47

Table 5. Fidelity/LDA Scores[4]

For all PL model types, EXPLORE achieves a fidelity of 1 which can be substantiated by a theoretical proof that guarantees a perfect LDA score: Let Z be the set of all random samples and $z \in Z$. Our representation of CSs allows to identify the CS $cs_{u,i}$ of the ML model f that contains z by checking for which linear region A_u and class i the corresponding system of inequalities is satisfied by z (cf. Definition 4). Since Lemma 1 guarantees the validity of the inequalities describing the CS, the class predicted by f agrees with the one associated to $cs_{u,i}$, namely i. Since this holds for all $z \in Z$, the local fidelity of any instance and hence, the LDA score of EXPLORE always equals 1. For all non-PL model types, EXPLORE achieves very high fidelity scores of at least 0.95 for both PL approximations across all datasets showing their effectiveness and the ability of EXPLORE to accurately match the prediction of the underlying non-PL model.

To evaluate the second criterion robustness, we need to assess the extent to which the outcomes of XAI methods on equal or similar instances are stable and reproducible. Aligned with Dombrowski et al. (2019) and Lakkaraju et al. (2020), we first generate explanations for a certain number $J \in \mathbb{N}$ of similar artificial samples $x_{i,j} \in X$ very close to each test instance x_i. Then, the criterion is defined as the agreement between all local explanations of similar samples $\bar{f}_{x_{i,j}}$ and the explanation $\bar{f}_{x_i}$ of the test instance on a set a of random samples Z:

$$Rob\left(\bar{f}_{x_i}, x_i\right) = \frac{1}{J}\sum_{j=1}^{J} Acc_{z\in Z}\left(\bar{f}_{x_i}(z), \bar{f}_{x_{i,j}}(z)\right) \tag{16}$$

In case of perfect robustness, the predictions of the explanation model $\bar{f}_{x_{i,j}}$ on each random sample $z \in Z$ fully match with the predictions of the test instance explanation model $\bar{f}_{x_i}$. Then, the accuracy $Acc_{z\in Z}\left(\bar{f}_{x_i}(z), \bar{f}_{x_{i,j}}(z)\right)$ would be 1 for each explanation $\bar{f}_{x_{i,j}}$ (also for instances close to DBs), leading to the perfect score $Rob\left(\bar{f}_{x_i}, x_i\right) = 1$ for x_i. Table 6 shows the results when generating five (i.e., $J = 5$) artificial samples whose feature values differ only by at most one percent of the respective features total value space, for each method and dataset. Our empirical analysis on all five datasets indicates that the robustness of some existing XAI methods such as

[4] When evaluating the runtime of the XAI methods, some of them (e.g., LIME, LS, and S-LIME) rarely exhibit shorter runtimes, while other (e.g., ROLEX, and especially LORE) have longer runtimes compared to EXPLORE across all datasets. Except for the LORE method, for which we had to reduce the number of test instances in the subsequent robustness evaluation, the runtimes on all datasets are unproblematic.

LORE is limited, i.e., that the generated surrogates are not stable even in close proximity of the test instance. Others, e.g., LIME, S-LIME, MAPLE—at first glance—have high robustness. However, it should be noted that according to Table 5, especially these XAI methods exhibit partially poor fidelity, resulting in 'robust' but poor generated explanations.

Dataset	Method	NN	DT	RF	SVM	Ada-Boost	NN (sig-moid)	NN (tanh)	SVM (poly)	SVM (RBF)
Fetal Health	EXPLORE	1.0	1.0	1.0	1.0	1.0	1.0/1.0	1.0/1.0	1.0	1.0
	LIME	0.91	0.83	0.94	0.92	0.92	0.91	0.89	0.97	0.95
	ROLEX	0.94	0.98	0.91	0.91	0.96	0.94	0.91	0.95	0.95
	LS	0.85	0.88	0.88	0.87	0.89	0.87	0.84	0.90	0.91
	MAPLE	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
	ANCHORS	0.93	0.96	0.91	0.98	0.9	0.92	0.92	0.90	0.80
	S-LIME	0.95	0.95	0.98	0.97	0.97	0.95	0.95	0.98	0.98
	LORE	0.63	0.61	0.60	0.62	0.63	0.63	0.58	0.60	0.61
Prosper	EXPLORE	1.0	1.0	1.0	1.0	1.0	1.0/1.0	1.0/1.0	1.0	1.0
	LIME	0.78	0.67	0.99	0.87	0.97	0.83	0.70	0.86	0.86
	ROLEX	0.74	0.77	0.88	0.81	0.94	0.76	0.72	0.91	0.90
	LS	0.61	0.58	0.81	0.68	0.87	0.64	0.64	0.84	0.86
	MAPLE	0.70	0.64	0.59	0.87	0.97	0.77	0.72	0.91	0.91
	ANCHORS	0.95	0.80	0.89	0.96	1.0	0.96	0.96	0.91	0.97
	S-LIME	0.92	0.90	1.0	0.95	0.98	0.93	0.87	0.95	0.95
	LORE	0.43	0.44	0.59	0.55	0.66	0.44	0.47	0.45	0.44
COMPAS	EXPLORE	1.0	1.0	1.0	1.0	1.0	1.0/1.0	1.0/1.0	1.0	1.0
	LIME	0.94	0.72	0.83	0.99	0.9	0.92	0.88	0.97	0.97
	ROLEX	0.83	0.67	0.61	0.82	0.60	0.81	0.72	0.96	0.97
	LS	0.60	0.45	0.43	0.65	0.47	0.6	0.57	0.85	0.52
	MAPLE	0.93	0.71	0.72	0.93	0.86	0.94	0.92	0.94	0.95
	ANCHORS	0.97	0.9	0.94	0.98	0.91	0.95	0.97	0.98	0.96
	S-LIME	0.93	0.90	0.92	0.98	0.90	0.96	0.93	0.98	0.98
	LORE	0.36	0.31	0.36	0.38	0.35	0.36	0.35	0.39	0.39
HSIS (Traffic)	EXPLORE	1.0	1.0	1.0	1.0	1.0	1.0/1.0	1.0/1.0	1.0	1.0
	LIME	0.78	0.57	0.75	0.87	0.84	0.85	0.79	0.92	0.91
	ROLEX	0.64	0.80	0.52	0.85	0.88	0.72	0.70	0.77	0.81
	LS	0.73	0.67	0.69	0.70	0.70	0.74	0.72	0.80	0.81
	MAPLE	0.90	0.85	0.85	0.92	0.91	0.91	0.90	0.93	0.97
	ANCHORS	0.93	0.94	0.87	0.95	0.9	0.94	0.98	0.90	0.77
	S-LIME	0.89	0.83	0.87	0.88	0.83	0.92	0.89	0.97	0.96
	LORE	0.49	0.42	0.50	0.72	0.57	0.58	0.53	0.60	0.57
MIMIC	EXPLORE	1.0	1.0	1.0	1.0	1.0	1.0/1.0	1.0/1.0	1.0	1.0
	LIME	0.69	0.29	0.39	1.0	0.46	0.57	0.70	0.41	0.59
	ROLEX	0.48	0.35	0.37	0.51	0.50	0.44	0.49	0.50	0.68
	LS	0.54	0.44	0.41	0.50	0.41	0.50	0.42	0.62	0.81
	MAPLE	0.77	0.49	0.52	0.87	0.58	0.76	0.77	0.77	0.80
	ANCHORS	0.79	0.69	0.67	0.92	0.78	0.82	0.84	0.95	0.92
	S-LIME	0.75	0.35	0.45	1.0	0.45	0.73	0.82	0.70	0.85
	LORE	0.50	0.29	0.35	0.49	0.54	0.41	0.44	0.66	0.62

Table 6. Robustness Scores

EXPLORE achieves a robustness of 1 which can again be substantiated by providing a theoretical proof for perfect robustness: Let $x_i \in X$ be the considered test instance and let $x_{i,j} \in X$ be an (artificial) sample in close proximity of x_i. Our proof for the perfect fidelity of EXPLORE for both explanations $\bar{f}_{x_{i,j}}$ and $\bar{f}_{x_i}$ implies that the class predictions of $\bar{f}_{x_{i,j}}$ and $\bar{f}_{x_i}$ are equal to the class

predictions of f on all samples $z \in Z$. This implies that the class predictions of $\bar{f}_{x_{i,j}}(z)$ and $\bar{f}_{x_i}(z)$ are also equal for all samples $z \in Z$. As a result, $Acc_{z \in Z}\left(\bar{f}_{x_i}(z), \bar{f}_{x_{i,j}}(z)\right) = 1$ for an arbitrary artificial instance $x_{i,j}$ and $Rob\left(\bar{f}_{x_i}, x_i\right) = 1$ follows. Note that this result also holds for the non-PL model types, as EXPLORE is deterministic (and thus generate stable and reproducible results) also when applied to the PL approximation models.

As third criterion, the faithfulness of local explanations is evaluated, expressing whether the relevant features and their attributions are correctly determined. To assess faithfulness, it is necessary to know the actual ground-truth explanations which are typically unknown for black-box ML models. For this reason, the literature often suggests utilizing ground-truth explanations of white-box ML models where the ground truth can be directly derived by the model parameters such as the coefficients of logistic regression models (Nauta et al., 2023). As literature points out, such white-box ML models hence enable an evaluation of faithfulness (e.g., Jia et al., 2019; Velmurugan et al., 2025), but we are also aware that they often represent a lower bound for general faithfulness due to their lower complexity, i.e., they are easier to approximate especially for methods using linear surrogates such as, e.g., LIME or MAPLE. To apply logistic regression models in multi-class scenarios, and to ensure fairness for existing XAI methods, a one-versus-all (other classes) logistic regression approach—similar to the procedure of LIME or S-LIME—is used. For a multi-class logistic regression, the ground-truth explanation indicating the shortest path towards a DB can be derived by (iteratively) applying the orthogonal projection onto the subsets of DBs and are directly induced by the regression weights. Thereby, we focus on ground-truth explanations that were correctly classified by the logistic regression within our five datasets (Yalcin et al., 2021). In literature, the faithfulness of local explanations is typically evaluated for the top-K features (cf., e.g., Agarwal et al., 2022; Nauta et al., 2023), as it is more important for users to accurately recognize and rank the top-3 or -5 features than less important ones ranked between, e.g., 20 and 30. Therefore, we focus on K=3, 5, and 10 features and use two well-known criteria: top-K feature agreement (or top-K intersection, see Agarwal et al., 2022; Ribeiro et al., 2016) and top-K pairwise rank agreement (or top-K Kendall's tau, cf. Agarwal et al., 2022). For a feature vector $w_{\bar{f}}$ and the ground-truth feature vector w_f (each containing the features in descending order regarding importance) both faithfulness criteria are defined as follows

$$Agree_K\left(w_{\bar{f}}, w_f\right) = \frac{\left|w_{\bar{f}|K} \cap w_{f|K}\right|}{K}, \tag{17}$$

$$\tau_K\left(w_{\bar{f}}, w_f\right) = \frac{2}{K \cdot (K-1)} \sum_{i<j} sgn\left(w_{\bar{f}|K,i} - w_{\bar{f}|K,j}\right) \cdot sgn\left(w_{f|K,i} - w_{f|K,j}\right), \tag{18}$$

where $w_{\bar{f}|K}$ and $w_{f|K}$ denote the first K features of $w_{\bar{f}}$ and w_f respectively. Note this standard definition of the top-K Kendall's tau is more 'optimistic' in the sense that it does not matter whether the top-K features are ranked correctly in positions 1–5 or in positions 21–25. In both cases, a tau value of 1 is obtained. Therefore, it is necessary to interpret both criteria in close

combination, i.e., assessing whether the correct top-*K* features are actually identified (agreement) and whether they are ranked in the correct order (Kendall's tau).

Further, for all XAI methods generating CF explanations, we additionally evaluate the two well recognized performance metrics of proximity and stability (Oliveira & Martens, 2021). The proximity describes the distance of a provided CF explanation to the original instance. Proximity is essential to ensure both realism and actionability, since the idea of providing a CF to a user requires the suggested change to be as attainable as possible. To this end, we use the relative distance-based metric (e.g., Bayrak & Bach, 2024) which calculates the relative proximity of a CF explanation $\bar{f}_x$ for an instance x as

$$p_{CF}(\bar{f}_x, x) = \frac{|\bar{f}_x - x|}{|f_x - x|}. \tag{19}$$

This metric describes the distance of $\bar{f}_x$ to x, relative to the closest possible CF f_x which is known in the case of a logistic regression. A value of 1 would be perfect; any value greater than 1 indicates a proportionally worse proximity of a CF example.

Second, we assess the stability of CF methods, which is related to the criterion of robustness. It aims at the similarity of CF examples resulting from repeated applications of an XAI method on the same or very similar instances. To this end, we follow a standard approach as described in Oliveira and Martens (2021) and apply each XAI method twice to receive explanations $f_{x,1}, f_{x,2}$ for each instance x. For individual instances, stability is binary and defined as

$$s_{CF}(f_{x,1}, f_{x,2}) = \begin{cases} 1 \; if \; f_{x,1} = f_{x,2} \\ 0 \; else. \end{cases} \tag{20}$$

All results of the evaluation regarding top-*K* agreement, the Kendall's tau rank coefficient and both CF performance metrics are reported in Table 7. Again, it becomes evident that for the Fetal Health dataset the results of existing XAI methods are still better compared to the other datasets, where in some cases poor values are reported for the two faithfulness criteria. Note that since ANCHORS and LORE return a rule set or a DT as surrogate model instead of a feature importance vector, they cannot be evaluated regarding faithfulness. While certain CF methods, such as GSCF, achieve strong proximity scores, their explanations tend to be unstable. Conversely, methods that produce stable explanations, such as DICE (KDTREE), show limited performance with respect to proximity. In contrast, EXPLORE attains both perfect proximity and perfect stability.

Dataset	Method	Agreement			Kendall's Tau			Relative	Stability
		Top 3	Top 5	Top 10	Top 3	Top 5	Top 10	Proximity	
Fetal Health	EXPLORE	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	LIME	0.60	0.67	0.79	0.43	0.50	0.50	-	-
	ROLEX	0.73	0.77	0.83	0.65	0.68	0.68	-	-
	LS	0.66	0.71	0.79	0.61	0.61	0.61	-	-
	MAPLE	0.34	0.43	0.60	0.13	0.18	0.21	-	-
	S-LIME	0.59	0.68	0.80	0.42	0.50	0.51	-	-
	SHAP	0.40	0.49	0.65	0.20	0.21	0.27	-	-
	DICE (KDTREE)	0.01	0.07	0.31	0.13	0.00	0.08	8.53	1.0
	DICE (RANDOM)	0.09	0.15	0.37	-0.02	0.06	-0.11	14.70	0.0
	DICE (GENETIC)	0.02	0.06	0.29	0.08	0.05	0.09	10.08	1.0
	FGM	0.23	0.29	0.47	0.15	0.15	0.08	3.27	1.0
	PGD	0.23	0.31	0.48	0.11	0.09	0.06	3.27	0.11
	GSCF	0.13	0.24	0.48	-0.04	-0.05	-0.02	1.80	0.0
Prosper	EXPLORE	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	LIME	0.46	0.51	0.69	0.21	0.28	0.32	-	-
	ROLEX	0.44	0.55	0.65	0.22	0.27	0.38	-	-
	LS	0.23	0.30	0.42	0.04	0.04	0.13	-	-
	MAPLE	0.20	0.29	0.39	-0.01	0.02	0.15	-	-
	S-LIME	0.46	0.51	0.7	0.22	0.3	0.33	-	-
	SHAP	0.26	0.38	0.53	0.07	0.13	0.23	-	-
	DICE (KDTREE)	0.21	0.37	0.25	0.26	0.21	0.43	20.42	1.0
	DICE (RANDOM)	0.18	0.17	0.31	0.26	0.13	-0.03	47.67	0.0
	DICE (GENETIC)	0.26	0.37	0.26	0.26	0.22	0.46	52.93	1.0
	FGM	0.08	0.17	0.21	-0.29	-0.12	0.17	16.73	1.0
	PGD	0.11	0.18	0.22	-0.25	-0.09	0.15	16.73	0.06
	GSCF	0.06	0.10	0.20	-0.05	-0.04	0.00	2.52	0.0
COMPAS	EXPLORE	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	LIME	0.57	0.51	0.71	0.73	0.57	0.33	-	-
	ROLEX	0.82	0.71	0.77	0.79	0.79	0.69	-	-
	LS	0.65	0.65	0.71	0.84	0.66	0.54	-	-
	MAPLE	0.52	0.61	0.74	0.34	0.44	0.46	-	-
	S-LIME	0.57	0.51	0.72	0.74	0.57	0.34	-	-
	SHAP	0.56	0.52	0.66	0.57	0.41	0.24	-	-
	DICE (KDTREE)	0.18	0.16	0.53	0.42	0.49	-0.02	4.68	1.0
	DICE (RANDOM)	0.19	0.25	0.69	0.03	-0.21	-0.08	69.11	0.0
	DICE (GENETIC)	0.15	0.15	0.53	0.34	0.40	-0.07	11.03	1.0
	FGM	0.25	0.36	0.60	0.16	0.28	0.13	20.23	1.0
	PGD	0.31	0.37	0.62	0.06	0.29	0.12	20.23	0.02
	GSCF	0.22	0.33	0.63	0.01	0.12	0.01	1.60	0.0
HSIS (Traffic)	EXPLORE	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	LIME	0.41	0.46	0.64	0.32	0.31	0.27	-	-
	ROLEX	0.62	0.63	0.70	0.62	0.52	0.47	-	-
	LS	0.57	0.58	0.68	0.75	0.52	0.44	-	-
	MAPLE	0.51	0.57	0.70	0.64	0.43	0.41	-	-
	S-LIME	0.41	0.46	0.64	0.32	0.31	0.27	-	-
	SHAP	0.21	0.26	0.54	0.43	0.20	0.0	-	-
	DICE (KDTREE)	0.13	0.23	0.54	0.40	0.21	0.17	41.85	1.0
	DICE (RANDOM)	0.14	0.22	0.57	0.21	0.05	0.06	97.13	0.0
	DICE (GENETIC)	0.12	0.22	0.53	0.38	0.17	0.15	56.29	1.0
	FGM	0.23	0.36	0.60	-0.14	-0.01	0.12	39.27	1.0
	PGD	0.15	0.31	0.67	-0.05	-0.08	0.07	39.27	0.0
	GSCF	0.16	0.28	0.54	-0.02	-0.03	0.04	1.79	0.0
MIMIC	EXPLORE	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	LIME	0.21	0.69	0.73	-0.04	-0.10	0.37	-	-
	ROLEX	0.13	0.3	0.64	0.18	0.24	-0.02	-	-
	LS	0.42	0.54	0.73	0.38	0.19	0.32	-	-

	MAPLE	0.16	0.37	0.71	0.19	-0.05	-0.01	-	-
	S-LIME	0.213	0.69	0.75	-0.04	-0.10	0.38	-	-
	SHAP	0.31	0.64	0.80	-0.06	-0.05	0.38	-	-
	DICE (KDTREE)	0.35	0.54	0.47	-0.11	0.17	0.11	2.27	1.0
	DICE (RANDOM)	0.49	0.57	0.65	-0.12	0.20	0.33	17.51	0.0
	DICE (GENETIC)	0.35	0.52	0.46	-0.07	0.16	0.10	3.17	1.0
	FGM	0.01	0.06	0.38	-0.26	-0.20	-0.07	7.25	1.0
	PGD	0.00	0.06	0.36	-0.28	-0.22	-0.08	7.25	0.06
	GSCF	0.17	0.27	0.54	-0.07	0.00	-0.03	1.64	0.0

Table 7. Faithfulness Scores

EXPLORE achieves a perfect faithfulness of 1 which is a consequence of the following proof: Let $x_i \in X$ be a test instance. Based on the exact reconstruction of all DBs of x_i, all vectors to any DB and CS (and thus all $VFAs$ and $SFAs$) can be correctly determined. For any ML model we can select the VFA_{DB} that minimizes the distance to any DB as the feature attribution vector of EXPLORE, since the minimum distance to a closed subset is unique. As discussed for the multi-class logistic regression, the ground truth explanation agrees with the vector towards the nearest DB point. Thus, since both the ground truth vector and the selected VFA_{DB} attain the nearest DB point, the vectors and their entries fully coincide. Therefore, a perfect top-K feature agreement and top-K pairwise rank agreement is attained by EXPLORE for any test instance x_i.

