

# **Machine Learning and Deep Learning in Ecology – from predictions to mechanistic inference**



DISSERTATION ZUR ERLANGUNG DES  
DOKTORGRADES DER NATURWISSENSCHAFTEN (DR.RER.NAT.)  
DER FAKULTÄT FÜR BIOLOGIE UND VORKLINISCHE MEDIZIN  
DER UNIVERSITÄT REGENSBURG

vorgelegt von

Maximilian Matthias Pichler

aus

Mindelheim

im Jahr

2024



# **Machine Learning and Deep Learning in Ecology – from predictions to mechanistic inference**



DISSERTATION ZUR ERLANGUNG DES  
DOKTORGRADES DER NATURWISSENSCHAFTEN (DR.RER.NAT.)  
DER FAKULTÄT FÜR BIOLOGIE UND VORKLINISCHE MEDIZIN  
DER UNIVERSITÄT REGENSBURG

vorgelegt von

Maximilian Matthias Pichler

aus

Mindelheim

im Jahr

2024

Das Promotionsgesuch wurde eingereicht am: 10. Januar 2024

Die Arbeit wurde angeleitet von:  
**Prof. Dr. Florian Hartig**

Unterschrift:



# Acknowledgements

This doctoral thesis has been an incredibly enjoyable experience. I probably owe this to the most important factor for the progress and success of a doctoral thesis, namely the environment, so I would like to take this opportunity to thank a few people.

First of all, I would like to thank my supervisor, Prof. Dr. Florian Hartig. Thank you for making this possible, for the endless discussions, the freedom you gave me and the always open door. I don't think we had a single official appointment in all those years (I hope I wasn't too annoying).

Thanks also to my work group and all my (former) colleagues Lisa, Lena and Melina. Special thanks to Lukas and Johannes, I will never forget our time together in the office.

I would also like to thank my friends and family for their support during the whole time.

Last but not least, I would like to thank my partner Sandra for always supporting me!



# Summary

Data analysis is a central component of modern ecology to advance our knowledge of nature. For many decades statistical models have been the backbone of data analysis in ecology and evolution (E&E). However, traditional statistical models are unable to cope with the complexity of ecological patterns and the increasing dimensionality of ecological data. Promising solutions to these challenges are offered by machine learning (ML), and deep learning (DL) algorithms. Unlike statistical models, ML and DL algorithms adjust their complexity data-dependent and provide highly optimized frameworks. But while DL algorithms have already achieved remarkable success in data annotation, the practical value of ML and DL algorithms for data analysis in E&E is still unclear. Their opaque algorithmic nature, limited interpretability, and their purpose for predictive modeling raise doubts about their suitability for inference which is crucial for E&E. To address these challenges, we ask the following questions: What are the underlying concepts of ML and DL algorithms? Can we use ML and DL algorithms to infer ecological effects? How reliable would be this inference? Can ML and DL algorithms be used for statistical computing?

The **first chapter** introduces the challenges of ecological data with statistical models and how ML and DL algorithms can provide solutions. In the **second chapter** we explain the principles of ML and DL algorithms such as their ability to automatically adjust their complexity and review their current role in E&E. We found that ML and DL algorithms are mostly used for data annotation and predictions, but they could also be used for inference. In the **third chapter** we investigate whether ML and DL algorithms can be used to infer complex ecological pattern. We found that ML and DL algorithms can successfully infer trait-matching plant-pollinator networks, better than statistical models. In the **fourth chapter** we study how computational statistics can benefit from DL frameworks. We found that the DL framework PyTorch allows joint species distribution models (JSDMs) to scale exceptionally well with the number of species. In addition, we found that this approach improves the accuracy of the estimates compared to other approaches. We conclude that DL frameworks can overcome computational bottlenecks in statistical models. In the **fifth chapter** we investigate the reliability of inference with ML and DL algorithms, in particular whether they can distinguish causal from correlative patterns. We found that most ML and DL algorithms are subject to the bias-variance tradeoff, but some algorithms (e.g., neural networks) have lower biases than others. We conclude that reliable inference with ML and DL algorithms depends on the algorithm, the hyperparameters, and the data. In the **sixth chapter** we present a new R package, 'cito' with a user-friendly interface for fitting deep neural networks based on torch, including uncertainty intervals for all outputs (e.g., predictions, xAI metrics). In the **seventh chapter** we found that ML algorithms can improve seed bank persistence predictions over statistical models. In the **eighth chapter**, we investigate how many levels are needed to reliably estimate random effects in mixed effects models. We found that five levels can be sufficient but in the case of a singular fit, switching to a fixed effects model can prevent an increase in the false positive rate. The final **ninth chapter** discusses the relevance of our studies to the question of whether and how ML and DL algorithms can support data analysis in E&E.

