EXPLORE: A Novel Method for Local Explanations

Completed Research Paper

Bernd Heinrich
University of Regensburg
Universitätsstr. 31, 93053 Regensburg
Bernd.Heinrich@ur.de

Thomas Krapf
University of Regensburg
Universitätsstr. 31, 93053 Regensburg
Thomas.Krapf@ur.de

Paul Miethaner
University of Regensburg
Universitätsstr. 31, 93053 Regensburg
Paul.Miethaner@ur.de

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

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.

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 $f: X \rightarrow Y$ 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.

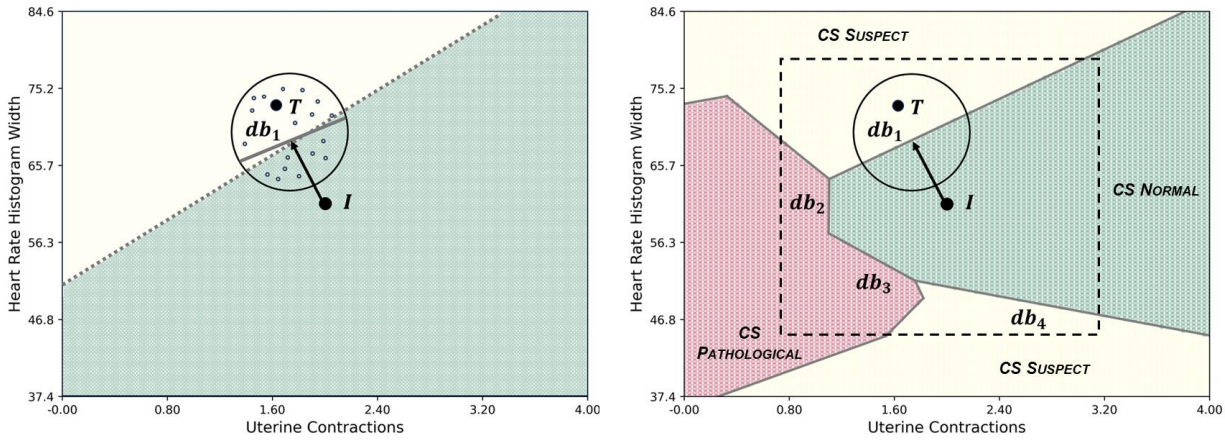

Figure 1: DBs Close to a Fetus Instance (Fetal Health Example)

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.

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.

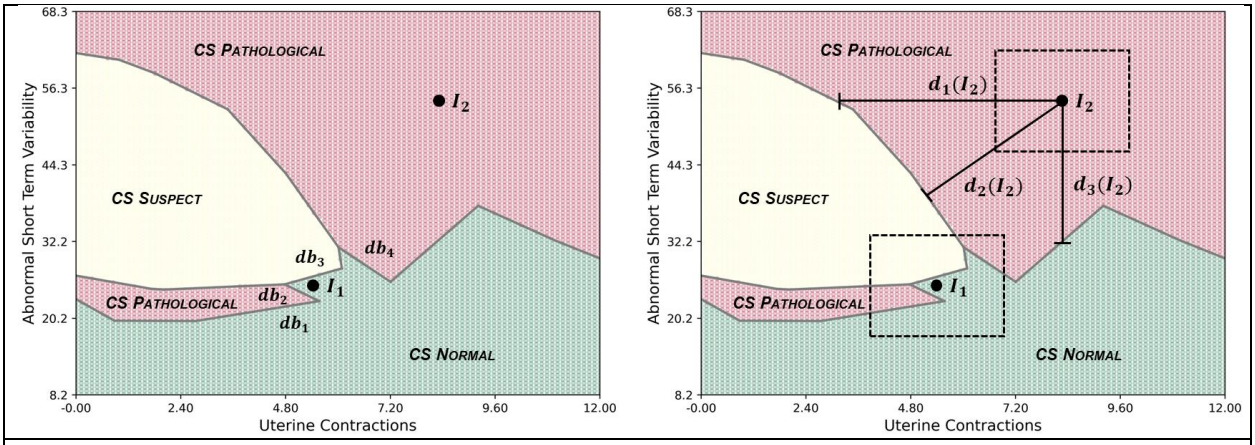

Figure 2. Reconstructing a ML model (Fetal Health Example)

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.