7 Discussion and Implications

In this paper, we proposed a novel local XAI method and new variants of local explanations. Based on these two contributions, we now discuss the main results and implications.

7.1 Representing DBs and CSs of an ML Model

First, we introduced EXPLORE, a method to reconstruct the DBs and the CSs of an ML model in a functional and definite manner, thus making the ML model transparent. We show that our method not only matches the ML model with very high accuracy or even perfectly (in the case of a PL model) in the local neighborhood of any instance to be explained but is also robust and reproducible upon repetition. In the evaluation, we substantiate these results with empirical evidence by measuring the fidelity and robustness of EXPLORE and comparing it to existing local XAI methods on five real-world datasets. Table 5 shows that EXPLORE clearly outperforms existing methods regarding fidelity, even in challenging settings such as the MIMIC and the COMPAS dataset. Further, our analysis shows that EXPLORE provides perfect robustness for identical or similar instances, which is substantiated by our robustness score of 1 across all datasets and model types. These theoretical and empirical results support the first contribution of our paper corresponding to our Proposition 1. By functionally reconstructing the DBs and CSs of an ML model in the neighborhood of an instance, we are able to represent the model behavior in that neighborhood. Essentially, this addresses the key task of local XAI methods to make the ML model around an instance transparent: Existing XAI methods that rely on perturbations, random sampling, and surrogate models often struggle to accurately capture the multiple DBs of an ML model even locally, especially in high-dimensional spaces. This becomes evident in classification tasks with a higher number of classes such as the MIMIC or Traffic dataset, where methods such

as ROLEX or LIME achieve significantly lower fidelity than on the Fetal Health dataset. Thereby, existing local XAI methods often explain an instance focusing on one DB, potentially ignoring other relevant DBs in the instance's neighborhood. In the case of the instance from the Fetal Health dataset (see Figure 1), this can have severe consequences as several DBs to Class SUSPECT and even Class PATHOLOGICAL are in fact completely disregarded.

Guidotti et al. (2019) emphasize high fidelity as a main desideratum for XAI methods, as otherwise the provided explanation and transparency cannot be correctly transferred to the black-box model. Literature has recognized the utmost importance of explainability and transparency of ML models in high-stakes domains. For instance, Kronblad et al. (2024, p. 1655) highlight explainability and transparency as important guiding principles for preventing, counteracting, and remedying detrimental consequences of algorithmic decision making, that is "fundamental to enable critical scrutiny by any party". Therefore, making the inner workings and predictive behavior of an ML model correctly and fully transparent to users is a key requirement for an ethical and fair use of ML, ensuring legal and social justice in domains such as the public sector where algorithms can exacerbate social inequalities, solidify historical biases, or reinforce discrimination (Kronblad et al., 2024). In this context, transparency also enables human intervention when problematic predictive behavior of an ML model is detected. Further, e.g., in model-based credit scoring, the transparency of the employed ML models can support verifying their algorithmic fairness by revealing the impact of protected features such as gender, age, or racial origin (Hurlin et al., 2024). EXPLORE addresses this transparency issue by functionally representing black-box models without relying on potentially distorting heuristics that could undermine the user trust in the XAI method (Guo et al., 2025; Petch et al., 2022).

The same holds when an XAI method exhibits limited robustness and reproducibility, i.e., when its results vary significantly for the same or very similar instances. Given the growing importance of XAI in fostering trust in high-stakes ML-augmented decision making, literature has recognized that "generating robust and reliable explanations […] can be a nontrivial issue" (Kim et al., 2023, p. 1325) and that the robustness of explanations poses a major challenge that needs to be addressed by further development and research (Panigutti et al., 2023). Indeed, this issue is of high relevance as a lack of robustness and reproducibility becomes evident in high-stakes applications such as credit scoring (Visani et al., 2022) or health (Saarela & Geogieva, 2022). Especially in cases where this lack of reproducibility leads to different explanations for the same data instance it is difficult for the user to know which explanation to rely on (Panigutti et al., 2023). For instance, in the case of skin cancer classification, several conflicting explanations also lead to legal questions that need to be resolved before an XAI method can be employed in clinical practice (Saarela & Geogieva, 2022). Besides hampering informed decision making, this can also heavily undermine the user trust in the provided explanations themselves, at worst leading to complete rejection of both the ML model and the XAI method. In contrast, EXPLORE reconstructs all DBs and CSs in an instance's neighborhood (of arbitrary size) in a functional, deterministic, and thus reproducible manner, hence addressing the criterion robustness.

7.2 Providing Local Explanations

Second, we propose new variants of local explanations regarding Proposition 2. These local explanations contribute to both key questions in XAI: why an instance was classified by the ML model in the way it was (abductive explanations), and why it was not assigned to a different class or what feature value changes are necessary and sufficient for the instance to be assigned to a different class (contrastive / CF explanations). In our evaluation, we assess the faithfulness of these local explanations on different real-world datasets. Table 7 shows that EXPLORE clearly outperforms existing XAI methods in terms of top-K agreement and ranking, even in challenging settings such as the Prosper and MIMIC datasets. For example, for the Prosper dataset, no existing XAI method exceeds a top-5 agreement score of over 0.6, indicating that on average the methods provide at least two improper features in their top-5 explanations. Regarding feature ranking, while in some settings a stronger correlation of over 0.49 (cf. Schober et al., 2018) is achieved by some XAI methods, most rank scores—even for the more optimistic top-K version of Kendall's Tau—remain moderate (between 0.26 and 0.49) or even weak (lower than 0.26).

On the one hand, our VFAs allow for a direct and intuitive interpretation of which concrete (modified) feature values (vector) lead to a point on a DB or within a CS. On the other hand, we introduced the SFAs as a comprehensive feature attribution—similar to the "aggregated" explanations provided by, e.g., SHAP—that can be analyzed by user-specific and context-dependent operators to reveal the course of DBs or the depth and volume of CSs. We provide formal proofs to demonstrate that both VFAs and SFAs can be accurately determined, ensuring the faithfulness of the local explanations provided by EXPLORE. This inherently resolves general concerns related to feature attributions that are discussed in literature, such as features that are assigned a disproportionately large or low value compared with their actual relevance for the prediction, or their insufficiency to communicate their true influence for the prediction (Fernández-Loría et al., 2022; Huang & Marques-Silva, 2024). Furthermore, Fernández-Loría et al. (2022, p. 1645) argue that "it is difficult to capture the impact of features on decisions with a single number, especially when features interact with each other", and suggest using CF examples to remedy this problem. By introducing the SFAs we pursue an even more general and flexible approach: Since an SFA defines a set that contains any VFA vector directing to a point on a DB or CS, a user can focus on different properties of the SFA by employing suitable operators (e.g., minimum, maximum, average,…), thus analyzing the extent or direction of a DB, or specific feature interactions that lead to preserving or changing a class. A CF operator on an SFA is also feasible by defining the needed properties, such as the maximum allowed distance of a CF to the instance to be explained, the minimum or maximal distance to a DB or the proximity to existing ground truth data instances. In this way, the notion of VFAs and SFAs can naturally combine both explanation types, feature attributions and CFs.

In general, enabling the faithfulness of local abductive and contrastive explanations is of paramount importance, as XAI methods that fail to identify the truly relevant features and their accurate attribution for an explanation can potentially cause substantial harm. Particularly in high-

stakes domains such as healthcare, explanations can pose a valuable tool to enhance patients' trust in ML models (see Kim et al., 2023 for fragility fracture prediction). However, Kim et al. (2023, p. 1321) also point out that such explanations sometimes contradict domain knowledge, e.g., "an inverse association between body fat and fracture risks has been reported […]. However, our analysis shows a positive relationship between the two". In many cases, the heuristic nature of perturbations, random sampling and surrogate models in high-dimensional spaces can lead to such results that should thus be treated with great caution. Here, local explanations provided by EXPLORE can support medical decision making to prevent both serious medical consequences for the patient and ethical issues, especially if previous explanations provided by XAI methods potentially do not correctly reflect the model. It is crucial to emphasize that the choice of the needed local explanations strongly depends on the specific application, the type of users, and their transparency and analysis objectives. Such choices must be carefully explored when designing workflows and visualizations for user interaction. We showcase such a scenario and thereby demonstrate the application of selected feature attributions for a real-world loan application in detail, utilizing the Prosper dataset in our Supplementary Material.

8 Limitations and Future Research

Our work also has limitations that provide a starting point for future research. To begin with, we instantiated our method for a group of widely adapted ML model types in this paper. However, our method can also be applied to other model types or NN architectures such as CNNs (e.g., ResNet), opening a variety of exciting and verifiable options for XAI on image data. Although the PL structure of CNNs can be functionally well determined using EXPLORE, the computational complexity—particularly for deep networks—remains a challenge, which we are currently addressing. Further, we presented a selection of SFA operators each of which focuses on particular properties of EXPLORE's explanations and enables further user- and context specific analyses on this basis. Future research could focus on conceptualizing and defining additional SFA operators for challenges in XAI and beyond. For the identification of CF examples, they can be optimized to return only such CFs that satisfy predefined desired properties. Finally, we only showcased some possible user interactions that leverage the transparency and explanations provided by EXPLORE. Building on these new possibilities, future research has to focus on designing structured workflows that enable users to systematically explore DBs and CSs of an ML model. Also, user-centric studies examining the understandability and interpretability of our local explanations have to be conducted. Crucially, as XAI research still lacks a clear consensus on what makes an explanation useful, the focus should be on the purpose of the explanation in the given context, as an inappropriate explanation can potentially be more harmful than an opaque model prediction. Such studies could further develop the alignment of our method with user needs and preferences, facilitating effective interaction and collaboration between users and AI.

9 References

Agarwal, C., Krishna, S., Saxena, E., Pawelczyk, M., Johnson, N., Puri, I., Zitnik, M., & Lakkaraju, H. (2022). OpenXAI: Towards a transparent evaluation of model explanations. In *Advances in Neural Information Processing Systems*.

Alvarez-Melis, D., & Jaakkola, T. (2018a). On the robustness of interpretability methods. In *Proc. ICML Workshop on Human Interpretability in Machine Learning*.

Alvarez-Melis, D., & Jaakkola, T. (2018b). Towards robust interpretability with self-explaining neural networks. In *Advances in Neural Information Processing Systems*.

Baehrens, D., Schroeter, T., Harmeling, S., Kawanabe, M., Hansen, K., & Müller, K.-R. (2010). How to explain individual classification decisions. *Journal of Machine Learning Research*, *11*, 1803–1831.

Bauer, K., & Gill, A. (2024). Mirror, mirror on the wall: Algorithmic assessments, transparency, and self-fulfilling prophecies. *Information Systems Research*, *35*(1), 226–248.

Bayrak, B., & Bach, K. (2024). Evaluation of instance-based explanations: An in-depth analysis of counterfactual evaluation metrics, challenges, and the CEval toolkit. *IEEE Access*, *12*, 137683–137695.

Berente, N., Gu, B., Recker, J., & Radhika, S. (2021). Managing artificial intelligence. *MIS Quarterly*, *45*(3), 1433–1450.

Bishop, C. M. (2006). *Pattern recognition and machine learning*. Springer.

Booth, R. (2024). *Revealed: Bias found in AI system used to detect UK benefits fraud*. https://www.theguardian.com/society/2024/dec/06/revealed-bias-found-in-ai-system-used-to-detect-uk-benefits

Brasse, J., Broder, H. R., Förster, M., Klier, M., & Sigler, I. (2023). Explainable artificial intelligence in information systems: A review of the status quo and future research directions. *Electronic Markets*, *33*(1), 1–30.

Brughmans, D., Leyman, P., & Martens, D. (2024). NICE: An algorithm for nearest instance counterfactual explanations. *Data Mining and Knowledge Discovery*, *38*(5), 2665–2703.

Chiluveru, S., Tripathy, M., & Bibhudutta (2021). Non-linear activation function approximation using a REMEZ algorithm. *IET Circuits, Devices & Systems*, *15*(7), 630–640.

Chu, L., Hu, X., Hu, J., Wang, L., & Pei, J. (2018). Exact and consistent interpretation for piecewise linear neural networks. In *Proc. 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Dombrowski, A.-K., Alber, M., Anders, C., Ackermann, M., Müller, K.-R., & Kessel, P. (2019). Explanations can be manipulated and geometry is to blame. In *Advances in Neural Information Processing Systems*.

EU AI Act. (2025). *The EU Artificial Intelligence Act*. https://artificialintelligenceact.eu/

Fernández-Loría, C., Provost, F., & Han, X. (2022). Explaining data-driven decisions made by AI systems: The counterfactual approach. *MIS Quarterly*, *45*(3), 1635–1660.

Fu, R., Huang, Y., & Singh, P. V. (2021). Crowds, lending, machine, and bias. *Information Systems Research*, *32*(1), 72–92.

Goodfellow, I., Courville, A., & Bengio, Y. (2016). *Deep learning*. MIT Press.

Goodfellow, I., Shlens, J., & Szegedy, C. (2015). Explaining and harnessing adversarial examples. In *Proc. 3rd International Conference on Learning Representations*.

Guidotti, R., Monreale, A., Ruggieri, S., Pedreschi, D., Turini, F., & Giannotti, F. (2018). *Local rule-based explanations of black box decision systems*. arXiv.

Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., & Pedreschi, D. (2019). A survey of methods for explaining black box models. *ACM Computing Surveys*, *51*, 1–42.

Guo, T., Bardhan, I. R., Ding, Y., & Zhang, S. (2025). An explainable artificial intelligence approach using graph learning to predict intensive care unit length of stay. *Information Systems Research*, *36*(3), 1478–1501.

Harutyunyan, H., Khachatrian, H., Kale, D. C., Ver Steeg, G., & Galstyan, A. (2019). Multitask learning and benchmarking with clinical time series data. *Scientific Data*, *6*(1), Article 96.

Herm, L.-V., Heinrich, K., Wanner, J., & Janiesch, C. (2023). Stop ordering machine learning algorithms by their explainability! A user-centered investigation of performance and explainability. *International Journal of Information Management*, *69*, 102538.

Hinton, G., Vinyals, O., & Dean, J. (2015). *Distilling the knowledge in a neural network*. arXiv.

Huang, X., & Marques-Silva, J. (2024). On the failings of Shapley values for explainability. *International Journal of Approximate Reasoning*, *171*, 109112.

Hurlin, C., Pérignon, C., & Saurin, S. (2024). The fairness of credit scoring models. *Management Science*.

Jia, Y., Bailey, J., Ramamohanarao, K., Leckie, C., & Houle, M. E. (2019). Improving the quality of explanations with local embedding perturbations. In *Proc. 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Kim, B., Srinivasan, K., Kong, S. H., Kim, J. H., Shin, C. S., & Ram, S. (2023). ROLEX: A novel method for interpretable machine learning using robust local explanations. *MIS Quarterly*, *47*(3), 1303–1332.

Kim, S., Park, Y., Lee, H., Yi, S., & Lee, J. W. (2024). Reluifying smooth functions: Low-cost knowledge distillation to obtain high-performance ReLU networks. In *Proc. Asian Conference on Computer Vision* (pp. 2162–2178).

Krapf, T., Hagn, M., Miethaner, P., Schiller, A., Luttner, L., & Heinrich, B. (2024). Piecewise linear transformation – Propagating aleatoric uncertainty in neural networks. In *Proc. 38th AAAI Conference on Artificial Intelligence*.

Kronblad, C., Essén, A., & Mähring, M. (2024). When justice is blind to algorithms: Multi-layered blackboxing of algorithmic decision making in the public sector. *MIS Quarterly*, *48*(4), 1637–1662.

Lakkaraju, H., Arsov, N., & Bastani, O. (2020). Robust and stable black box explanations. In *Proc. 37th International Conference on Machine Learning*.

Laugel, T., Lesot, M., Marsala, C., Renard, X., & Detyniecki, M. (2018a). Comparison-based inverse classification for interpretability in machine learning. In *Information Processing and Management of Uncertainty in Knowledge-Based Systems. Theory and Foundations*.

Laugel, T., Renard, X., Lesot, M., Marsala, C., & Detyniecki, M. (2018b). Defining locality for surrogates in post-hoc interpretablity. In *Proc. ICML Workshop on Human Interpretability in Machine Learning*.

Li, C., Wang, H., Jiang, S., & Gu, B. (2024). The effect of AI-enabled credit scoring on financial inclusion: Evidence from an underserved population of over one million. *MIS Quarterly*, *48*(4), 1803–1834.

Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., & Vladu, A. (2018). Towards deep learning models resistant to adversarial attacks. In *Proc. 6th International Conference on Learning Representations*.

Mehbodniya, A., Lazar, A. J. P., Webber, J., Sharma, D. K., Jayagopalan, S., K, K., Singh, P., Rajan, R., Pandya, S., & Sengan, S. (2022). Fetal health classification from cardiotocographic data using machine learning. *Expert Systems*, *39*(6).

Mersha, M., Lam, K., Wood, J., AlShami, A. K., & Kalita, J. (2024). Explainable artificial intelligence: A survey of needs, techniques, applications, and future direction. *Neurocomputing*, *599*, 128111.

Miller, T. (2019). Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, *267*, 1–38.

Moreira, C., Chou, Y.-L., Hsieh, C., Ouyang, C., Pereira, J., & Jorge, J. (2025). Benchmarking instance-centric counterfactual algorithms for XAI: From white box to black box. *ACM Computing Surveys*, *57*(6), 1–37.