# Contribution statement

This thesis is based on seven manuscripts that are largely identical to those accepted or submitted for publication:

## Chapter 2

**Pichler, M.**, & Hartig, F. (2023). Machine learning and deep learning—A review for ecologists. *Methods in Ecology and Evolution*, 14, 994–1016. <https://doi.org/10.1111/2041-210X.14061>

**Author contribution:** **M.P.** and F.H. jointly conceived and designed the study. Both authors contributed equally to the writing and preparation of the manuscript.

## Chapter 3

**Pichler, M.**, Boreux, V, Klein, A-M, Schleuning, M, Hartig, F. (2020). Machine learning algorithms to infer trait-matching and predict species interactions in ecological networks. *Methods in Ecology and Evolution*, 11, 281–293. <https://doi.org/10.1111/2041-210X.13329>

**Author contribution:** **M.P.** and F.H. conceived the ideas and designed methodology; V.B., A.M.K. and M.S. provided data; **M.P.** performed the analyses. **M.P.** and F.H. wrote the first draft of the manuscript. All authors contributed critically to the completion and revision of the manuscript.

## Chapter 4

**Pichler, M.**, & Hartig, F. (2021). A new joint species distribution model for faster and more accurate inference of species associations from big community data. *Methods in Ecology and Evolution*, 12, 2159–2173. <https://doi.org/10.1111/2041-210X.13687>

**Author contribution:** F.H. and **M.P.** jointly conceived and designed the study; **M.P.** implemented the sjSDM software, ran the experiments and analysed the data. Both authors contributed equally to discussing and interpreting the results, and to the preparation of the manuscript.

## Chapter 5

**Pichler, M.**, & Hartig, F. (2023). Can predictive models be used for causal inference? *In Preparation*. Preprint available at <https://arxiv.org/abs/2306.10551>

**Author contribution:** **M.P.** and F.H. jointly conceived and designed the study. Both authors contributed equally to the writing and preparation of the manuscript.

## Chapter 6

Amesoder, C., Hartig, F., & **Pichler, M.** (2023). cito: An R package for training neural networks using torch Preprint available at <https://arxiv.org/abs/2303.09599>

**Author contribution:** C.A.: Conceptualization, Software, Methodology, Writing – Original Draft, Visualization, Investigation. F.H.: Conceptualization, Writing – Review & Editing, Visualization, Supervision, Methodology, Investigation. **M.P.:** Conceptualization, Software, Methodology, Writing – Review & Editing, Visualization, Supervision, Investigation.

## Chapter 7

Rosbakh, S., **Pichler, M.** & Poschlod, P. (2022) Machine-learning algorithms predict soil seed bank persistence from easily available traits. *Applied Vegetation Science*, 25, e12660. <https://doi.org/10.1111/avsc.12660>

**Author contribution:** S.R. and **M.P.** contributed equally to the study. S.R. conceived of

the study, **M.P.** conducted the analysis, and the first version of the manuscript was written by S.R. together with **M.P.** All authors contributed to manuscript editing.

## Chapter 8

Oberpriller, J., de Souza Leite, M., & **Pichler, M.** (2022). Fixed or random? On the reliability of mixed-effects models for a small number of levels in grouping variables. *Ecology and Evolution*, 12, e9062. <https://doi.org/10.1002/ece3.9062>

**Author contribution:** O.P. and **M.P.** contributed equally to the study. **M.P.**, J.O. and M.S.L. designed the study. **M.P.** and J.P. ran the simulations, analyzed the results and wrote a first draft. All authors contributed equally to revising the manuscript and interpreting and discussing results.