Explaining Machine Learning Models by Local Reconstruction

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.

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 describe 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 $k \neq i$)

$$\begin{gathered} v_{k,i} \cdot T_u \cdot x + v_{k,i} \cdot c_u \leq 0, \\ M_u x \leq l_u \end{gathered} \tag{6}$$

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.

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., $\Omega_x = \{x' \in X \mid |x'_i - x_i| \leq \varepsilon/2\,, \forall i = 1, \dots, m\}$. 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 $(\Omega_{\mathrm{x}}, \mathfrak{B}(\Omega_{\mathrm{x}}), P_{\mathrm{x}})$ 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.

$$p_i \coloneqq \int_{\Omega_{x,i}} p_{\mathrm{x}}(z)\, dz \tag{8}$$

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) $m_x := (p_1, \dots, p_n)$ 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:

$$d_{db}(x) = \min_{\mathrm{u,i,j}} dist(x, db_{u,i,j}) \tag{9}$$

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 $\overrightarrow{d_{db}}(x) := x_{db,min} - x$ 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 $1 \le e \le n$, we define

$$d^e_{db}(x) = \min_{\mathrm{u,i,j}} dist(x, db^{e,x}_{u,i,j}), \tag{10}$$

where $db^{e,x}_{u,i,j}$ 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, $db^{e,x}_{u,i,j}$ can be defined as $db^{e,x}_{u,i,j} = \{z \in db_{u,i,j} \mid \forall d \neq e : (z)_d = (x)_d\}$. Intuitively, $d^e_{db}(x)$ 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, $\Omega_{x,i} = (\mathrm{argmax}\, f)^{-1}(i) = \bigcup_{u=1}^{t} cs_{u,i}$ 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 $d^e_{db}(x)$ because this argument also holds for the (one-dimensional) restriction of the feature space $X^e = \{z \in X \mid \forall d \neq e : (z)_d = (x)_d\}$. □

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.

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.

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., $\exists z_1, z_2 \in Z: \text{argmax}\, f(z_1) \neq \text{argmax}\, f(z_2)$. 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.

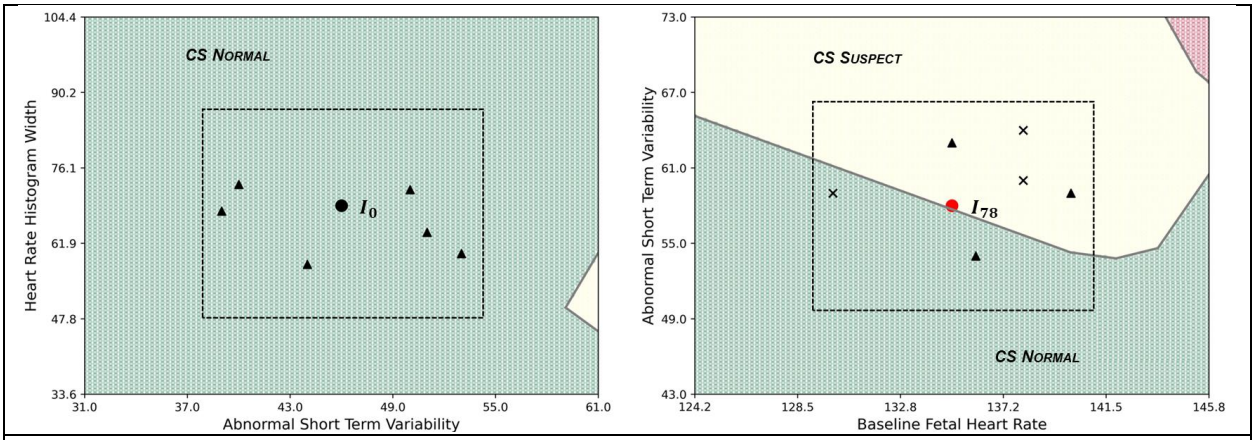

Figure 3. Sensitivity of the two Instances 0 and 78 from the Fetal Health Dataset (Ground Truth of Selected Training Instances: Class "NORMAL": Δ, Class "SUSPECT": x)

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.

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.

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.

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.