Mothilal, R. K., Sharma, A., & Tan, C. (2020). Explaining machine learning classifiers through diverse counterfactual explanations. In *Proc. 2020 Conference on Fairness, Accountability, and Transparency*.

Nauta, M., Trienes, J., Pathak, S., Nguyen, E., Peters, M., Schmitt, Y., Schlötterer, J., van Keulen, M., & Seifert, C. (2023). From anecdotal evidence to quantitative evaluation methods: A systematic review on evaluating explainable AI. *ACM Computing Surveys*, *55*(13s), 1–42.

Oliveira, R. M. B. de, & Martens, D. (2021). A framework and benchmarking study for counterfactual generating methods on tabular data. *Applied Sciences*, *11*(16).

Panigutti, C., Hamon, R., Hupont, I., Fernandez Llorca, D., Fano Yela, D., Junklewitz, H., Scalzo, S., Mazzini, G., Sanchez, I., Soler Garrido, J., & Gomez, E. (2023). The role of explainable AI in the context of the AI Act. In *Proc. 2023 ACM Conference on Fairness, Accountability, and Transparency*.

Pawlicki, M., Pawlicka, A., Uccello, F., Szelest, S., D'Antonio, S., Kozik, R., & Choraś, M. (2024). Evaluating the necessity of the multiple metrics for assessing explainable AI: A critical examination. *Neurocomputing*, *602*, 128282.

Petch, J., Di, S., & Nelson, W. (2022). Opening the black box: The promise and limitations of explainable machine learning in cardiology. *The Canadian Journal of Cardiology*, *38*(2), 204–213.

Petsiuk, V., Das, A., & Saenko, K. (2018). *RISE: Randomized input sampling for explanation of black-box models*. arXiv.

Plumb, G., Molitor, D., & Talwalkar, A. S. (2018). Model agnostic supervised local explanations. In *Advances in Neural Information Processing Systems*.

Poyiadzi, R., Sokol, K., Santos-Rodriguez, R., Bie, T. de, & Flach, P. (2020). FACE: Feasible and actionable counterfactual explanations. In *Proc. AAAI/ACM Conference on AI, Ethics, and Society*.

Reddy, P., & Gujral, A. S. (2025). Improving neural network efficiency using piecewise linear approximation of activation functions. In *Proc. FLAIRS-38*.

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why should I trust you?" Explaining the predictions of any classifier. In *Proc. 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Ribeiro, M. T., Singh, S., & Guestrin, C. (2018). Anchors: High-precision model-agnostic explanations. In *Proc. 32nd AAAI Conference on Artificial Intelligence*.

Saarela, M., & Geogieva, L. (2022). Robustness, stability, and fidelity of explanations for a deep skin cancer classification model. *Applied Sciences*, *12*(19), 9545.

Salih, A. M., Raisi-Estabragh, Z., Galazzo, I. B., Radeva, P., Petersen, S. E., Lekadir, K., & Menegaz, G. (2025). A perspective on explainable artificial intelligence methods: SHAP and LIME. *Advanced Intelligent Systems*, *7*(1), 2400304.

Schober, P., Boer, C., & Schwarte, L. A. (2018). Correlation coefficients: Appropriate use and interpretation. *Anesthesia and Analgesia*, *126*(5), 1763–1768.

Sundararajan, M., Taly, A., & Yan, Q. (2017). Axiomatic attribution for deep networks. In *Proc. 34th International Conference on Machine Learning*.

Unicef. (2024). *Looking ahead: Child survival and the Sustainable Development Goals*. https://data.unicef.org/topic/child-survival/child-survival-sdgs/

van Looveren, A., & Klaise, J. (2021). Interpretable counterfactual explanations guided by prototypes. In *Machine Learning and Knowledge Discovery in Databases*.

Velmurugan, M., Ouyang, C., Sindhgatta, R., & Moreira, C. (2025). Through the looking glass: evaluating post hoc explanations using transparent models. *International Journal of Data Science and Analytics*, *20*(2), 615–635.

Visani, G., Bagli, E., Chesani, F., Poluzzi, A., & Capuzzo, D. (2022). Statistical stability indices for LIME: Obtaining reliable explanations for machine learning models. *Journal of the Operational Research Society*, *73*(1), 91–101.

Wachter, S., Mittelstadt, B., & Russell, C. (2018). Counterfactual explanations without opening the black box: Automated decisions and the GDPR. *Harvard Journal of Law and Technology*, *31*(2), 841–887.

Williamson, B., & Feng, J. (2020). Efficient nonparametric statistical inference on population feature importance using Shapley values. In *Proc. 37th International Conference on Machine Learning*.

Yalcin, O., Fan, X., & Liu, S. (2021). *Evaluating the correctness of explainable AI algorithms for classification*. arXiv.

Zhou, Z., Hooker, G., & Wang, F. (2021). S-LIME: Stabilized-LIME for model explanation. In *Proc. 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.

10 Supplementary Material

10.1 Supplement A: EXPLORE Method

In this section of our Supplementary Material, we provide proofs and additional technical details on selected contents from Section 5 on the EXPLORE method.

10.1.1 Supplement A.1: Background Information on Explore Instantiation for Neural Networks

Definition 2 (Output Space DBs). If a point $y \in Y$ in the output space has two equal entries that are also its largest entries (i.e., it satisfies $y_i - y_j = 0$ for $i \neq j$ and $y_i, y_j \geq y_k$ and all $k \neq i, j$ with y_i denoting the i-th vector entry of $y \in Y \subset \mathbb{R}^n$), then y lies on an output space DB. Therefore, the DBs between two (different) classes i, j in the output space Y are given by

$$db_{i,j}{}^{o} = \{y \in \mathrm{Y} \mid y_i - y_j = 0 \ \wedge \forall k \neq i, j \colon y_i \geq y_k\} \tag{A1}$$

for all $1 \leq i \neq j \leq n$. By this definition, the symmetry $db_{i,j}{}^{o} = db_{j,i}{}^{o}$ holds.

Proof. In NNs used for classification (with at least two classes, i.e., $n \geq 2$), the prediction for an instance with output value y is determined by selecting the class with the highest corresponding vector entry of y. Therefore, the DB between the classes i and j in the output space lies where their corresponding entries are equal, i.e., where $y_i - y_j = 0$ holds. However, this equation only defines a DB where no other class k attains a higher value than y_i and y_j, i.e., where $y_i, y_j \geq y_k, \forall k \neq i, j$ holds. Thus, the output space DB between the classes i and j is given by Term (A1) and can also be described by the system of (in)equalities

$$\begin{aligned} y_i - y_j &= 0, \\ y_k - y_i &\leq 0 \; for \; k \neq i, j. \end{aligned} \tag{A2}$$

□

Lemma 1 (Feature Space DBs). Let again $X \subset \mathbb{R}^m, Y \subset \mathbb{R}^n$ and $f \colon X \to \mathrm{Y}$ be the function representing an NN. Moreover, let A be a linear region of f (as in Definition 1), and $db_{i,j}{}^{o}$ be a DB in the output space between the (different) classes i and j. Then, the corresponding DB in the linear region A in the feature space can be explicitly calculated. Utilizing the affine linear mapping T, c representing f on A, the DB is characterized by the system of (in)equalities

$$\begin{aligned} v_{i,j} \cdot T \cdot x + v_{i,j} \cdot c &= 0, \\ v_{k,i} \cdot T \cdot x + v_{k,i} \cdot c &\leq 0 \end{aligned} \tag{A3}$$

for $x \in A$. In this term, $v_{i,j} \in \mathbb{R}^n$ is defined as the vector with 1 in the i-th entry, -1 in the j-th entry and 0 otherwise.

Proof. According to Definition 1, $f(x) = Tx + c$ holds for $x \in A$ and according to Term (3), the output space DB $db_{i,j}{}^{o}$ can be written by $v_{i,j} \cdot y = 0, v_{k,i} \cdot y \leq 0$ for $y \in Y, k \neq i, j$. For $y = f(x) = T \cdot x + c$, this system of (in)equalities can directly be rewritten as in Term (A3). □

Definition 12 (Feature Sub Spaces as Neighborhoods). For $x \in X$ in a neighborhood Ω_x, any subset of features $D \subseteq \{1, \dots, m\}$ and any set of constraints on these features $R = \{(r_{d,-}, r_{d,+}) \in \mathbb{R}^2 \mid d \in D\}$, all introduced feature attribution values (VFA and SFA for DBs and CSs) can be defined and accurately determined on the constrained neighborhood $\Omega_{x,R} \coloneqq \{x' \in \Omega_x \mid \forall d \in \{1, \dots, m\} \backslash D \colon x'_d = x_d \ \wedge \forall d \in D \colon x'_d \in (r_{d,-}, r_{d,+})\}$.

10.1.2 Supplement A.2: Background Information for Defining Feasible Operators on Feature Attributions

Lemma (Operators on SFA_{DB}). For $x \in X$ with $f(x) = i \neq j$, $\Omega_x \subset X$ be a neighborhood of x, and $c > 0$. Then, the following statements hold:

1) **Contrastive Minimum:** If the absolute value of the k-th entry is larger than c, i.e., $|vfa_k| > c$ for all vectors in $vfa \in SFA^j_{DB,x}$ then an absolute value change of (at least) c is needed for the assignment to class j, i.e., $f(x') = j$ for $x' \in \Omega_x \Rightarrow |x'_k - x_k| > c$ holds.
2) **Abductive Minimum:** If $|vfa_k| > c$ holds for all vectors $vfa \in SFA^i_{DB,x}$, then for any $x' \in \Omega_x$ with $|x'_k - x_k| < c$ the class assignment $f(x') = i$ is retained.
3) **Contrastive Maximum:** If the absolute value of the k-th entry is smaller than c, i.e., $|vfa_k| < c$ for all vectors in $vfa \in SFA^j_{DB,x}$ then no absolute value change larger than c is necessary to the assignment to class j, i.e. for any $x' \in \Omega_x$, such that $f(x') = j$ and $|x'_k - x_k| > c$, there exists $0 < \lambda < 1$, s.t. $f(\lambda x + (1-\lambda)x') = j$ and $|\lambda x_k{}' + (1-\lambda)x_k| \leq c$.
4) **Further operators** include the variation, average, mean or standard deviation of the absolute values in the $SFA_{DB,x}$. In particular, operators can be constructed to fit the needs of specific applications, one example being the selection of one VFA from the SFA_{DB}, satisfying any application specific properties.
5) **Integration:** One complex and interesting operator for the aggregation of a $SFA_{DB,x}$ is given by integration, which is called cumulative feature attribution and defined as follows: We define the contrastive CFA_{DB} (with respect to class j) of an instance x and the abductive CFA_{DB} of instance x as

$$CFA^{C,j}_{DB}(x) = \int_{S'_{db,j} \cap \Omega'_x} |VFA^j_{DB,x}(x_{db})| \, \mathrm{d}x_{db},$$

$$CFA^{A,f(x_{db})}_{DB}(x) = \int_{(S'_{db} \cup \partial\Omega) \cap \Omega'_x} |VFA^{f(x_{db})}_{DB,x}(x_{db})| \, \mathrm{d}x_{db},$$

where $|\cdot|$ denotes taking absolute values in each entry of the vector.

Proof. 1) For a proof by contradiction, we assume the existence of a point $x' \in \Omega_x$ with $f(x') = j$ such that $|x'_k - x_k| \leq c$ holds. Since $f(x) = i$, the connection $\lambda x + (1-\lambda)x'$ contains a point $x_{db} \in S'_{db,j}$ on a DB towards class j with $x_{db} = \lambda x + (1-\lambda)x'$ for a $\lambda \in [0,1]$. The vector $\overrightarrow{d_x}(x_{db}) = x_{db} - x$ then is contained in $SFA^j_{DB,x}$, and $|\overrightarrow{d_x}(x_{db})_k| = |x_{db,k} - x_k| = |\lambda x_k + (1-\lambda)x'_k - x_k| = (1-\lambda)|x'_k - x_k| \leq c$ holds, contradicting $|vfa_k| > c$ for all $vfa \in SFA^j_{DB,x}$.

2) We assume the existence of a point $x' \in \Omega_x$ with $|x'_k - x_k| \leq c$ such that $f(x') = j$ for a class $j \neq i$. Then, $\lambda x + (1-\lambda)x'$ contains a point $x_{db} \in S'_{db,i}$ on a DB towards class i with $x_{db} = \lambda x + (1-\lambda)x'$ for a $\lambda \in [0,1]$. Again, $\overrightarrow{d_x}(x_{db}) \in SFA^i_{DB,x}$ and $|\overrightarrow{d_x}(x_{db})_k| = |x_{db,k} - x_k| =$

$|\lambda x_k + (1-\lambda)x'_k - x_k| = (1-\lambda)|x'_k - x_k| \leq c$ contradicts $|vfa_k| > c$ for all $vfa \in SFA^i_{DB,x}$.

3) Let $x' \in \Omega_x$, such that $f(x') = j$ and $|x'_k - x_k| > c$. Then $\lambda' x + (1-\lambda')x'$ contains a point $x_{db} \in S'_{db,j}$ on a DB towards class i with $x_{db} = \lambda' x + (1-\lambda')x'$ for a $\lambda' \in [0,1]$. Since $|vfa_k| < c$ holds for all vectors $vfa \in SFA^j_{DB,x}$, $(1-\lambda')|x'_k - x_k| = |\lambda' x_k + (1-\lambda')x'_k - x_k| = |x_{db,k} - x_k| = |\overrightarrow{d_x}(x_{db})_k| < c$ follows. From $|x'_k - x_k| > c$, we obtain a $0 < \lambda < \lambda'$ such that $(1-\lambda)|x'_k - x_k| = c$. Then, $|\lambda x_k + (1-\lambda)x'_k| \leq c$ and $f(\lambda x + (1-\lambda)x') = j$ holds since $\lambda x + (1-\lambda)x'$ lies between the DB and x'. Note that all values $[\lambda, \lambda')$ are also feasible.

This concludes the proof of the Lemma, as for 4) and 5) there is nothing to show. □

All results in this Lemma can be calculated without taking the absolute value (interpretation changes accordingly). While we exemplified some operators for the SFA_{DB} here, the same operators lead to corresponding interpretations on the SFA_{CS}. Note, that while these operators refer to PL structures given by constraining equations, the use of some operators like integration will in general still require the use of numerical approximation techniques.

10.2 Online Supplement B: Evaluation

This section of our Supplemental Material includes additional background information and technical details on our evaluation. We provide information on the dataset provenance, the training and hyperparameter tuning for the models used in our evaluation, our instantiation of the approximation strategies for non-PL models, theoretical bounds of the approximation error, and formal definitions of the explanations by EXPLORE in our fidelity, robustness, and faithfulness analyses.

10.2.1 Supplement B.1: Dataset Provenance

Background information and details for access for the datasets used in the evaluation can be found under the URLs listed in Table 8.

Dataset	URL
Fetal Health	https://www.kaggle.com/datasets/andrewmvd/fetal-health-classification
Prosper	https://www.kaggle.com/datasets/henryokam/prosper-loan-data
COMPAS	https://www.kaggle.com/datasets/danofer/compass
HSIS (traffic):	https://highways.dot.gov/media/32406
MIMIC	https://physionet.org/content/mimiciii/1.4/

Table 8. Dataset Provenance

10.2.2 Supplement B.2: Background Information on Hyperparameter Tuning for ML Models

We performed a fivefold cross-validation to find optimal hyperparameters for all ML model types for each dataset. The optimal hyperparameter configuration found for each classifier was determined with respect to the average balanced accuracy. For NN models, we use two hidden layers

each with ten neurons with ReLU activation function, and two hidden layers with five neurons with sigmoid and tanh activation functions respectively.

The **maximum depth** (MaxDepth) of the tree is applicable to tree-based algorithms, i.e., DT and RF. Possible values are {10, 20, 30}.

The **number of estimators** (#Estimators) is the maximum number of estimators in ensemble-based algorithms, i.e., RF and AdaBoost. Possible values are {10, 25, 50, 100}.

The **learning rate** (LR) for AdaBoost is the weight applied to the classifier at every iteration, whereas for NN, it refers to the step size when updating the model parameters during training. Possible values are {1, 0.1, 0.01, 0.001}.

The **number of epochs**: (#Epochs) is the maximum number of times each data instance is used to update the NN model parameters during training. Possible values are {100, 500, 1,000}.

The **regularization parameter**: (RegParam): Regularization parameter controlling the ratio of the loss function and a penalty term when fitting a SVM model. Possible values are {1, 5, 10}.

The **degree of polynomial** (PolyDeg) of the polynomial used for the kernel when fitting a SVM model with polynomial kernel. Possible values are {1, 2, 3, 4}.