The following co-author manuscripts are **not included** but were produced in connection with this thesis:

Chalmandrier, L., Hartig, F., Laughlin, D. C., Lischke, H., **Pichler, M.**, Stouffer, D. B., & Pellissier, L. (2021). Linking functional traits and demography to model species-rich communities. *Nature Communications*, 12(1), 2724. <https://doi.org/10.1038/s41467-021-22630-1>

Li, Y., Devenish, C., Tosa, M., Luo, M., Bell, D., Lesmeister, D., Greenfield, P., **Pichler, M.**, Levi, T. & Yu, D. W. (2023). Combining environmental DNA and remote sensing for efficient, fine-scale mapping of arthropod biodiversity. In preparation. Preprint available at <https://doi.org/10.1101/2023.09.07.556488>.

Hartig, F., Abrego, N., Bush, A., Chase, J. M., Guillera-Arroita, G., Leibold, M. A., Ovaskainen, O., Pellissier, L., **Pichler, M.**, Poggiato, G., Pollock, L., Si-Moussi, S., Thuiller, W., Viana, D. S., Warton, D. I., Zurell, D., & Douglas, W. Y. (2024). Novel community data in ecology-properties and prospects. *Trends in Ecology & Evolution*. <https://doi.org/10.1016/j.tree.2023.09.017>

Cai, W., **Pichler, M.**, Biggs, J., Nicolet, P., Ewald, N., Griffiths, R. A., Bush, A., Leibold, M. A., Hartig, F., & Yu, D. W. (2023). Environmental DNA captures the internal structure of a pond metacommunity. In preparation. Preprint available at <https://doi.org/10.1101/2023.12.12.571176>

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	ML and DL algorithms automatically adjust their complexity . . . . .	2
1.2	From predictions to inference with ML and DL algorithms . . . . .	3
1.3	Leveraging ML and DL for computational statistics . . . . .	4
1.4	Research questions . . . . .	5
<b>2</b>	<b>Machine Learning and Deep Learning — A review for Ecologists</b>	<b>7</b>
2.1	Introduction . . . . .	8
2.2	History of ML and DL and its relation to statistics . . . . .	10
2.3	Important ML algorithms in more detail . . . . .	13
2.4	Why does ML work? . . . . .	18
2.5	Emerging trends in ML (in E&E) . . . . .	22
2.6	Conclusion . . . . .	28
<b>3</b>	<b>Machine learning algorithms to infer trait-matching and predict species interactions in ecological networks</b>	<b>31</b>
3.1	Introduction . . . . .	32
3.2	Material and Methods . . . . .	34
3.3	Results . . . . .	38
3.4	Discussion . . . . .	41
3.5	Conclusion . . . . .	44
<b>4</b>	<b>A new joint species distribution model for faster and more accurate inference of species associations from big community data</b>	<b>45</b>
4.1	Introduction . . . . .	46
4.2	Material and Methods . . . . .	48
4.3	Results . . . . .	53
4.4	Discussion . . . . .	57
4.5	Conclusions . . . . .	61
<b>5</b>	<b>Can predictive models be used for causal inference?</b>	<b>63</b>
5.1	Introduction . . . . .	64
5.2	Results . . . . .	67
5.3	Discussion . . . . .	73
5.4	Methods . . . . .	76
<b>6</b>	<b>cito: An R package for training neural networks using torch</b>	<b>79</b>
6.1	Introduction . . . . .	80
6.2	Design of the cito package . . . . .	81
6.3	Performance comparison and validation of cito . . . . .	82
6.4	Workflow and case study . . . . .	84
6.5	Conclusion . . . . .	87
<b>7</b>	<b>Machine-learning algorithms predict soil seed bank persistence from easily available traits</b>	<b>89</b>
7.1	Introduction . . . . .	90

7.2	Material and Methods . . . . .	92
7.3	Results . . . . .	96
7.4	Discussion . . . . .	100
<b>8</b>	<b>Fixed or random? On the reliability of mixed-effects models for a small number of levels in grouping variables</b>	<b>105</b>
8.1	Introduction . . . . .	106
8.2	Methods . . . . .	108
8.3	Results . . . . .	113
8.4	Discussion . . . . .	116
8.5	Conclusion . . . . .	121
<b>9</b>	<b>Discussion</b>	<b>123</b>
9.1	Discussion of the results . . . . .	123
9.2	Conclusion and future research . . . . .	126
<b>10</b>	<b>Appendix</b>	<b>129</b>
	Supporting Information S1 . . . . .	129
	Supporting Information S2 . . . . .	131
	Supporting Information S3 . . . . .	139
	Supporting Information S4 . . . . .	155
	Supporting Information S5 . . . . .	171
	Supporting Information S6 . . . . .	173
	<b>References</b>	<b>193</b>