Dataset	Model Type	Max Depth	#Esti-mators	LR	#Epochs	Reg Param	Poly Deg	BalancedAcc/ Accuracy	Balanced F1/ Weighted F1
Fetal Health	NN	-	-	0.01	500	-	-	0.822/0.84	0.828/0.84
	DT	10	-	-	-	-	-	0.897/0.894	0.896/0.894
	RF	20	100	-	-	-	-	0.912/0.914	0.914/0.913
	SVM	-	-	-	-	5	-	0.852/0.863	0.854/0.863
	AdaBoost	-	100	0.1	-	-	-	0.859/0.869	0.87/0.869
	NN (sigmoid)	-	-	0.01	1,000	-	-	0.799/0.834	0.812/0.830
	NN (tanh)	-	-	0.01	1,000	-	-	0.833/0.854	0.843/0.853
	SVM (poly)	-	-	-	-	10	4	0.881/0.878	0.871/0.879
	SVM (RBF)	-	-	-	-	10	-	0.883/0.878	0.871/0.879
Prosper	NN	-	-	0.001	1,000	-	-	0.803/0.779	0.791/0.774
	DT	10	-	-	-	-	-	0.807/0.777	0.795/0.769
	RF	30	100	-	-	-	-	0.842/0.811	0.823/0.799
	SVM	-	-	-	-	10	-	0.666/0.665	0.656/0.659
	AdaBoost	-	100	0.1	-	-	-	0.655/0.61	0.618/0.574
	NN (sigmoid)	-	-	0.01	500	-	-	0.765/0.792	0.756/0.778
	NN (tanh)	-	-	0.001	1,000	-	-	0.798/0.775	0.781/0.766
	SVM	-	-	-	-	10	2	0.708/0.703	0.699/0.726

	(poly)								
	SVM (RBF)	-	-	-	-	10	-	0.719/0.721	0.715/0.738
COMPAS	NN	-	-	0.001	1,000	-	-	0.472/0.503	0.465/0.487
	DT	30	-	-	-	-	-	0.714/0.707	0.715/0.708
	RF	30	50	-	-	-	-	0.74/0.736	0.742/0.736
	SVM	-	-	-	-	5	-	0.39/0.426	0.368/0.392
	AdaBoost	-	100	1	-	-	-	0.427/0.462	0.416/0.441
	NN (sigmoid)	-	-	0.01	500	-	-	0.468/0.498	0.461/0.483
	NN (tanh)	-	-	0.01	1,000	-	-	0.458/0.487	0.455/0.473
	SVM (poly)	-	-	-	-	10	2	0.459/0.420	0.362/0.441
	SVM (RBF)	-	-	-	-	10	-	0.400/0.411	0.340/0.440
HSIS	NN	-	-	0.001	1,000	-	-	0.333/0.354	0.324/0.338
	DT	10	-	-	-	-	-	0.312/0.333	0.307/0.322
	RF	30	100	-	-	-	-	0.347/0.369	0.344/0.359
	SVM	-	-	-	-	5	-	0.332/0.351	0.31/0.318
	AdaBoost	-	25	1	-	-	-	0.327/0.349	0.317/0.333
	NN (sigmoid)	-	-	0.001	1,000	-	-	0.336/0.353	0.324/0.334
	NN (tanh)	-	-	0.001	1,000	-	-	0.329/0.348	0.32/0.332
	SVM (poly)	-	-	-	-	10	2	0.334/0.323	0.286/0.356
	SVM (RBF)	-	-	-	-	10	-	0.331/0.314	0.277/0.348
MIMIC	NN	-	-	0.001	1,000	-	-	0.377/0.431	0.312/0.356
	DT	10	-	-	-	-	-	0.357/0.379	0.355/0.373
	RF	10	50	-	-	-	-	0.386/0.433	0.36/0.398
	SVM	-	-	-	-	10	-	0.37/0.423	0.314/0.358
	AdaBoost	-	100	0.1	-	-	-	0.386/0.441	0.342/0.391
	NN (sigmoid)	-	-	0.01	1,000	-	-	0.38/0.433	0.322/0.366
	NN (tanh)	-	-	0.001	1,000	-	-	0.379/0.432	0.329/0.375
	SVM (poly)	-	-	-	-	10	4	0.438/0.388	0.258/0.481

	SVM (RBF)	-	-	-	-	10	-	0.350/0.409	0.264/0.516

Table 9. Parameter Selection and Model Performance

10.2.3 Supplement B.3: Theoretical Bounds on Approximation Error

Theoretical properties of the approximation errors of PL NNs are well studied in literature. For example, works like Hu et al. (2020) and Chiluveru (2021) construct PL NN approximators for smooth activation functions and study their error bounds theoretically. Second, regarding the KD-based technique, theoretical work shows that ReLU NNs can approximate smooth functions effectively in the context of KD (Ohn & Kim, 2019; Belomestny et al., 2022). To apply this to our specific instantiations, we state the following theorem:

Theorem (Arbitrary Exactness)

Let M be a non-piecewise linear ML model. Then for any convergence criterion $\varepsilon > 0$, it holds:

a) Functional approximation: If M admits a component-wise PL approximation, we find such an approximation M', such that, as functions, the models satisfy $\|f_M - f_{M'}\|_{L^2} < \varepsilon$.
b) KD-based approximation: There exists an output-centric approximation M', such that, as functions, the models satisfy $\|f_M - f_{M'}\|_{L^2} < \varepsilon$.

Hence, the approximations provided by both approaches are arbitrarily exact.

Proof. a) If M admits component-wise PL approximation, this means, that the function f representing M can be written as a composition $f = f_1 \circ \ldots \circ f_n$ of continuous functions f_i, such that each of the non-PL function f_j in the composition can be approximated arbitrarily well by a PL function g_j. Let $\varepsilon > 0$. We then find a composition of approximations $g = g_1 \circ \ldots \circ g_n$, such that the output of g_i never differs more than $\frac{1}{n \cdot \varepsilon}$ from f_i. By continuity, the error propagation through this composition is controlled by an according factor C_i and therefore, also the output of g is arbitrarily close to f. Therefore, it holds that $\|f_M - f_{M'}\|_{L^2} < C \cdot \varepsilon$ for $C = \prod_i C_i$. By arbitrary choice of ε we receive the stated claim. Similar constructions are instantiated in literature, where Hu et al. (2020) demonstrate such an approximation for NNs with non-PL activation functions.

b) For the KD-based (output-centric) approximation, we refer to the well-known fact in literature, that ReLU NNs are universal approximators (Lu et al., 2017). We can therefore choose M' to be a ReLU NN that is an arbitrarily close approximation of M. This yields the stated claim.

While this provides arbitrary exactness of both categories in general, we will also validate them empirically in the evaluation section for our specific purpose of model explanation.

10.2.4 Supplement B.4: Instantiation of Approximation Strategies

We now provide details on the instantiation and configuration of both approximation techniques for our evaluation.

Functional approximation. The goal of the functional approximation technique is to substitute all non-PL components of the models with PL approximate components. In our evaluation, we employ this technique for NNs with (smooth) sigmoid and tanh activation functions. For this, we train small (PL) ReLU NNs and substitute them for the non-linear activations of the original NN. We use the same approximation instantiation for each type of activation function for the whole NN, however it is possible to specifically focus on the most relevant parts of the activation function for each individual neuron. The approximate ReLU NN always has one neuron in its input and output layer, whereas number of hidden layers and neurons in each hidden layer is arbitrary. For the tanh and sigmoid activation functions, it is beneficial to focus on an interval around 0 (e.g., [-3, 3]) for training, where we use the mean square distance between the approximation and the true activation function in this interval as loss function. Finally, we manually add a pre-defined two-neuron hidden layer (before the output neuron) that bounds the NN output to [0, 1] for sigmoid and to [-1, 1] for tanh respectively. Naturally, the approximation quality depends on the capacity of ReLU NN, i.e., the number and size of its hidden layers. However, NNs with two hidden layers with three and two neurons respectively, already provide very good approximations and were therefore employed in our experiments. Figure 7 presents a selection of functional approximations with varying capacity for the tanh activation function.

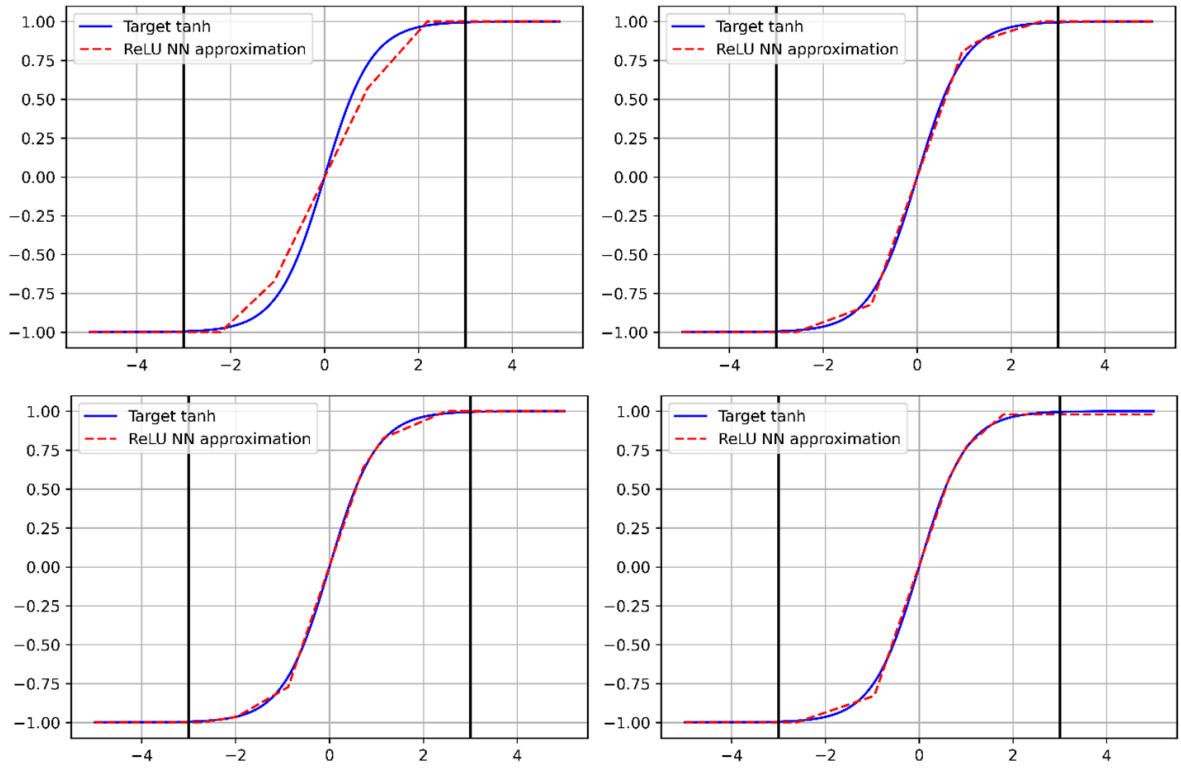


Figure 7: Functional ReLU NN Approximations with Varying Capacity for the tanh Activation Functions.

In Figure 7, approximations with two and three hidden layers are illustrated. The top left approximation has two two-neuron hidden layers and achieves a loss of 0.0167, the top right has one three-neuron and one two-neuron hidden layer and achieves a loss of 0.00078, the bottom left has one five-neuron and one two-neuron hidden layer and achieves a loss of 0.00056, and the

bottom right has two three-neuron and one two-neuron hidden layer and achieves a loss of only 0.00054.

KD-based approximation. The goal of the KD-based approximation approach is to train a PL model that closely mimics the non-PL original model as a whole. In our evaluation, we employ this approach for NNs with sigmoid and tanh activation functions as well as SVMs with non-linear polynomial and RBF kernels. For all of these non-PL models, we train a ReLU NN with either two or three hidden layers with ten neurons each. The training data for these proxy models are artificially generated by randomly sampling around each instance used for the training of the original, non-PL model. Following a heuristic from Kim et al. (2023) for local neighborhoods, for each training instance $x_{train,i}$ of the original model, 1,000 samples are drawn uniformly from a sphere with a radius r_i proportional to the maximum distance to another training instance (i.e., $r_i = 0.2 \cdot \max_{x'_{train}} \left\| x_{train,i} - x'_{train} \right\|_{eucl}$). The resulting samples are then labeled with the predictions of the original model.

10.2.5 Supplement B.5: Evaluation of EXPLORE

For the fidelity and robustness analyses, EXPLORE's explanation $\bar{f}_{x_i}$ for an instance $x_i \in X$ from the test dataset is given by the functional reconstruction of the DBs and CSs in a neighborhood of the instance that is large enough to contain all samples $z \in Z$ for the local fidelity metric $LocalFid\left(\bar{f}_{x_i}, x_i\right) = Acc_{z \in Z}\left(f(z), \bar{f}_{x_i}(z)\right)$ as in Term (15) of the main manuscript. Following Kim et al. (2023), we uniformly draw 1,000 samples from a sphere around x_i to obtain Z. The radius r of the sphere is given by $r = 0.2 \cdot \max_{x_{train}} \|x_i - x_{train}\|_{eucl}$, where the maximum of the distance of x_i and each (other) training instance is taken. Then, $\bar{f}_{x_i}$ is given by the reconstruction of f in a subspace of X that contains this sphere. More precisely, $\bar{f}_{x_i}$ is defined as the function that maps each $z \in Z$ to the class of the CS it lies in, determined by the system of inequalities (cf. Term (5) in Definition 4 in the manuscript) that z satisfies. In our fidelity and robustness evaluation, we determine all DBs and CSs of f globally in a single step for the entire feature space of an ML model as this allows the assessment of all instances regarding an ML model in one single step. For the faithfulness analysis, the explanation of EXPLORE for a test instance $x \in X$, we set $\bar{f}_x = \overrightarrow{d_{db,j}}(x)$ for the class j that is closest to x, i.e., the class j for which $d_{db,j}(x) = \min_k d_{db,k}(x)$ (cf. Definition 7) holds. The feature vector $w_{\bar{f}_x}$ thus contains the features in descending order with respect to the (absolute) feature attributions in $\bar{f}_x$. For the stability of the counterfactual methods in the faithfulness analysis, we assessed whether two explanations are equal by checking the condition $|\bar{f}_{x,1} - \bar{f}_{x,2}|_i < 10^{-8}$ for every vector entry i to account for small numerical inaccuracies (e.g., to prevent rounding errors).

We implemented EXPLORE as a Python tool that is not yet linked here to maintain anonymity. Moreover, we applied an interim performance-enhanced version of the EXPLORE algorithm for all approximations of non-PL models. For the evaluation of S-LIME regarding fidelity and robustness for the sigmoid NN on the COMPAS dataset, we observed a convergence error of S-

LIME for the fifth fold model. We therefore report the average fidelity and robustness over the first four folds in this case.

10.3 Online Supplement C: Prototypical Workflow and Visualization for User Interaction

To showcase an exemplary scenario and thereby demonstrate the application of selected feature attributions, we utilize the Prosper dataset in the following. Here, we examine a loan application x with the class prediction LOANDENIED (high estimated loss) by the NN used. The loan application represents an individual recently re-employed, including features such as a loan amount of $ 4127 and a loan term of 36 months. The loan broker, together with the applicant, aims to explore the individual prediction of the NN in the scenario, i.e., why the application was not assigned to the class LOANAPPROVAL and precisely, what feature value changes are needed for the application to be assigned to LOANAPPROVAL (contrastive explanations). By determining the Insensitive Sphere for the class LOANAPPROVAL and the feature vector to the nearest DB (i.e., $VFA_{DB,x}^{C,LoanApproval}$), we gain initial insights into the feature attributions required for the desired class change. To make these attributions directly understandable for users, we initially focus on the two most relevant features with the largest absolute standardized values in this determined feature vector: *Stated Monthly Income* and *Monthly Loan Payment*. The real-world values of these two features in the 2-dimensional vector are $\begin{pmatrix}+\$724.93\\+\$24.15\end{pmatrix}$ (as illustrated in Figure 8, right side), meaning the applicant would need to increase the monthly income by approximately $725 and the monthly repayment amount by about $25 to qualify for the loan (with all other features remaining unchanged). For verification, the loan broker and the applicant can examine the neighborhood of the considered instance which includes both past loan applications from the original Prosper dataset with the ground truth class label LOANDENIED and past applications beyond the identified DB that indeed have the ground truth class label LOANAPPROVAL. This additional information further supports the provided predictions and the local explanations for the ML model provided by EXPLORE. However, the presented nearest contrastive explanation with $\begin{pmatrix}+\$724.93\\+\$24.15\end{pmatrix}$ is not an economically viable option for the applicant due to the significant increase in the Feature *Stated Monthly Income*. Therefore, it is analyzed whether by maintaining the current direction (and thus the identified DB), the proposed nearest explanation can be altered by additionally changing the feature *Term*, thus yielding an economically feasible VFA. Incrementally expanding the Insensitive Sphere, e.g., by 1%-steps of the current radius, is a systematical procedure to obtain the nearest $VFAs$ included in $SFA_{DB,x}^{C,LoanApproval}$. This reveals the existence of various feasible $VFAs$, e.g., up to a maximum extended loan term of approximately 3 months resulting in $\begin{pmatrix}+\$589.99\\+\$24.37\\+2.97\text{ months}\end{pmatrix}$, as illustrated by the vector in Figure 8 (left side). These allow extending the loan term, while not increasing *Stated Monthly Income* as much as before and maintaining the slight increase in *Monthly Loan Payment*. As this scenario illustrates, the nearest DBs and

CSs can be explored in a systematic manner, while preserving clarity and interpretability. Moreover, users can also choose to explore the neighborhood in greater detail using adaptable 2- or 3-dimensional visualizations exactly representing the actual DBs and CSs as shown in Figure 8.

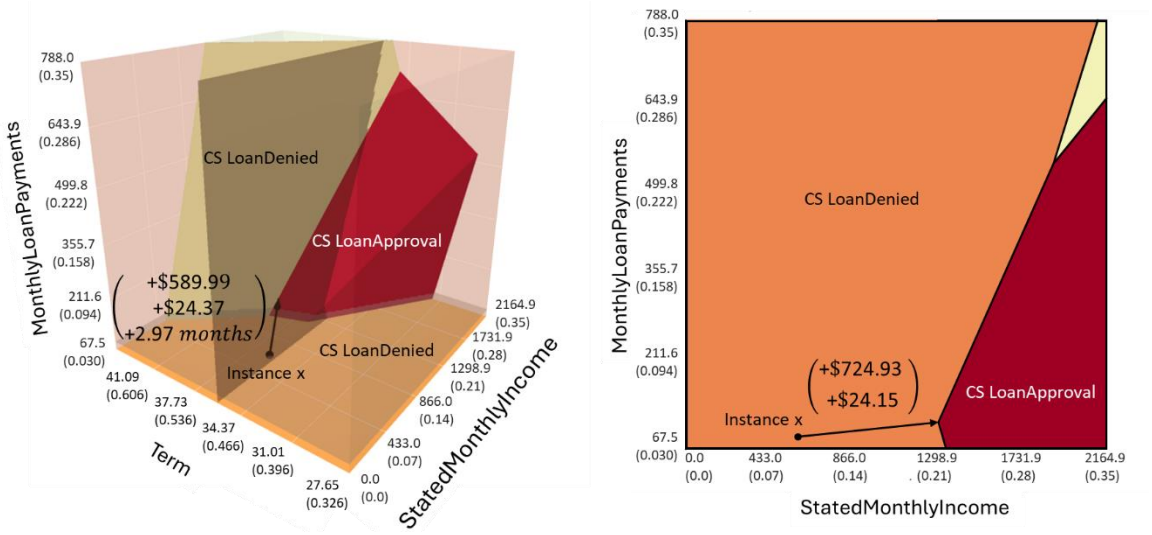


Figure 8: 3D and 2D Visualization of Neighborhood and VFAs

The 3-dimensional visualization (left side) shows the considered instance in its neighborhood together with the DBs to Class LOANAPPROVAL (red), the Class LOANDENIED of high estimated loss in which the instance x is located (orange) as well as the Class LOANDENIED of lower estimated loss (yellow)[5]. The 2-dimensional plot (right side) is indicated in the 3-dimensional plot as the shadowed vertical hyperplane. It outlines the *VFA* presented above which only relies on changes in the most relevant attributes *Stated Monthly Income* and *Monthly Loan Payment*, and represents the closest vector in this subspace. In addition to examining the nearest DBs and CSs, users can also explore whether a class change to LOANAPPROVAL is achievable by selecting specific features (cf. Proposition 2c). Such selections typically align with the user's personal preferences or the practical constraints of the features and their interactions in the specific application context. In our scenario of the applicant, some features may be very difficult or even impossible to alter. Other features, however, can be adjusted upward or downward, or only within a specific interval (cf. *Stated Monthly Income* in previous example). For instance, the applicant wants to know if changing only the *Term, Employment Status Duration, Credit Score Range Lower, Credit Score Range Upper,* and/or *Recommendations* could result in a change to LOANAPPROVAL (with all other feature values held constant). Given this user-defined subspace, EXPLORE can guarantee (with certainty) that no DBs or CSs of the class LOANAPPROVAL exist within such a user-defined subspace. This demonstrates that the desired class LOANAPPROVAL cannot be achieved solely by modifying these features. Instead, using the set of contrastive explanations in the considered instance's neighborhood, it can be shown that at least one of the three features *Stated Monthly*

[5] Note that since fewer ground truth instances exist in the immediate proximity of the DB between CS LOANDENIED und CS LOANAPPROVAL, the trained NN relies more on interpolation in this region.

Income, Loan Original Amount, or Monthly Loan Payment must be altered to achieve the class LOANAPPROVAL.

On this basis, various feasible local explanations that align with possible user preferences can now be determined and carefully analyzed using EXPLORE. Finally, suitable alternatives can be selected based on criteria such as the magnitude of feature changes or the number and the type of features to be adjusted. While the exemplary scenario above primarily involved analyzing contrastive explanations due to the desired class change, in the case of maintaining the current class (i.e., preserving the class LOANAPPROVAL), abductive explanations can also be explored. This shows the potential of our method to enhance the transparency of the ML model and its predictions. However, we recognize that real instances in high-dimensional spaces may be surrounded by multiple DBs, creating challenges in effectively conveying this complexity. Nonetheless, when different feasible abductive and contrastive explanations actually exist, it is essential to present them transparently and not ignore them.

10.4 References

Belomestny, D., Naumov, A., Puchkin, N., & Sergey Samsonov. (2023). Simultaneous approximation of a smooth function and its derivatives by deep neural networks with piecewise-polynomial activations. *Neural Networks*, *161*, 242–253.

Chiluveru, S., Tripathy, M., & Bibhudutta (2021). Non-linear activation function approximation using a REMEZ algorithm. *IET Circuits, Devices & Systems*, *15*(7), 630–640.

Hu, X., Liu, W., Bian, J., & Pei, J. (2020). Measuring model complexity of neural networks with curve activation functions. In *Proc. 26th ACM SIGKDD International Conference on knowledge discovery & data mining* (pp. 1521-1531).

Kim, B., Srinivasan, K., Kong, S. H., Kim, J. H., Shin, C. S., & Ram, S. (2023). ROLEX: A novel method for interpretable machine learning using robust local explanations. *MIS Quarterly*, *47*(3), 1303–1332.

Lu, Z., Pu, H., Wang, F., Hu, Z., & Wang, L. (2017). The expressive power of neural networks: A view from the width. In *Advances in Neural Information Processing Systems*.

Ohn, I., & Kim, Y. (2019). Smooth function approximation by deep neural networks with general activation functions. *Entropy*, *21*(7), 627.

Petersen, P., & Voigtlaender, F. (2018). Optimal approximation of piecewise smooth functions using deep ReLU neural networks. *Neural Networks*, *108*, 296–330.

Reddy, P., & Gujral, A. S. (2025). Improving neural network efficiency using piecewise linear approximation of activation functions. In *Proc. FLAIRS-38*.

4 Conclusion

Section 4.1 summarizes the major findings of the dissertation. Thereafter, Section 4.2 provides an outlook and examines directions for future work. This section primarily focuses on overarching points that are not specific to individual papers, whose respective conclusions are presented within the papers themselves.

4.1 Major Findings

In recent years, the rapid technological progress driven by AI has fundamentally reshaped societies and transformed the way businesses and organizations operate. However, as AI systems are increasingly deployed in complex and high-stakes settings, uncertainty emerges as central and pervasive challenge, as it can fundamentally affect the validity and transparency of AI-based decisions. Against this background, this dissertation examines nine research questions regarding the two focal topics of addressing data uncertainty (Section 2) and explaining ML model predictions (Section 3). In this way, it contributes to enabling the valid and transparent use of AI in organizational and societal application settings. The major findings of this dissertation are outlined in the following.

One major finding supported by all papers on the first focal topic is that a precise and differentiated understanding of the underlying DQ defects plays a central role in effectively addressing data uncertainty in the context of AI applications. In particular, the papers highlight that data uncertainty should not be viewed as a vaguely or uniformly considered phenomenon, but one that requires careful and explicit characterization and modeling. Indeed, the taxonomy proposed in Section 2.1 (RQ1) highlights that methods addressing DQ defects in ML can and should be distinguished based on the specific DQ dimension and particular DQ-related task that they focus on. At the same time, it reveals that the underlying DQ defects are frequently insufficiently specified by these methods from literature. As a result, it is often unclear whether a given method can be effectively applied (or transferred) to address the DQ defects in the application context at hand. The theoretical foundations of DQ presented in Section 2.2 (RQ2) further support a detailed understanding of DQ defects. In particular, the formal and precise specifications of the DQ dimensions accuracy and currency resolve the vagueness and incompatibility of existing definitions in literature and enable a rigorous characterization of DQ defects related to these DQ dimensions. As demonstrated for an ML task from the healthcare domain, these specifications can also be directly leveraged to effectively address DQ defects, resulting in a substantial improvement in model performance. Moreover, the detailed understanding of the relationship between accuracy and currency provides insights into how DQ defects across these dimensions relate to one another, indicating when they can be addressed by the same method. Building on this conceptual foundation, the benefits of a detailed understanding of DQ defects further become evident in the concrete methods proposed in Sections 2.3 (RQ3) and 2.4 (RQ4). Existing methods for uncertainty propagation often treat data uncertainty uniformly and generally assume that it can be modeled by a specific form of distribution such as a Gaussian. Because of this assumption,

they provide approximate and inexact uncertainty estimates, which can undermine the validity of subsequent decisions. In contrast, the method proposed in Section 2.3 allows incorporating the individual underlying DQ defects and exactly propagates the arbitrary probability distributions modeling them, hence providing a valid basis for decision-making. Addressing RQ4, the language model-based method for assessing and improving the currency of unstructured textual data (proposed in Section 2.4) enables the identification and targeted improvement of outdated parts in textual statements. In the respective paper, the method is evaluated on a dataset of fact-based Wikipedia sentences, where it achieves strong performance. Importantly, these results highlight that effectively addressing such DQ defects requires a detailed understanding of their characteristics, as the method's operationalization explicitly utilizes information sources that capture the underlying causes for the defects. For instance, news articles are used to track events affecting the currency of the respective Wikipedia sentences, while other application settings or DQ defects may require alternative sources. Taken together, this underlines that a detailed understanding of DQ defects is not merely a theoretical concern, but a necessary requirement for both the development and the effective deployment of methods addressing data uncertainty in ML.

Building on the detailed understanding of DQ defects, a second major finding regarding the first focal topic is that methods for addressing data uncertainty and DQ defects—despite forming a broad, fragmented, and thus complex landscape—can be systematically structured and thereby made accessible for researchers and practitioners alike. The taxonomy proposed in Section 2.1 provides such a structure, thus enabling the systematic classification of methods addressing data uncertainty and DQ defects in the context of ML, and supporting their development, selection, and application. It organizes methods along seven dimensions and seventeen subdimensions, covering aspects such as how data uncertainty is modeled, which DQ defects are addressed, the strategies used to identify and resolve DQ defects, the type of ML models involved, and how the method is evaluated. The reliability, understandability, completeness, and usefulness of the taxonomy was confirmed by surveyed researchers and practitioners, thus providing a clear and intuitive framework for addressing data uncertainty in ML. Importantly, the taxonomy not only describes this research landscape as a whole (including, e.g., remaining research gaps) but using it to classify a specific method also yields valuable insights into the method's characteristics and position within the research field. For example, the taxonomy reveals that the method proposed in Section 2.3 addresses data uncertainty at the stage of the model output by exactly propagating it through the model. This clearly distinguishes it from methods that address data uncertainty at the input stage of ML-based data analysis before the model is applied. Such a classification can also be used to identify a clear set of relevant taxonomy characteristics that defines which methods are comparable, e.g., as competing artifacts in an evaluation. Further, the taxonomy allows exploring a method's full range of applicability in a structured and systematic manner. This can be illustrated by the method proposed in Section 2.4 which leverages a language model and external information in the form of news articles to address currency defects in textual data. Crucially, since large language models are powerful and flexible tools and the external information sources can be configured arbitrarily, the method's applicability can be systematically

explored based on the taxonomy. For example, it can be assessed whether the method could be operationalized for DQ defects related to other DQ dimensions, such as accuracy or completeness, or whether it could also be applied to structured data, which modern language models are capable of handling even though this is not their typical use case (e.g., Jiang et al., 2023). Overall, viewing the research field of addressing data uncertainty and DQ defects from this structured, taxonomy-based perspective provides valuable guidance for method selection, evaluation, and development, thus enabling meaningful contributions that advance this important field.

The third major finding of this dissertation is that reconstructing ML models as piecewise linear functions provides a powerful foundation for enabling rigorous analyses of their behavior. More precisely, it yields a functional and definite representation of the input-output relationship of ML models from a broad range of model types that can either be represented exactly by piecewise linear functions or be approximated by them with arbitrary precision. Crucially, this representation reveals the local behavior even of complex and opaque ML models, as they are expressed by simple affine linear functions within their respective linear regions. This powerful concept is leveraged to tackle complex challenges across both focal topics of the dissertation. In the method for propagating data uncertainty through neural networks proposed in Section 2.3, the piecewise linear reconstruction of the network plays a central role. More precisely, representing the neural network as piecewise linear function allows the method to avoid the computationally expensive and error-prone propagation of uncertainty through each network layer. The empirical evaluation on real-world classification and regression datasets demonstrates that, in this manner, arbitrary probability distributions modeling data uncertainty can be propagated through neural networks with very high exactness, thereby providing a valid basis for subsequent decisions under uncertainty. Furthermore, the piecewise linear representation provides access to the functional and definite characterization of the decision boundaries and class subspaces of ML models (RQ7), forming the basis for rigorous sensitivity analyses and the derivation of faithful explanations addressing RQ8 and RQ9, respectively. In Section 3.2, the decision boundaries are used to define two well-founded sensitivity measures, enabling a rigorous analysis of how changes in the feature values of an instance affect the corresponding ML model prediction (RQ8). More precisely, the first measure captures the class distribution in the proximity of a given instance, and the second measure determines the minimal feature value changes that are necessary to alter the model's predicted class. This yields valuable insights beyond the mere class prediction, as it reveals whether a prediction is stable, i.e., it remains unchanged even under substantial feature value changes, or whether it is highly sensitive, with even small changes already leading to a different prediction. Such information is particularly critical in high-stakes applications, e.g., when a low-risk model prediction is made despite the local neighborhood being dominated by a high-risk class or when a decision boundary towards a high-risk class lies in close proximity. Finally, in Section 3.3, the functional definition of the decision boundaries of a model enables deriving faithful feature attribution and counterfactual explanations (RQ9). Instead of relying on simplifying assumptions or heuristics, the explanations correctly reflect the behavior of the underlying model. More precisely, they identify the features that are truly relevant for a given prediction and provide accurate attributions or required changes, thereby yielding correct and clearly

interpretable results. Overall, the piecewise linear reconstruction of ML models proves to be a highly flexible and powerful concept, enabling the exact propagation of data uncertainty, a rigorous analysis of sensitivity, and the derivation of faithful explanations, demonstrating its potential for a variety of ML-related analyses and applications.

A fourth major finding of this dissertation is that explanation uncertainty can be substantially reduced or even completely eliminated, thereby reducing explanation errors and enabling faithful explanations. First, the explanation uncertainty caused by the XAI method itself is clearly distinguished from model uncertainty in Section 3.1 (RQ6). Furthermore, both types of uncertainty are shown to be decomposable into a structural bias component and an estimation variance component, providing a detailed understanding of the respective uncertainties and their impact on explanation correctness. For explanation uncertainty, the structural bias component describes the methodological ability of an XAI method to reflect the underlying ML model and its decision behavior. For instance, this ability is often limited by the use of simplifying surrogate models (such as linear regressions as used in LIME), which are typically unable to capture the complex decision behavior of highly non-linear models like neural networks. On the other hand, the estimation variance of an XAI method refers to the uncertainty arising from limited information about the model's decision behavior, for instance, due to a limited number of samples used to train the surrogate model. Against this background, by leveraging the true decision boundaries of the underlying ML models, the novel XAI method EXPLORE introduced in Sections 3.2 and 3.3 effectively addresses both components, thereby enabling faithful explanations that are not affected by explanation uncertainty (RQ7 & RQ9). More precisely, EXPLORE relies on a functional and definite reconstruction of the decision boundaries, which is either exact or can be approximated with arbitrary precision depending on the ML model type. By directly operating on these decision boundaries, EXPLORE can flexibly capture any possible decision behavior of the given model, and is therefore not subject to structural bias. Furthermore, because the decision boundaries are reconstructed based on the model parameters rather than partial information on the model's decision behavior (such as random samples or perturbations), they are represented deterministically and completely and hence do not suffer from estimation variance. Taken together, EXPLORE enables the derivation of both abductive and contrastive explanations that correctly reflect the true decision behavior of the underlying ML model and are therefore not affected by explanation uncertainty. This is substantiated by the theoretical and empirical evaluation which confirms the fidelity of EXPLORE's local reconstruction and the faithfulness of its feature attribution and counterfactual explanations. Overall, EXPLORE enhances model transparency and thereby strongly supports the assessment of ML models and their predictions, particularly in high-stakes settings, providing a sound basis for subsequent decisions.

4.2 Directions for Future Research

This dissertation comprises seven papers addressing nine research questions related to uncertainty in AI. However, as discussed, this is a complex and multifaceted topic, and certain aspects

and potential analyses remain beyond the scope of this work. Therefore, several possible directions for future research are outlined in the following.

To begin with, the dissertation presents a taxonomy to structure and systematize the research field of methods addressing data uncertainty and DQ defects in the context of ML (Section 2.1). At the same time, the theoretical framework of this dissertation introduced in Section 1.1.2 as well as existing works in literature (e.g., Chiaburu et al., 2024; Förster et al., 2025; Förster et al., 2026) highlight that uncertainty in ML is not limited to data uncertainty induced by DQ defects, but also encompasses other sources of uncertainty such as model uncertainty or explanation uncertainty. Building on this, future work could focus on developing similar taxonomies for methods addressing model uncertainty and explanation uncertainty, respectively. As for data uncertainty, methods targeting these sources of uncertainty are highly diverse and differ substantially in their underlying assumptions, mechanisms, and the pipeline stage at which the uncertainty is addressed. For example, model uncertainty can be addressed at the model stage through approaches such as ensembles (e.g., Lakshminarayanan et al., 2017) or Monte Carlo dropout (Gal & Ghahramani, 2016), but also at the input stage through approaches such as active learning (e.g., Settles, 2012) or data augmentation (Mumuni & Mumuni, 2022). Regarding explanation uncertainty, some methods aggregate explanations from different XAI methods to address variability across explanations (Decker et al., 2024; Pirie et al., 2023), while others aim to improve the stability (i.e., the estimation variance) within a given XAI method such as LIME, e.g., by adaptively determining the required number of random samples to obtain stable results (Zhou et al., 2021). Systematically structuring these methods with a taxonomy could support their comparison, selection, and further development in a manner similar to the taxonomy proposed for data uncertainty and DQ defects. The taxonomy by Förster et al. (2025) examines the different sources of uncertainty in more detail within the context of XAI. Even though it does not focus on corresponding methods addressing them, it may serve as a valuable starting point for this future research direction.

Furthermore, the first major finding in Section 4.1 has already discussed the importance of having a precise and deep understanding of the DQ defects that induce data uncertainty. To this end, theoretical foundations of DQ together with formal specifications of the DQ dimensions accuracy and currency are introduced in Section 2.2. On this basis, a highly relevant direction for future research is to rigorously connect these theoretical foundations to formal approaches for modeling data uncertainty and to systematically use them as basis for defining and developing such modeling approaches. In this context, the theoretical foundations could also be extended beyond the specifications of accuracy and currency to cover further important DQ dimensions such as completeness and consistency, thereby broadening its applicability to a wider range of uncertainties corresponding to different DQ dimensions. For example, a first concrete realization of such uncertainty modeling based on DQ-related events and the specification of currency was already outlined in the appendix of Paper 2 on the theoretical foundations of DQ. In this case, for specific drug administration events, changes in the associated clinical features (such as body temperature) of intensive care unit patients were quantified using rules generated based on the

dataset, yielding a well-founded modeling of data uncertainty based on the theoretical foundations of DQ. Consequently, this enabled DQ improvements, which in turn resulted in substantially improved performance in the associated length-of-stay prediction task. This illustrates that a theoretically grounded modeling of data uncertainty that explicitly reflects the underlying DQ defect provides substantial advantages and supports valid downstream decision-making. Notably, as discussed in the first major finding, this is also directly relevant for methods for uncertainty propagation (cf. Section 2.3). To summarize, future work should aim at grounding existing and novel approaches for data uncertainty modeling in the formal theoretical foundations of DQ introduced in this dissertation. In particular, these foundations could also be connected to existing probabilistic DQ metrics such as the ones proposed by Heinrich and Klier (2015) and Heinrich et al. (2018) to further strengthen their theoretical grounding.

The language model-based method for improving the currency of textual data proposed in Section 2.4 also allows for several interesting extensions that could be explored in future research. In the presented operationalization, the method used external information extracted from news articles to assess and improve the currency of textual statements from a public wiki. First, the method could also be applied to other data types such as structured data, as modern language models are increasingly capable of handling such formats even though this is not their originally envisioned use case (Jiang et al., 2023). Second, the method could also be applied to other textual data that refer to continuously changing real-world facts but must remain current, such as enterprise wikis or textual information on webpages. In this case, the sources of external information need to be adjusted accordingly, e.g., by using more recent internal documents within the organization maintaining the wiki or webpage. Third, DQ is a multidimensional construct, and the method could also be used to assess and improve DQ with respect to other dimensions, such as accuracy, consistency, or completeness. Similarly, the sources for external information need to be selected accordingly and should contain reliable information on the potentially incorrect, inconsistent, or missing data. Fourth, subsequent ML applications (e.g., RAG systems) could be applied to the considered textual data, enabling the evaluation of the method not only at the input stage (as conducted in Section 2.4), but also at the model stage of an associated ML-based data analysis pipeline.

A further intriguing direction for future research is the development of a comprehensive, unifying theoretical framework that views different types of ML models through the common lens of piecewise linear functions. While different ML model types such as neural networks, convolutional neural networks, decision trees, random forests, boosting methods, or support vector machines differ substantially in their internal structure and learning mechanisms, they all can be represented exactly or approximated very closely by piecewise linear functions. As discussed for a broad set of ML model types in Sections 3.2 and 3.3, this enables detailed insights into the inner workings and the decision behavior of these models. In particular, the piecewise linear representation allows for a functional representation of the decision boundaries and class subspaces of the models, thereby enhancing their transparency. Notably, existing literature that views ML models as piecewise linear functions predominantly focuses on ReLU neural networks

and typically considers individual models in isolation, e.g., to analyze their expressivity (Hanin & Rolnick, 2019; Montúfar et al., 2014; Raghu et al., 2017) or robustness (Croce et al., 2019; Jordan et al., 2019). Crucially for this future research avenue, the common piecewise linear perspective enables a wide range of analyses across multiple ML models, even of different types, as they can now be viewed and analyzed using a shared theoretical basis. As a result, this perspective would directly generalize results for piecewise linear functions to a broad set of model types. In particular, the uncertainty propagation method proposed in Section 2.3 could be readily applied to any model for which a piecewise linear representation is available. As another example, the agreement of different models could be rigorously analyzed by functionally intersecting their class subspaces, thereby precisely characterizing the areas of the input space in which their predictions coincide or differ, respectively. This enables an exact, functional assessment of the agreement between models in specific areas, which may remain undetected by instance-based comparisons, particularly in sparsely populated areas of the input space. Beyond this example, this comprehensive framework would also enable a large variety of additional analyses regarding, e.g., model uncertainty and its quantification, model training procedures, or targeted model improvement, which would apply for all ML models allowing for a piecewise linear representation. Overall, this illustrates the broad impact of the common piecewise linear perspective and its ability to enable such analyses even across different model types, underlining its significant potential for future research.