---

## INTRODUCTION

---

A fundamental task of modern ecology is to unravel the processes underlying species distributions. Species are entangled in a web of interactions, locally with their environment, with other species, and at larger scales through dispersal and community dynamics (CHESSON, 2000; LEIBOLD, HOLYOAK, *et al.*, 2004). Research in ecology and evolution (E&E) is gradually assembling the fragments of this complex puzzle.

Network ecology seeks to understand when and why species interact. The distribution of the links (species interactions) within ecological networks can arise from random (e.g. due to different abundances) and deterministic causes such as trait-matching (chapter 3; PICHLER, BOREUX, *et al.*, 2020; EKLÖF *et al.*, 2013; BARTOMEUS *et al.*, 2016; DORMANN, FRÜND, and SCHAEFER, 2017). Trait-matching describes the phenomenon that two individuals of two species interact when their traits match in a favorable way (chapter 3). If trait-matching is the main driver of species interactions it could shed new light on the debate about pollination syndromes which postulates that plant and pollinator traits have co-evolved (OLLERTON *et al.*, 2009; ROSAS-GUERRERO *et al.*, 2014a). However, previous studies reported mixed results for trait-matching (EKLÖF *et al.*, 2013; BARTOMEUS *et al.*, 2016). While trait-matching signals were strong in food webs (IDALINE *et al.*, 2018), trait-matching in plant-pollinator networks only weakly predicted species interactions (BROUSSEAU, GRAVEL, and HANDA, 2018b; POMERANZ *et al.*, 2019).

The goal of community ecology is to understand how species communities are assembled. Species composition in communities is determined by four mechanisms, namely environmental filtering, interactions, ecological drift, and dispersal that connect local communities (LEIBOLD, HOLYOAK, *et al.*, 2004; LEIBOLD and CHASE, 2017). The four mechanisms are not mutually exclusive and contribute differently to a set of communities, the metacommunity (LEIBOLD, HOLYOAK, *et al.*, 2004; LEIBOLD and CHASE, 2017). Metacommunity theory attempts to empirically disentangle the contributions of these mechanisms, which is facilitated by the emergence of novel community data (HARTIG, ABREGO, *et al.*, 2024; LEIBOLD, RUDOLPH, *et al.*, 2022). Novel community data consist of many rows, many columns of data (many sites, many species) (HARTIG, ABREGO, *et al.*, 2024), potentially numbering in the hundreds or thousands of communities (e.g. CAI *et al.*, 2023). However, large numbers of species (and sites) lead to runtime limitations and parameter identifiability problems in community models (when the number of species exceeds the number of sites, chapter 4).

Both fields of E&E, as well as many other, rely on statistical models. Which is not surprising given that statistical models are the backbone of data analysis for structured data (e.g. see citations of ZUUR *et al.*, 2009 and BOLKER *et al.*, 2009). In structured (tabular) data, variables have usually a specific meaning and the advantage of statistical models is that they can describe the relationships between variables and the response in a comprehensive way, unlike many ML and DL algorithms

(chapter 2). Statistical models have undeniably advanced scientific progress in E&E over the past hundred years, but there are often two fundamental problems with statistical models that can impede progress.

First, nature is probably more complex than we can specify in statistical models which can lead to erroneous conclusions. Statistical models are often relatively rigid, unable to capture complex patterns such as trait-matching (chapter 3). However, the lack of flexibility of statistical models is only part of a larger problem. The fundamental challenge with statistical models is finding the "right" level of complexity. Oversimplification can lead to false negatives (e.g. trait-matching), and overly complex models can lead to false positives and low statistical power (FORSTMEIER, WAGENMAKERS, and PARKER, 2017). This is further complicated by the fact that the analyst must determine the complexity in statistical model such as the model structure or the functional form of the effects with severe consequences if misspecified (e.g. see the debate about JUNG *et al.*, 2014).