In Section 3.1, many well-known XAI methods have been examined in the context of model and explanation uncertainty, which are further decomposed and analyzed in detail. Another promising direction for future research is to extend this analysis and evaluation of multiple XAI methods regarding data uncertainty. In particular, the variability of explanations (produced by a specific XAI method) induced by data uncertainty could be systematically investigated. Notably, some existing works in literature already examine the robustness of explanations from selected XAI methods under small perturbations (e.g., Alvarez-Melis & Jaakkola, 2018; Dombrowski et al., 2019). Crucially, however, this perspective can now be complemented and enhanced by evaluating EXPLORE (Sections 3.2 and 3.3) which, as discussed, is not affected by explanation uncertainty. As a result, for a given, fixed ML model (i.e., in the absence of model uncertainty), the isolated effect of data uncertainty on explanations can now be precisely examined using EXPLORE. This is not readily feasible for other XAI methods, where this effect is typically confounded by explanation uncertainty. Interestingly, this also enables a comparison of how explanations should vary (or remain stable) under data uncertainty according to the true decision boundaries of the model, and how they actually behave when generated by a given XAI method such as LIME. This could, in turn, serve as basis to reveal and study novel XAI-related phenomena, similar to those identified for model and explanation uncertainty, such as the Consistency Trap discussed in Section 3.1. Against the background that the robustness of explanations is often considered a key desideratum in XAI literature (e.g., Alvarez-Melis & Jaakkola, 2018; Guidotti et al., 2024), this type of critical analysis is highly promising as it can provide further insights by distinguishing cases in which robustness should indeed hold and those in which explanations should be expected to change, even under small perturbations.

Further, the analysis of the effects of uncertainty on the consistency and correctness of explanations in this dissertation exclusively focuses on XAI methods yielding feature attributions (cf. Section 3.1). However, it would be highly intriguing to extend this analysis to other types of explanations, such as surrogate models, decision rule sets, or counterfactual explanations. While the methodology of the existing analysis for feature attribution methods can be transferred very naturally to counterfactual explanations, for surrogate models and rule-based explanations alternative metrics for correctness and consistency may need to be identified or developed. Furthermore, an extension to other data types such as image or text data would also be very relevant as literature also offers a wide range of attribution methods for these settings (Madsen et al., 2022; Ribeiro et al., 2016; Zhang et al., 2021b). Overall, these extensions would enable more comprehensive conclusions about the behavior of explanations across different application settings and XAI methods.

Another promising avenue for future research is the development of an EXPLORE variant that explains a given black-box ML model using a more interpretable surrogate model. In Sections 3.2 and 3.3, EXPLORE uses an exact and complete representation of the decision boundaries and class subspaces of an ML model, and thus captures its full complexity. While this property is one of EXPLORE's main strengths and guarantees perfect fidelity scores for many ML models, it also raises the question of how this powerful representation can best be made interpretable to users. Indeed, both fidelity and interpretability are widely recognized as key desiderata of surrogate models although they often exhibit an inherent trade-off (Gilpin et al., 2018; Guidotti et al., 2018). More precisely, simpler surrogate models are typically more interpretable but often lack the capacity to capture the decision behavior of the underlying black-box model with high fidelity, especially when the model is highly complex. Therefore, a promising direction for future research is to fully leverage EXPLORE's potential to provide a full spectrum of possible representations with varying degrees of fidelity and interpretability. Starting from the exact and complete representation with perfect fidelity (which may, however, be too complex to be interpretable) one could proceed via targeted, stepwise simplifications of this representation to obtain increasingly interpretable surrogate models while still preserving high levels of fidelity. In particular, this could involve summarizing multiple, very small decision boundary segments, or adjacent boundaries with similar orientation in the feature space. In this way, EXPLORE has an advantage over existing surrogate model-based XAI methods such as LIME, ROLEX, or MAPLE, as it does not rely on a fixed surrogate model type (e.g., a linear regression or decision tree). Instead, it is based on the true decision boundaries of the model and is therefore not inherently restricted in the range of different decision behaviors it can reflect. This stepwise simplification process could be traversed until the desired degree of simplification and interpretability is reached, which allows the trade-off between fidelity and interpretability to be calibrated in a flexible and controlled manner. Finally, the explanation error of this EXPLORE variant could also be analyzed in the context of the explanation error decomposition introduced in Section 3.1, where the controlled simplification of the decision boundary representation would correspond to a controlled increase in the XAI structural bias, depending on the chosen configuration.

As ML models are increasingly deployed in high-stakes decision-making contexts, issues of bias and fairness have become of central concern. In such settings, it is widely recognized as problematic when ML models make predictions or recommendations based on sensitive features such as sex or race, or on features that act as proxies for them (Mehrabi et al., 2021; Pessach & Shmueli, 2022; Singh et al., 2024b). In this context, XAI has become increasingly important, as it enhances model transparency and enables analyzing the reasoning behind individual predictions, including the features that most strongly influence them (Deck et al., 2024; Zhao et al., 2023). In this sense, EXPLORE and its vector- and set-based feature attributions introduced in Section 3.3 could be used to examine the fairness of ML model predictions by determining which changes in feature values lead to the retention or a change of the current class prediction, respectively. In particular, the feature attribution vectors indicate whether model predictions are influenced by sensitive features and, if so, to what extent. This also directly relates to the research strand of counterfactual fairness, which assesses whether a prediction would change if only the values of sensitive features are altered while all other "fair" features remain unchanged (e.g., Kusner et al., 2017; Wu et al., 2019). Moreover, EXPLORE also offers starting points for the development of novel fairness metrics. Indeed, existing fairness metrics largely focus on group-level comparisons of model predictions across subpopulations regarding sensitive features (Pessach & Shmueli, 2022; Rabonato & Berton, 2025). Therefore, incorporating the decision boundaries in the proximity of predicted instances may provide additional valuable insights into model fairness beyond solely considering and comparing the prediction outcomes. EXPLORE may therefore serve as a basis for promising, novel approaches for the detection and quantification of bias and discrimination in ML models, and could thus contribute meaningfully to the research field of fairness in ML.

Finally, this dissertation and the theoretical framework introduced in Section 1.1.2 focus on data, model, and explanation uncertainty and therefore do not explicitly consider the role of the human user in this context. However, incorporating this perspective would be very valuable, particularly for assessing the practical utility and impact of the methods proposed in both focal topics. Considering the human user as a complementary source of uncertainty represents an important direction for future research in the overarching context of uncertainty in ML as examined in this dissertation. Human uncertainty refers to inconsistencies and variations in how individuals perceive, interpret, and respond to AI systems (Förster et al., 2026). It is largely determined by the individual's subjective experiences, task-specific knowledge, and AI literacy, and critically influences how model outputs or explanations are understood and acted upon in practice (Fügener et al., 2022; Gilpin et al., 2018; Pinski et al., 2023). In particular, a human-centered evaluation of EXPLORE would be highly valuable for studying and understanding how its faithful explanations reflecting the exact decision behavior of the model are processed and utilized by users and domain experts such as, e.g., physicians or credit scoring analysts. This would also provide insights into which representations and visualizations of decision boundaries and explanations best foster the interpretability, trust, and usability of EXPLORE.

Of course, future research opportunities regarding both focal topics are not limited to the specific contributions discussed in this dissertation but can also be considered from a more general standpoint. Indeed, DQ and data uncertainty as well as the explainability of ML models remain active and rapidly evolving areas of high interest and relevance, where substantial progress is being made while many challenges still remain and continue to emerge. This is particularly evident in the case of large language models, whose ongoing increases in capacity and performance continue to drive their widespread adoption across a broad range of domains and applications in business and society alike (Maslej et al., 2025). At the same time, such large language models generate vast amounts of unstructured text which are known to be frequently affected by hallucinations and thus, DQ defects (Bender et al., 2021; Ji et al., 2023). This has critical implications for downstream applications, as such generated text is increasingly used as decision support or as training data for new generative models, which may in turn amplify and reinforce existing errors (e.g., Huynh, 2024; Shumailov et al., 2024). However, successfully addressing these DQ-related challenges is substantially hampered by the lack of transparency in how modern generative models produce their outputs and predictions, largely due to their complex, parameter-heavy architectures. This makes it very difficult to explain their generated outputs and in particular, to reliably identify and understand the origins of DQ defects, which is crucial for their effective detection and resolution. Accordingly, research on the explainability and interpretability of large language models has begun to emerge, with initial concepts and techniques being proposed in the literature, in part by revisiting and extending established perspectives (Singh et al., 2024a; Zhao et al., 2024). Importantly, future research is required to further advance this field by developing novel concepts and methods that enhance the interpretability of generative models and provide meaningful explanations for their behavior, which may help to establish effective safeguards against hallucinations and other DQ-related insufficiencies. Such developments are essential to ensure the responsible and trustworthy use of these highly capable models, particularly in high-risk applications and in light of regulatory frameworks such as the EU AI Act (European Parliament, 2025).

To conclude, this dissertation contributes novel concepts and methods for modeling and addressing uncertainty in AI. In the two focal topics of addressing data uncertainty and explaining ML model predictions, these contributions support the valid and transparent use of ML in organizational and societal decision-making contexts. At the same time, the rapid advancement of AI systems, together with the multifaceted and pervasive nature of uncertainty and its diverse sources, gives rise to a range of open challenges that warrant further investigation. Building on its major findings, this dissertation provides a foundation for advancing the understanding and handling of uncertainty in AI, thereby opening up several promising directions for future research in this highly relevant field.

5 References

Abdelaziz, A. H., Watanabe, S., Hershey, J. R., Vincent, E., & Kolossa, D. (2015). Uncertainty propagation through deep neural networks. In *Interspeech*.

Aebi, D., & Perrochon, L. (1993). Towards improving data quality. In *Proceedings of the International Conference on Information Systems and Management of Data*.

Agarwal, C., Krishna, S., Saxena, E., Pawelczyk, M., Johnson, N., Puri, I., Zitnik, M., & Lakkaraju, H. (2022). OpenXAI: Towards a transparent evaluation of model explanations. In *Advances in Neural Information Processing Systems*.

Alasalmi, T., Koskimäki, H., Suutala, J., & Röning, J. (2015). Classification uncertainty of multiple imputed data. In *Proceedings of the 2015 IEEE Symposium Series on Computational Intelligence*.

Alvarez-Melis, D., & Jaakkola, T. (2018). On the robustness of interpretability methods. In *Proceedings of the ICML Workshop on Human Interpretability in Machine Learning*.

Asrzad, A., Li, X [Xiaobai], & Sarkar, S. (2025). Risk-sensitive counterfactual explanations for AI model predictions. In *Proceedings of the 45th International Conference on Information Systems*.

Barredo Arrieta, A., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., Garcia, S., Gil-Lopez, S., Molina, D., Benjamins, R., Chatila, R., & Herrera, F. (2020). Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion*, *58*, 82–115. https://doi.org/10.1016/j.inffus.2019.12.012

Bartley, K. (2025). *Big data statistics: How much data is there in the world.* Rivery. https://rivery.io/blog/big-data-statistics-how-much-data-is-there-in-the-world/

Batini, C., Cappiello, C., Francalanci, C., & Maurino, A. (2009). Methodologies for data quality assessment and improvement. *ACM Computing Surveys*, *41*(3), 1–52. https://doi.org/10.1145/1541880.1541883

Beck, R., Rai, A., Fischbach, K., & Keil, M. (2015). Untangling knowledge creation and knowledge integration in enterprise wikis. *Journal of Business Economics*, *85*(4), 389–420. https://doi.org/10.1007/s11573-014-0760-2

Bender, E. M., Gebru, T., McMillan-Major, A., & Shmitchell, S. (2021). On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*.

Berente, N., Gu, B., Recker, J., & Santhanam, R. (2021). Managing artificial intelligence. *MIS Quarterly*, *45*(3), 1433–1450. https://doi.org/10.25300/MISQ/2021/16274

Berger, J. O. (1985). *Statistical decision theory and bayesian analysis* (Second Edition). Springer. https://doi.org/10.1007/978-1-4757-4286-2

Bhatt, U., Antorán, J., Zhang, Y [Yunfeng], Liao, Q. V., Sattigeri, P., Fogliato, R., Melançon, G., Krishnan, R., Stanley, J., Tickoo, O., Nachman, L., Chunara, R., Srikumar, M., Weller, A., & Xiang, A. (2021). Uncertainty as a form of transparency: Measuring, communicating, and using uncertainty. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*.

Bhatti, Z. A., Baile, S., & Yasin, H. M. (2018). Assessing enterprise wiki success from the perspective of end-users: An empirical approach. *Behaviour & Information Technology*, *37*(12), 1177–1193. https://doi.org/10.1080/0144929X.2018.1488992

Brasse, J., Broder, H. R., Förster, M., Klier, M., & Sigler, I. (2023). Explainable artificial intelligence in information systems: A review of the status quo and future research directions. *Electronic Markets*, *33*(1), 1–30. https://doi.org/10.1007/s12525-023-00644-5

Brayne, S. (2017). Big data surveillance: The case of policing. *American Sociological Review*, *82*(5), 977–1008. https://doi.org/10.1177/0003122417725865

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5–32. https://doi.org/10.1023/A:1010933404324

Broughton, J. (2008). *Wikipedia: The missing manual* (1st ed.). O'Reilly.

Brynjolfsson, E., Li, D., & Raymond, L. (2025). Generative AI at work. *The Quarterly Journal of Economics*, *140*(2), 889–942. https://doi.org/10.1093/qje/qjae044

Buller, R. (2025). *The untapped power of unstructured data in enterprise AI.* Forbes Technology Council. https://www.forbes.com/councils/forbestechcouncil/2025/11/24/the-untapped-power-of-unstructured-data-in-enterprise-ai/

Cardellicchio, A., Nitti, M., Patruno, C., Mosca, N., Di Summa, M., Stella, E., & Renò, V. (2024). Automatic quality control of aluminium parts welds based on 3D data and artificial intelligence. *Journal of Intelligent Manufacturing*, *35*(4), 1629–1648. https://doi.org/10.1007/s10845-023-02124-1

Chiaburu, T., Haußer, F., & Bießmann, F. (2024). Uncertainty in XAI: Human perception and modeling approaches. *Machine Learning and Knowledge Extraction*, *6*(2), 1170–1192. https://doi.org/10.3390/make6020055

Cichy, C., & Rass, S. (2019). An overview of data quality frameworks. *IEEE Access*, *7*, 24634–24648. https://doi.org/10.1109/ACCESS.2019.2899751

Côté, P.-O., Nikanjam, A., Ahmed, N., Humeniuk, D., & Khomh, F. (2024). Data cleaning and machine learning: A systematic literature review. *Automated Software Engineering*, *31*(2), 54. https://doi.org/10.1007/s10515-024-00453-w

Council of Europe. (2024). *Council of Europe framework convention on artificial intelligence and human rights, democracy and the rule of law.* Council of Europe Treaty Office. https://www.coe.int/en/web/conventions/full-list?module=treaty-detail&treatynum=225

Council of Europe. (2026). *Chart of signatures and ratifications of Treaty 225.* Council of Europe Treaty Office. https://www.coe.int/en/web/Conventions/full-list/?module=signatures-by-treaty&treatynum=225

Croce, F., Andriushchenko, M., & Hein, M. (2019). Provable robustness of ReLU networks via maximization of linear regions. In *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*.

Cui, Z., Demirer, M., Jaffe, S., Musolff, L., Peng, S., & Salz, T. (2024). *The effects of generative AI on high skilled work: Evidence from three field experiments with software developers.* https://doi.org/10.2139/ssrn.4945566

D'Amour, A., Heller, K., Moldovan, D., Adlam, B., Alipanahi, B., Beutel, A., Chen, C., Deaton, J., Eisenstein, J., Hoffman, M. D., Hormozdiari, F., Houlsby, N., Hou, S., Jerfel, G., Karthikesalingam, A., Lucic, M., Ma, Y., McLean, C., Mincu, D., . . . Sculley, d. (2022). Underspecification presents challenges for credibility in modern machine learning. *Journal of Machine Learning Research*, *23*(226), 1–61.

Deck, L., Schoeffer, J., De-Arteaga, M., & Kühl, N. (2024). A critical survey on fairness benefits of explainable AI. In *Proceedings of the 2024 ACM Conference on Fairness, Accountability, and Transparency*.

Decker, T., Bhattarai, A. R., Gu, J., Tresp, V., & Buettner, F. (2024). *Provably better explanations with optimized aggregation of feature attributions*. arXiv:2406.05090 [cs.LG]. https://doi.org/10.48550/arXiv.2406.05090

Dell'Acqua, F., McFowland III, E., Mollick, E. R., Lifshitz-Assaf, H., Kellogg, K., Rajendran, S., Krayer, L., Candelon, F., & Lakhani, K. R. (2023). Navigating the jagged technological frontier: Field experimental evidence of the effects of AI on knowledge worker productivity and quality. *Harvard Business School Technology & Operations Mgt. Unit Working Paper*, Article 24-013.

Der Kiureghian, A., & Ditlevsen, O. (2009). Aleatory or epistemic? Does it matter? *Structural Safety*, *31*(2), 105–112. https://doi.org/10.1016/j.strusafe.2008.06.020

Dombrowski, A.-K., Alber, M., Anders, C., Ackermann, M., Müller, K.-R., & Kessel, P. (2019). Explanations can be manipulated and geometry is to blame. In *Advances in Neural Information Processing Systems*.

European Commission. (2025). *EU launches InvestAI initiative to mobilise €200 billion of investment in artificial intelligence*. https://ec.europa.eu/commission/presscorner/detail/en/ip_25_467

European Parliament. (2025). *The EU Artificial Intelligence Act*. https://artificialintelligenceact.eu/

Fernández-Loría, C., Provost, F., & Han, X. (2022). Explaining data-driven decisions made by AI systems: The counterfactual approach. *MIS Quarterly*, *46*(3), 1635–1660. https://doi.org/10.25300/MISQ/2022/16749

Fisher, A., Rudin, C., & Dominici, F. (2019). All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously. *Journal of Machine Learning Research*, *20*(177), 1–81.

Förster, M., Hagn, M., Hambauer, N., Jaki, P., Obermeier, A., Pinski, M., Schauer, A., Schiller, A., Benlian, A., Heinrich, B., Jussupow, E., Klier, M., Kraus, M., & Schnurr, D. (2025). A taxonomy for uncertainty-aware explainable AI. In *Proceedings of the 33rd European Conference on Information Systems*.

Förster, M., Hagn, M., Hambauer, N., Jaki, P., Obermeier, A., Schauer, A., Schiller, A., Wohlschlegel, J., Benlian, A., Heinrich, B., Jussupow, E., Klier, M., Kraus, M., & Schnurr, D. (2026). Understanding uncertainties in AI: A systematic literature review and research agenda. In *Proceedings of the 34th European Conference on Information Systems*.

Fügener, A., Grahl, J., Gupta, A., & Ketter, W. (2022). Cognitive challenges in human–artificial intelligence collaboration: Investigating the path toward productive delegation. *Information Systems Research*, *33*(2), 678–696. https://doi.org/10.1287/isre.2021.1079

Gal, Y. (2016). *Uncertainty in deep learning* [PhD dissertation]. University of Cambridge, Cambridge.

Gal, Y., & Ghahramani, Z. (2016). Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of the 33rd International Conference on Machine Learning*.

Gast, J., & Roth, S. (2018). Lightweight probabilistic deep networks. In *Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Gawlikowski, J., Tassi, C. R. N., Ali, M., Lee, J [Jongseok], Humt, M., Feng, J., Kruspe, A., Triebel, R., Jung, P., Roscher, R., Shahzad, M., Yang, W., Bamler, R., & Zhu, X. X. (2023). A survey of uncertainty in deep neural networks. *Artificial Intelligence Review*, *56*(1), 1513–1589. https://doi.org/10.1007/s10462-023-10562-9

Geiger, R. S., Cope, D., Ip, J., Lotosh, M., Shah, A., Weng, J., & Tang, R. (2021). "Garbage in, garbage out" revisited: What do machine learning application papers report about human-labeled training data? *Quantitative Science Studies*, *2*(3), 795–827. https://doi.org/10.1162/qss_a_00144

Gilpin, L. H., Bau, D., Yuan, B. Z., Bajwa, A., Specter, M., & Kagal, L. (2018). Explaining explanations: An overview of iterpretability of machine learning. In *Proceedings of the IEEE 5th International Conference on Data Science and Advanced Analytics*.

Gong, Y., Liu, G., Xue, Y., Li, R., & Meng, L. (2023). A survey on dataset quality in machine learning. *Information and Software Technology*, *162*, 107268. https://doi.org/10.1016/j.infsof.2023.107268

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press.

Gosiewska, A., & Biecek, P. (2019). *iBreakDown: Uncertainty of model explanations for non-additive predictive models*. arXiv:1903.11420v1 [cs.LG]. https://doi.org/10.48550/arXiv.2406.05090

Grauslund, J. (2022). Diabetic retinopathy screening in the emerging era of artificial intelligence. *Diabetologia*, *65*(9), 1415–1423. https://doi.org/10.1007/s00125-022-05727-0

Gudivada, V. N., Apon, A., & Ding, J. (2017). Data quality considerations for big data and machine learning: Going beyond data cleaning and transformations. *International Journal on Advances in Software*, *10*(1), 1–20.

Guidotti, R., Monreale, A., Ruggieri, S., Naretto, F., Turini, F., Pedreschi, D., & Giannotti, F. (2024). Stable and actionable explanations of black-box models through factual and counterfactual rules. *Data Mining and Knowledge Discovery*, *38*(5), 2825–2862. https://doi.org/10.1007/s10618-022-00878-5

Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., & Pedreschi, D. (2018). A survey of methods for explaining black box models. *ACM Computing Surveys*, *51*(5), 1–42. https://doi.org/10.1145/3236009

Guillen-Aguinaga, M., Aguinaga-Ontoso, E., Guillen-Aguinaga, L., Guillen-Grima, F., & Aguinaga-Ontoso, I. (2025). Data quality in the age of AI: A review of governance, ethics, and the FAIR principles. *Data*, *10*(12). https://doi.org/10.3390/data10120201

Guo, T., Bardhan, I. R., Ding, Y., & Zhang, S. (2025). An explainable artificial intelligence approach using graph learning to predict intensive care unit length of stay. *Information Systems Research*, *36*(3), 1478–1501. https://doi.org/10.1287/isre.2023.0029

Gupta, N., Mujumdar, S., Patel, H., Masuda, S., Panwar, N., Bandyopadhyay, S., Mehta, S., Guttula, S., Afzal, S., Sharma Mittal, R., & Munigala, V. (2021). Data quality for machine learning tasks. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.

Haley, P., & Burrell, D. N. (2025). Using artificial intelligence in law enforcement and policing to improve public health and safety. *Law, Economics and Society*, *1*(1), 46-59. https://doi.org/10.30560/les.v1n1p46

Hanin, B., & Rolnick, D. (2019). Complexity of linear regions in deep networks. In *Proceedings of the 36th International Conference on Machine Learning*.

Haug, A. (2021). Understanding the differences across data quality classifications: A literature review and guidelines for future research. *Industrial Management and Data Systems*, *121*(12), 2651–2671. https://doi.org/10.1108/IMDS-12-2020-0756

Haug, A. (2025). Intrinsic data quality dimensions: Expanding on Wand and Wang's data quality model. *Industrial Management & Data Systems*, *125*(1), 238–261. https://doi.org/10.1108/IMDS-02-2024-0100

He, W., & Yang, L. (2016). Using wikis in team collaboration: A media capability perspective. *Information & Management*, *53*(7), 846–856. https://doi.org/10.1016/j.im.2016.06.009

Heinrich, B., & Klier, M. (2015). Metric-based data quality assessment — Developing and evaluating a probability-based currency metric. *Decision Support Systems*, *72*, 82–96. https://doi.org/10.1016/j.dss.2015.02.009

Heinrich, B., Klier, M., Obermeier, A., & Schiller, A. (2025). Different but the same? An event-driven approach to determine probabilities of data duplication. *MIS Quarterly*, *49*(4), 1539–1566. https://doi.org/10.25300/MISQ/2025/18178

Heinrich, B., Klier, M., Schiller, A., & Wagner, G. (2018). Assessing data quality – A probability-based metric for semantic consistency. *Decision Support Systems*, *110*, 95–106. https://doi.org/10.1016/j.dss.2018.03.011

Hiniduma, K., Byna, S., & Bez, J. L. (2025). Data readiness for AI: A 360-degree survey. *ACM Computing Surveys*, *57*(9). https://doi.org/10.1145/3722214

Hora, S. C. (1996). Aleatory and epistemic uncertainty in probability elicitation with an example from hazardous waste management. *Reliability Engineering & System Safety*, *54*(2), 217–223. https://doi.org/10.1016/S0951-8320(96)00077-4

Huang, X., & Marques-Silva, J. (2024). On the failings of Shapley values for explainability. *International Journal of Approximate Reasoning*, *171*, 109112. https://doi.org/10.1016/j.ijar.2023.109112

Hüllermeier, E., & Waegeman, W. (2021). Aleatoric and epistemic uncertainty in machine learning: an introduction to concepts and methods. *Machine Learning*, *110*(3), 457–506. https://doi.org/10.1007/s10994-021-05946-3

Hurlin, C., Pérignon, C., & Saurin, S. (2024). The fairness of credit scoring models. *Management Science*, *72*(1), 406–425. https://doi.org/10.1287/mnsc.2022.03888

Huynh, M.-T. (2024). Using generative AI as decision-support tools: Unraveling users' trust and AI appreciation. *Journal of Decision Systems*, 1–32. https://doi.org/10.1080/12460125.2024.2428166

Jaki, P., Jussupow, E., & Benlian, A. (2025). Uncertainty-awareness in AI-augmented decision-making: Explaining appropriate reliance through cognitive mechanisms. In *Proceedings of the 46th International Conference on Information Systems*.

James, L. (2025). *Taiwan's $3.2 billion plan for 'AI island' with data centers, quantum hubs, and AI robotics labs faces risks — Power and geopolitical headwinds may threaten country's ambition*. https://www.tomshardware.com/tech-industry/semiconductors/taiwan-to-spend-billions-on-ai-island-push-despite-grid-risks

Jansen, M., Nguyen, H. Q., & Shams, A. (2025). Rise of the machines: The impact of automated underwriting. *Management Science*, *71*(2), 955–975. https://doi.org/10.1287/mnsc.2024.4986

Jayawardene, V., Sadiq, S., & Indulska, M. (2013). The curse of dimensionality in data quality. In *Proceedings of the 24th Australasian Conference on Information Systems*.

Jehangir, B., Radhakrishnan, S., & Agarwal, R. (2023). A survey on named entity recognition — Datasets, tools, and methodologies. *Natural Language Processing Journal*, *3*, 100017. https://doi.org/10.1016/j.nlp.2023.100017

Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y. J., Madotto, A., & Fung, P. (2023). Survey of hallucination in natural language generation. *ACM Computing Surveys*, *55*(12). https://doi.org/10.1145/3571730

Jia, Y., Bailey, J., Ramamohanarao, K., Leckie, C., & Houle, M. E. (2019). Improving the quality of explanations with local embedding perturbations. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Jiang, J., Zhou, K., Dong, Z., Ye, K., Zhao, W. X., & Wen, J. (2023). StructGPT: A general framework for large language model to reason over structured data. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

Jiang, S [Shengli], Qin, S., Pulsipher, J. L., & Zavala, V. M. (2024). Chapter 3 - Convolutional neural networks: Basic concepts and applications in manufacturing. In M. Soroush & R. D Braatz (Eds.), *Artificial intelligence in manufacturing* (pp. 63–102). Academic Press. https://doi.org/10.1016/B978-0-323-99134-6.00007-4

Jin, J., Dundar, A., & Culurciello, E. (2015). *Robust Convolutional Neural Networks under Adversarial Noise*. arXiv:1511.06306 [cs.LG]. https://doi.org/10.48550/arXiv.1511.06306

Jordan, M., Lewis, J., & Dimakis, A. G. (2019). Provable certificates for adversarial examples: Fitting a ball in the union of polytopes. In *Advances in Neural Information Processing Systems*.

Jussupow, E., Spohrer, K., & Heinzl, A. (2022). Radiologists' usage of diagnostic AI systems. *Business & Information Systems Engineering*, *64*(3), 293–309. https://doi.org/10.1007/s12599-022-00750-2

Karakurt, E., & Akbulut, A. (2026). Retrieval-augmented generation (RAG) and large language models (LLMs) for enterprise knowledge management and document automation: A systematic literature review. *Applied Sciences*, *16*(1), 368. https://doi.org/10.3390/app16010368

Kendall, A., & Gal, Y. (2017). What uncertainties do we need in bayesian deep learning for computer vision? In *Advances in Neural Information Processing Systems*.

Kim, B., Srinivasan, K., Kong, S. H., Kim, J. H., Shin, C. S., & Ram, S. (2023). ROLEX: A novel method for interpretable machine learning using robust local explanations, *47*(3), 1303–1332. https://doi.org/10.25300/MISQ/2022/17141

Kläs, M., & Vollmer, A. M. (2018). Uncertainty in machine learning applications: A practice-driven classification of uncertainty. In *Proceedings of the International Conference on Computer Safety, Reliability, and Security*.

Klier, M., Moestue, L., Obermeier, A., & Widmann, T. (2021). Event-driven assessment of currency of wiki articles: A novel probability-based metric. In *Proceedings of the 42nd International Conference on Information Systems*.

Kochenderfer, M. J. (2015). *Decision making under uncertainty: Theory and application*. The MIT Press. https://doi.org/10.7551/mitpress/10187.001.0001

Kon, H., Lee, J [Jacob], & Wang, R. Y. (1993). *A process view of data quality*. MIT TDQM Research Program.

König, M., Bosman, A. W., Hoos, H. H., & van Rijn, J. N. (2024). Critically Assessing the state of the art in neural network verification. *Journal of Machine Learning Research*, *25*(12), 1–53.

Krantz, T., & Jonker, A. (2026). *A compounding threat: The true cost of poor data quality.* IBM. https://www.ibm.com/think/insights/cost-of-poor-data-quality

Krempl, S. (2025). *Wikipedia criticized for many errors and outdated information.* Heise. https://www.heise.de/en/news/Wikipedia-criticized-for-many-errors-and-outdated-information-10476153.html

Krishna, S., Han, T., Gu, A., Wu, S., Jabbari, S., & Lakkaraju, H. (2022). *The disagreement problem in explainable machine learning: A practitioner's perspective*. arXiv:2202.01602v4 [cs.LG]. https://doi.org/10.48550/arXiv.2202.01602

Kronblad, C., Essén, A., & Mähring, M. (2024). When justice is blind to algorithms: Multi-layered blackboxing of algorithmic decision-making in the public sector. *MIS Quarterly*, *48*(4), 1637–1662. https://doi.org/10.25300/MISQ/2024/18251

Kusner, M. J., Loftus, J., Russell, C., & Silva, R. (2017). Counterfactual fairness. In *Advances in Neural Information Processing Systems*.

L'Heureux, A., Grolinger, K., Elyamany, H. F., & Capretz, M. A. M. (2017). Machine learning with big data: Challenges and approaches. *IEEE Access*, *5*, 7776–7797. https://doi.org/10.1109/ACCESS.2017.2696365

Laberge, G., Pequignot, Y., Mathieu, A., Khomh, F., & Marchand, M. (2023). Partial order in chaos: Consensus on feature attributions in the Rashomon set. *Journal of Machine Learning Research*, *24*(364), 1–50.

Lakshminarayanan, B., Pritzel, A., & Blundell, C. (2017). Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems*.

Laugel, T., Renard, X., Lesot, M., Marsala, C., & Detyniecki, M. (2018). Defining locality for surrogates in post-hoc interpretablity. In *Proceedings of the ICML Workshop on Human Interpretability in Machine Learning*.

Laumer, S., Maier, C., & Weitzel, T. (2017). Information quality, user satisfaction, and the manifestation of workarounds: A qualitative and quantitative study of enterprise content management system users. *European Journal of Information Systems*, *26*(4), 333–360. https://doi.org/10.1057/s41303-016-0029-7

Lee, Y. W. (2003). Crafting rules: Context-reflective data quality problem solving. *Journal of Management Information Systems*, *20*(3), 93–119.

Lee, Y. W., Strong, D. M., Kahn, B. K., & Wang, R. Y. (2002). AIMQ: A methodology for information quality assessment. *Information & Management*, *40*(2), 133–146. https://doi.org/10.1016/S0378-7206(02)00043-5

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W., Rocktäschel, T., Riedel, S., & Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*.

Lewoniewski, W. (2017). Enrichment of information in multilingual Wikipedia based on quality analysis. In W. Abramowicz (Ed.), *Business Information Systems Workshops. BIS 2017. Lecture Notes in Business Information Processing* (pp. 216–227). Springer.

Lewoniewski, W. (2019). Measures for quality assessment of articles and infoboxes in multilingual Wikipedia. In W. Abramowicz & A. Paschke (Eds.), *Business Information Systems Workshops. BIS 2018. Lecture Notes in Business Information Processing* (pp. 619–633). Springer.

Li, C [Chunxiao], Wang, H., Jiang, S [Songtao], & Gu, B. (2024a). The effect of AI-enabled credit scoring on financial inclusion: Evidence from an underserved population of over one million. *MIS Quarterly*, *48*(4), 1803–1834. https://doi.org/10.25300/MISQ/2024/18340

Li, N., Qi, Y., Li, C [Chaoran], & Zhao, Z. (2024b). Active learning for data quality control: A survey. *ACM Journal of Data and Information Quality*, *16*(2), 1–45.

Little, R. J. A., & Rubin, D. B. (2020). *Statistical analysis with missing data* (3rd edition). Wiley. https://doi.org/10.1002/9781119482260

Liu, Y. A., & Li, W. (2026). Transforming knowledge management with AI: Leveraging retrieval-augmented generation (RAG) in business strategy. In *AI-Driven Knowledge Management Processes, Volume 1: Strategies for the Modern Business Landscape.* Emerald Publishing Limited. https://doi.org/10.1108/978-1-80592-391-620261008

Lu, Y. (2019). Artificial intelligence: a survey on evolution, models, applications and future trends. *Journal of Management Analytics*, *6*(1), 1–29. https://doi.org/10.1080/23270012.2019.1570365

Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*.

Madsen, A., Reddy, S., & Chandar, S. (2022). Post-hoc interpretability for neural NLP: A survey. *ACM Computing Surveys*, *55*(8), 1–42. https://doi.org/10.1145/3546577

Marx, C., Park, Y., Hasson, H., Wang, Y [Yuyang], Ermon, S., & Huan, L. (2023). But are you sure? An uncertainty-aware perspective on explainable AI. In *Proceedings of the 26th International Conference on Artificial Intelligence and Statistics*.

Maslej, N., Fattorini, L., Perrault, R., Gil, Y., Parli, V., Kariuki, N., Capstick, E., Reuel, A., Brynjolfsson, E., Etchemendy, J., Ligett, K., Lyons, T., Manyika, J., Niebles, J. C., Shoham, Y., Wald, R., Walsh, T., Hamrah, A., Santarlasci, L., . . . Oak, S. (2025). *Artificial intelligence index report 2025.* Stanford. https://doi.org/10.48550/arXiv.2504.07139

Mehbodniya, A., Lazar, A. J. P., Webber, J., Sharma, D. K., Jayagopalan, S., K, K., Singh, P., Rajan, R., Pandya, S., & Sengan, S. (2022). Fetal health classification from cardiotocographic data using machine learning. *Expert Systems*, *39*(6), e12899.

Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., & Galstyan, A. (2021). A survey on bias and fairness in machine learning. *ACM Computing Surveys*, *54*(6), 1–35. https://doi.org/10.1145/3457607

Milana, C., & Ashta, A. (2021). Artificial intelligence techniques in finance and financial markets: A survey of the literature. *Strategic Change*, *30*(3), 189–209. https://doi.org/10.1002/jsc.2403

Miller, R., Chan, S. H. M., Whelan, H., & Gregório, J. (2025). A comparison of data quality frameworks: A review. *Big Data and Cognitive Computing*, *9*(4). https://doi.org/10.3390/bdcc9040093

Miller, T. (2019). Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, *267*, 1–38. https://doi.org/10.1016/j.artint.2018.07.007

Mintarya, L. N., Halim, J. N., Angie, C., Achmad, S., & Kurniawan, A. (2023). Machine learning approaches in stock market prediction: A systematic literature review. *Procedia Computer Science*, *216*, 96–102. https://doi.org/10.1016/j.procs.2022.12.115

MLQ. (2026). *Big tech's $650-700 billion AI infrastructure push reshapes cash flow dynamics*. https://mlq.ai/news/big-techs-650-700-billion-ai-infrastructure-push-reshapes-cash-flow-dynamics/

Moestue, L. (2024). Currency assessment of wiki articles: A novel event-based, content-aware approach. In *Proceedings of the 32nd European Conference on Information Systems*.