Techniques for automatically optimizing the complexity of statistical models have been discussed for a long time (JOHNSON and OMLAND, 2004). Approaches such as backward or forward model selection are used to select the best model (JOHNSON and OMLAND, 2004). Also several models could be averaged (BURNHAM and ANDERSON, 2004). However, these methods are criticized because they can mess with the "causality" of the models leading to inflated type I error rates and biased effect (LAUBACH *et al.*, 2021; ARIF and MACNEIL, 2022b). In addition, there is simply a combinatorial limit to how many different effect forms and interactions can be tested in statistical models.

Second, conventional statistical models are often not well equipped for high-dimensional data. For example, joint species distribution models (JSDM) emerged as novel community models to account for co-occurrences (POLLOCK *et al.*, 2014). JSDM model species occurrences jointly and account for co-occurrences that cannot be explained by the environment (associations). These associations are parametrized by a variance-covariance matrix. However, the variance-covariance matrix scales poorly with the number of species (almost quadratically, (WARTON *et al.*, 2015)), leading to runtime limitations for large community data (chapter 4; PICHLER and HARTIG, 2021a).

Machine learning (ML) and deep learning (DL) algorithms could provide solutions to these problems. In other areas of science, ML and DL algorithms advanced scientific challenges such as protein folding, weather prediction (LAM *et al.*, 2023), and antibiotic discovery (WONG *et al.*, 2023). In E&E, DL algorithms have become unparalleled tools at the frontier of automatic data annotation (chapter 2; (CHRISTIN, HERVET, and LECOMTE, 2019; PICHLER and HARTIG, 2023b)) because of their ability to identify biological structures (TABAK *et al.*, 2019; BERGLER *et al.*, 2019; GUIRADO *et al.*, 2018; WILLI *et al.*, 2019). ML and DL algorithms are considered to be exceptional predictive models due to properties that make them also interesting for data analysis of structured data in E&E (chapter 2).

## 1.1 ML and DL algorithms automatically adjust their complexity

ML and DL algorithms (e.g. boosted regression trees (BRT) and deep neural networks (DNN)) are known to be able to approximate any mathematical function (GOODFELLOW, BENGIO, and COURVILLE, 2016) with fewer a priori assumptions about the model structure than statistical models. ML and DL algorithms rely on an algorithmic approach instead of a mathematical model which gives them a high degree of flexibility (chapter 2). Hence, ML and DL algorithms promise

to detect complex ecological patterns that otherwise often elude rigid statistical models such as complex variable-variable interactions (chapter 3).

Since the high flexibility of ML and DL algorithms increases the risk of overfitting, they have been designed to automatically optimize the bias-variance tradeoff. Thus, ML and DL algorithms are expected to both capture the complexity of the real world and minimize the risk of overfitting by letting the data, rather than the analyst, determine the necessary model complexity (chapter 2, GOODFELLOW, BENGIO, and COURVILLE, 2016). As a result, ML and DL algorithms gained a reputation as excellent predictive models.

While predictions provide valuable insights into ecological processes, predictions are insufficient to explain the underlying mechanisms (patterns) that drive these processes. In principle, ML and DL algorithms promise to improve the detection of complex effects because a model that predicts better should also explain better (BREIMAN, 2001b). However, the interpretability of ML and DL algorithms is limited and it is difficult to extract meaningful effects from them (chapter 2).

## 1.2 From predictions to inference with ML and DL algorithms

Many ML and DL algorithms are considered non-explanatory for two reasons: Their interpretation is challenging and the reliability of their estimated effects is debated (SHMUELI, 2010). In the following, I will refer to explanation as inference which focuses on estimating effects from empirical data with models.

### 1.2.1 Discrepancy between predicting and explaining

It is said that the purpose of a model determines its application, either for inference or for prediction (TREDENNICK *et al.*, 2021; SHMUELI, 2010). Predictive models optimize the bias-variance tradeoff on predictions, while explanatory models optimize the explanatory power of a model, i.e. the reliability of the estimated effects. As a result predictive and explanatory models optimize different model structures especially with respect to collinearity.