Mohammed, S., Budach, L., Feuerpfeil, M., Ihde, N., Nathansen, A., Noack, N., Patzlaff, H., Naumann, F., & Harmouch, H. (2025). The effects of data quality on machine learning performance on tabular data. *Information Systems*, *132*, 102549. https://doi.org/10.1016/j.is.2025.102549

Montúfar, G., Pascanu, R., Cho, K., & Bengio, Y. (2014). On the number of linear regions of deep neural networks. In *Advances in Neural Information Processing Systems*.

Müller, L., Holstein, J., Bause, S., Satzger, G., & Kühl, N. (2025). Data quality challenges in retrieval-augmented generation. In *Proceedings of the 46th International Conference on Information Systems*.

Mumuni, A., & Mumuni, F. (2022). Data augmentation: A comprehensive survey of modern approaches. *Array*, *16*, 100258. https://doi.org/10.1016/j.array.2022.100258

Nauta, M., Trienes, J., Pathak, S., Nguyen, E., Peters, M., Schmitt, Y., Schlötterer, J., van Keulen, M., & Seifert, C. (2023). From anecdotal evidence to quantitative evaluation methods: A systematic review on evaluating explainable AI. *ACM Computing Surveys*, *55*(13s), 1–42. https://doi.org/10.1145/3583558

Neuhof, B., & Benjamini, Y. (2024). Confident feature ranking. In *Proceedings of the 27th International Conference on Artificial Intelligence and Statistics*.

Nuthalapati, S. (2024). *Generative AI And big data analytics: Transforming decision making for leaders.* Forbes Technology Council. https://www.forbes.com/councils/forbestech-council/2024/08/01/generative-ai-and-big-data-analytics-transforming-decision-making-for-leaders/

OECD. (2024). *Principles for trustworthy AI*. https://oecd.ai/en/ai-principles

Padmanabhan, B., Fang, X., Sahoo, N., & Burton-Jones, A. (2022). Editor's comments: Machine learning in information systems research. *MIS Quarterly*, *46*(1), iii–xix. https://doi.org/10.25300/MISQ/2022/461E1

Paleyes, A., Urma, R.-G., & Lawrence, N. D. (2022). Challenges in deploying machine learning: A survey of case studies. *ACM Computing Surveys*, *55*(6). https://doi.org/10.1145/3533378

Parssian, A., Sarkar, S., & Jacob, V. S. (2004). Assessing data quality for information products: Impact of selection, projection, and cartesian product. *Management Science*, *50*(7), 967–982. https://doi.org/10.1287/mnsc.1040.0237

Pawlicki, M., Kozik, R., & Choraś, M. (2022). A survey on neural networks for (cyber-) security and (cyber-) security of neural networks. *Neurocomputing*, *500*, 1075–1087. https://doi.org/10.1016/j.neucom.2022.06.002

Pawlicki, M., Pawlicka, A., Uccello, F., Szelest, S., D'Antonio, S., Kozik, R., & Choraś, M. (2024). Evaluating the necessity of the multiple metrics for assessing explainable AI: A critical examination. *Neurocomputing*, *602*, 128282. https://doi.org/10.1016/j.neucom.2024.128282

Peng, J., Hahn, J., & Huang, K.-W. (2023). Handling missing values in information systems research: A review of methods and assumptions. *Information Systems Research*, *34*(1), 5–26. https://doi.org/10.1287/isre.2022.1104

Pertama Partners. (2026). *AI project failure statistics 2026: The complete picture*. https://www.pertamapartners.com/insights/ai-project-failure-statistics-2026

Pessach, D., & Shmueli, E. (2022). A review on fairness in machine learning. *ACM Computing Surveys*, *55*(3), 1–44. https://doi.org/10.1145/3494672

Petch, J., Di, S., & Nelson, W. (2022). Opening the black box: The promise and limitations of explainable machine learning in cardiology. *Canadian Journal of Cardiology*, *38*(2), 204–213. https://doi.org/10.1016/j.cjca.2021.09.004

Pinski, M., Adam, M., & Benlian, A. (2023). AI knowledge: Improving AI delegation through human enablement. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*.

Pirie, C., Wiratunga, N., Wijekoon, A., & Moreno-Garcia, C. F. (2023). AGREE: A feature attribution aggregation framework to address explainer disagreements with alignment metrics. In *CEUR Workshop Proceedings*.

Price, R., & Shanks, G. (2005). A semiotic information quality framework: Development and comparative analysis. *Journal of Information Technology*, *20*(2), 88–102. https://doi.org/10.1057/palgrave.jit.2000038

Priestley, M., O'donnell, F., & Simperl, E. (2023). A Survey of data quality requirements that matter in ML development pipelines. *ACM Journal of Data and Information Quality*, *15*(2). https://doi.org/10.1145/3592616

Raaijmakers, S. (2019). Artificial intelligence for law enforcement: Challenges and opportunities. *IEEE Security & Privacy*, *17*(5), 74–77. https://doi.org/10.1109/MSEC.2019.2925649

Rabonato, R. T., & Berton, L. (2025). A systematic review of fairness in machine learning. *AI and Ethics*, *5*(3), 1943–1954. https://doi.org/10.1007/s43681-024-00577-5

Raghu, M., Poole, B., Kleinberg, J., Ganguli, S., & Sohl-Dickstein, J. (2017). On the expressive power of deep neural networks. In *Proceedings of the 34th International Conference on Machine Learning*.

Razzaq, K., & Shah, M. (2025). Machine learning and deep learning paradigms: From techniques to practical applications and research frontiers. *Computers*, *14*(3). https://doi.org/10.3390/computers14030093

Redman, T. C. (1996). *Data quality for the information age*. Artech House.

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why should I trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Ross, A. S., Hughes, M. C., & Doshi-Velez, F. (2017). Right for the right reasons: Training differentiable models by constraining their explanations. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*.

Roxburgh, N., Stringer, L. C., Evans, A. J., Williams, T. G., & Müller, B. (2022). Wikis as collaborative knowledge management tools in socio-environmental modelling studies. *Environmental Modelling & Software*, *158*, 105538. https://doi.org/10.1016/j.envsoft.2022.105538

Russell, S., & Norvig, P. (2021). *Artificial intelligence: A modern approach*. Pearson Education.

Saarela, M., & Geogieva, L. (2022). Robustness, stability, and fidelity of explanations for a deep skin cancer classification model. *Applied Sciences*, *12*(19), 9545. https://doi.org/10.3390/app12199545

Saha Roy, R., Schlotthauer, J., Hinze, C., Foltyn, A., Hahn, L., & Kuech, F. (2025). Evidence contextualization and counterfactual attribution for conversational QA over heterogeneous data with RAG systems. In *Proceedings of the 18th ACM International Conference on Web Search and Data Mining*.

Salih, A. M., Raisi-Estabragh, Z., Galazzo, I. B., Radeva, P., Petersen, S. E., Lekadir, K., & Menegaz, G. (2025). A perspective on explainable artificial intelligence methods: SHAP and LIME. *Advanced Intelligent Systems*, *7*(1), 2400304. https://doi.org/10.1002/aisy.202400304

Sama. (2024). *Top 3 examples of how poor data quality impacts ML*. https://www.sama.com/blog/top-3-examples-of-how-poor-data-quality-impacts-ml

Savolainen, I., Ylinen, L., Grönroos, R., & Oksanen, A. (2026). AI transformation in working life: A systematic review of usage and attitudes towards AI among workers. *Digital Business*, *6*(1), 100162. https://doi.org/10.1016/j.digbus.2025.100162

Scannapieco, M., Missier, P., & Batini, C. (2005). Data quality at a glance. *Datenbank-Spektrum*, *14*, 6–14.

Schulz, J., Santos-Rodriguez, R., & Poyiadzi, R. (2022). Uncertainty quantification of surrogate explanations: An ordinal consensus approach. In *Proceedings of the Northern Lights Deep Learning Workshop*.

Senoner, J., Netland, T., & Feuerriegel, S. (2022). Using explainable artificial intelligence to improve process quality: Evidence from semiconductor manufacturing. *Management Science*, *68*(8), 5704–5723. https://doi.org/10.1287/mnsc.2021.4190

Settles, B. (2012). *Active learning*. Morgan & Claypool Publishers.

Shadbahr, T., Roberts, M., Stanczuk, J., Gilbey, J., Teare, P., Dittmer, S., Thorpe, M., Torné, R. V., Sala, E., Lió, P., Patel, M., Preller, J., Selby, I., Breger, A., Weir-McCall, J. R., Gkrania-Klotsas, E., Korhonen, A., Jefferson, E., Langs, G., . . . AIX-COVNET Collaboration (2023). The impact of imputation quality on machine learning classifiers for datasets with missing values. *Communications Medicine*, *3*(1), 139. https://doi.org/10.1038/s43856-023-00356-z

Shamala, P., Ahmad, R., Zolait, A., & Sedek, M. (2017). Integrating information quality dimensions into information security risk management (ISRM). *Journal of Information Security and Applications*, *36*, 1–10. https://doi.org/10.1016/j.jisa.2017.07.004

Shen, Y., & Zhang, X. (2024). The impact of artificial intelligence on employment: The role of virtual agglomeration. *Humanities and Social Sciences Communications*, *11*(1), 122. https://doi.org/10.1057/s41599-024-02647-9

Shumailov, I., Shumaylov, Z., Zhao, Y [Yiren], Gal, Y., Papernot, N., & Anderson, R. (2024). *The Curse of recursion: Training on generated data makes models forget*. arXiv:2305.17493 [cs.LG]. https://doi.org/10.48550/arXiv.2305.17493

Sicari, S., Rizzardi, A., Miorandi, D., Cappiello, C., & Coen-Porisini, A. (2016). A secure and quality-aware prototypical architecture for the internet of things. *Information Systems*, *58*, 43–55. https://doi.org/10.1016/j.is.2016.02.003

Singh, C., Inala, J. P., Galley, M., Caruana, R., & Gao, J. (2024a). *Rethinking interpretability in the era of large language models*. arXiv:2402.01761v1 [cs.CL]. https://doi.org/10.48550/arXiv.2402.01761

Singh, N., Kapoor, A., & Soni, N. (2024b). A sociotechnical perspective for explicit unfairness mitigation techniques for algorithm fairness. *International Journal of Information Management Data Insights*, *4*(2), 100259. https://doi.org/10.1016/j.jjimei.2024.100259

Slack, D., Hilgard, S., Jia, E., Singh, S., & Lakkaraju, H. (2020). Fooling LIME and SHAP: Adversarial attacks on post hoc explanation methods. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*.

Slack, D., Hilgard, A., Singh, S., & Lakkaraju, H. (2021). Reliable post hoc explanations: Modeling uncertainty in explainability. In *Advances in Neural Information Processing Systems*.

Song, H., Kim, M., Park, D., Shin, Y., & Lee, J.-G. (2023). Learning From noisy labels with deep neural networks: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, *34*(11), 8135–8153. https://doi.org/10.1109/TNNLS.2022.3152527

Stelloh, T. (2024). *An AI tool used in thousands of criminal cases is facing legal challenges.* NBC News. https://www.nbcnews.com/news/crime-courts/ai-tool-used-thousands-criminal-cases-facing-legal-challenges-rcna149607

Strong, D. M., Lee, Y. W., & Wang, R. Y. (1997). Data quality in context. *Communications of the ACM*, *40*(5), 103–110. https://doi.org/10.1145/253769.253804

Sundararajan, M., Taly, A., & Yan, Q. (2017). Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning*.

Sunyaev, A., Benlian, A., Pfeiffer, J., Jussupow, E., Thiebes, S., Maedche, A., & Gawlitza, J. (2025). High-risk artificial intelligence. *Business & Information Systems Engineering*, *67*(6), 981–994. https://doi.org/10.1007/s12599-025-00942-6

Taha, K. (2025). Big data analytics in IoT, social media, NLP, and information security: trends, challenges, and applications. *Journal of Big Data*, *12*(1), 150. https://doi.org/10.1186/s40537-025-01192-9

Taleb, I., Serhani, M. A., & Dssouli, R. (2018). Big data quality assessment model for unstructured data. In *Proceedings of the 2018 International Conference on Innovations in Information Technology*.

Tian, C., Cheng, T [Tongtong], Peng, Z., Zuo, W., Tian, Y., Zhang, Q., Wang, F.-Y., & Zhang, D. (2025). A survey on deep learning fundamentals. *Artificial Intelligence Review*, *58*(12), 381. https://doi.org/10.1007/s10462-025-11368-7

Titensky, J. S., Jananthan, H., & Kepner, J. (2018). Uncertainty propagation in deep neural networks using Extended Kalman Filtering. In *Proceedings of the 2018 IEEE MIT Undergraduate Research Technology Conference*.

Tran, T., & Cao, T. H. (2013). Automatic detection of outdated information in Wikipedia infoboxes. *Research in Computing Science*, *70*(1), 211–222.

Truong, D. (2021). Estimating the impact of COVID-19 on air travel in the medium and long term using neural network and Monte Carlo simulation. *Journal of Air Transport Management*, *96*, 102126. https://doi.org/10.1016/j.jairtraman.2021.102126

University of Amsterdam. (2023). *The Dutch childcare benefit scandal shows that we need explainable AI rules*. https://www.uva.nl/en/shared-content/faculteiten/en/faculteit-der-rechtsgeleerdheid/news/2023/02/childcare-benefit-scandal-transparency.html?cb

van Stein, B., Raponi, E., Sadeghi, Z., Bouman, N., van Ham, R. C. H. J., & Bäck, T. (2022). A comparison of global sensitivity analysis methods for explainable AI with an application in genomic prediction. *IEEE Access*, *10*, 103364–103381. https://doi.org/10.1109/ACCESS.2022.3210175

Verma, P. R., Singh, N. P., Pantola, D., & Cheng, X. (2024a). Neural network developments: A detailed survey from static to dynamic models. *Computers and Electrical Engineering*, *120*, 109710. https://doi.org/10.1016/j.compeleceng.2024.109710

Verma, S., Boonsanong, V., Hoang, M., Hines, K., Dickerson, J., & Shah, C. (2024b). Counterfactual explanations and algorithmic recourses for machine learning: A review. *ACM Computing Surveys*, *56*(12). https://doi.org/10.1145/3677119

Vigneswaran, R. K., Vinayakumar, R., Soman, K. P., & Poornachandran, P. (2018). Evaluating shallow and deep neural networks for network intrusion detection systems in cyber security. In *Proceedings of the 9th International Conference on Computing, Communication and Networking Technologies*.

Visani, G., Bagli, E., Chesani, F., Poluzzi, A., & Capuzzo, D. (2022). Statistical stability indices for LIME: Obtaining reliable explanations for machine learning models. *Journal of the Operational Research Society*, *73*(1), 91–101. https://doi.org/10.1080/01605682.2020.1865846

Wand, Y., & Wang, R. Y. (1996). Anchoring data quality dimensions in ontological foundations. *Communications of the ACM*, *39*(11), 86–95. https://doi.org/10.1145/240455.240479

Wang, R. Y., & Strong, D. M. (1996). Beyond accuracy: What data quality means to data consumers. *Journal of Management Information Systems*, *12*(4), 5–33. https://doi.org/10.1080/07421222.1996.11518099

Wang, S [Shiqi], Zhang, H., Xu, K., Lin, X., Jana, S., Hsieh, C.-J., & Kolter, J. Z. (2021). Beta-CROWN: Efficient bound propagation with per-neuron split constraints for neural network robustness verification. In *Advances in Neural Information Processing Systems*.

Wang, X [Xinwei], Feng, Y., Wang, S [Sutong], Wang, D., & Cheng, T [T.C.E.] (2025). An explainable lesion detection transformer model for medical imaging diagnosis decision support: Design science research. *Decision Support Systems*, *196*, 114492. https://doi.org/10.1016/j.dss.2025.114492

Wang, Z., Fredrikson, M., & Datta, A. (2022). Robust models are more interpretable because attributions look normal. In *Proceedings of the 39th International Conference on Machine Learning*.

Wu, Y., Zhang, L., & Wu, X. (2019). Counterfactual fairness: Unidentification, bound and algorithm. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*.

Zak, Y., & Even, A. (2017). Development and evaluation of a continuous-time Markov chain model for detecting and handling data currency declines. *Decision Support Systems*, *103*, 82–93. https://doi.org/10.1016/j.dss.2017.09.006

Zha, D., Bhat, Z. P., Lai, K.-H., Yang, F., Jiang, Z., Zhong, S., & Hu, X. (2025). Data-centric artificial intelligence: A survey. *ACM Computing Surveys*, *57*(5). https://doi.org/10.1145/3711118

Zhang, B., & Shin, Y. C. (2021). An adaptive Gaussian mixture method for nonlinear uncertainty propagation in neural networks. *Neurocomputing*, *458*, 170–183. https://doi.org/10.1016/j.neucom.2021.06.007

Zhang, R., Indulska, M., & Sadiq, S. (2019a). Discovering data quality problems. *Business and Information Systems Engineering*, *61*(5), 575–593. https://doi.org/10.1007/s12599-019-00608-0

Zhang, W., Li, X [Xiang], Ma, H., Luo, Z., & Li, X [Xu] (2021a). Federated learning for machinery fault diagnosis with dynamic validation and self-supervision. *Knowledge-Based Systems*, *213*, 106679. https://doi.org/10.1016/j.knosys.2020.106679

Zhang, Y [Yu], Tiňo, P., Leonardis, A., & Tang, K. (2021b). A survey on neural network interpretability. *IEEE Transactions on Emerging Topics in Computational Intelligence*, *5*(5), 726–742. https://doi.org/10.1109/TETCI.2021.3100641

Zhang, Y [Yujia], Song, K., Sun, Y., Tan, S., & Udell, M. (2019b). *"Why should you trust my explanation?" Understanding uncertainty in LIME explanations*. arXiv:1904.12991v2 [cs.LG]. https://doi.org/10.48550/arXiv.1904.12991

Zhao, H., Chen, H., Yang, F., Liu, N., Deng, H., Cai, H., Wang, S [Shuaiqiang], Yin, D., & Du, M. (2024). Explainability for large language models: A survey. *ACM Transactions on Intelligent Systems and Technology*, *15*(2). https://doi.org/10.1145/3639372

Zhao, Y [Yuying], Wang, Y [Yu], & Derr, T. (2023). Fairness and explainability: Bridging the gap towards fair model explanations. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Zhou, Y., Booth, S., Ribeiro, M. T., & Shah, J. (2022). Do feature attribution methods correctly attribute features? In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Zhou, Z., Hooker, G., & Wang, F. (2021). S-LIME: Stabilized-LIME for model explanation. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*.

Zschech, P., Weinzierl, S., & Kraus, M. (2025). Inherently interpretable machine learning: A contrasting paradigm to post-hoc explainable AI. *Business and Information Systems Engineering.* Advance online publication. https://doi.org/10.1007/s12599-025-00964-0