Collinearity increases the variance (uncertainty) of all models, both in predictive and explanatory models. In explanatory models, however, collinearity is considered a "necessary evil" because it can be crucial for unbiased effect estimates (PEARL, 2009). Especially in observational studies, often models must be adjusted for collinear structures such as confounders by including them in the model (chapter 5, PEARL, 2009). Thus, much of explanatory modeling revolves around the choice of variables and the causal dependencies between variables.

From a predictive perspective, the shared (collinear) part of two collinear variables is redundant (no information gain) and just increases the variance. Hence, predictive models can improve their performance by reducing collinearity (DORMANN, ELITH, *et al.*, 2013). In statistical models, collinearity can be reduced by dropping collinear variables. ML and DL algorithms probably suppress collinearity as a result of their automatic complexity adjustment.

However, the exact details of how the complexity adjustments of ML and DL algorithms deal with collinearity are unclear, which is a problem if we want to use ML and DL algorithms for inference. For certain techniques in predictive modeling, such as regularization, we know that spillover can occur, i.e., the smaller of two collinear variables is overestimated (e.g., HOERL and KENNARD, 1970; chapter 5). However, ML and DL algorithms depend on regularization and other mechanisms

such as boosting and bagging, where to our knowledge it is mostly unknown how collinearity affects the final inference of ML and DL algorithms (chapter 5).

### 1.2.2 Making ML and DL algorithms interpretable

Another disadvantage of many ML and DL algorithms is that it is difficult to extract meaningful effects. Many ML and DL algorithms are no longer based on unambiguous mathematical equations (e.g. BREIMAN, 2001a) and thus variable-response relationships are no longer unambiguous. However, it has been recognized that there is a need to understand ML and DL models for reasons such as assessing shareholder confidence, finding flaws in the model, or detecting biases in the data being transferred to the models (TUVA *et al.*, 2022).

In response to this premise, the field of Explainable AI (xAI) emerged that allows to understand, mostly post hoc, how ML and DL models generate predictions (RYO *et al.*, 2021). For example, Friedmans H-statistic can be used to infer variable-variable interactions from fitted ML and DL models, to assess the importance of trait-matching in predicting plant-pollinator interactions (chapter 3).

But xAI tools were not designed as tool for inference. xAI tools lack statistical properties, and some of them are prone to data subtleties such as collinearity. Moreover, they simplify complex functions (the ML and DL algorithms) to make them interpretable, thus losing information during the process (RUDIN, 2019). Finally, they are sensitive to hyperparameters and can explain a prediction in different ways (RUDIN, 2019). Nevertheless, they come closest to our expectations of effects and make ML and DL algorithms interpretable (chapter 7, chapter 3).

## 1.3 Leveraging ML and DL for computational statistics

In addition to the inference of complex patterns, ML and DL frameworks could also make statistical models scalable. ML and DL algorithms are already computationally expensive by definition, a lot of effort has been invested in the development of optimized frameworks, in particular for DL algorithms (PASZKE, GROSS, CHINTALA, *et al.*, 2017; ABADI *et al.*, 2015). It seems therefore to be an obvious step to consider using these optimized frameworks for computational statistics.

State-of-the-art DL frameworks have interesting features for statistical models. Frameworks such as PyTorch or TensorFlow are essentially linear algebra libraries with the option of moving operations to hardware accelerators such as graphical processing units (GPUs). In addition, NN are optimized using back-propagation which requires gradients. For this, modern DL frameworks support automatic differentiation which automatically (analytically) derives gradients by tracking all operators with respect to parameters (e.g., NN weights). These features make them attractive for statistical models (e.g., NUTS, HOFFMAN, GELMAN, *et al.*, 2014) and could improve the scalability of statistical models.

However, simply reimplementing the statistical models in DL framework is insufficient. Other factors need to be considered to make efficient use of GPUs. GPUs have many, but weak, cores, and only operations that can be highly parallelized can take advantage of them. For example, in Markov-Chain-Monte-Carlo algorithms, the sampling cannot be well parallelized because the samples depend on each other (e.g. TER BRAAK and VRUGT, 2008; HOFFMAN, GELMAN, *et al.*, 2014). On the other hand, in a simple Monte Carlo approximation, the samples are independent, and the approximation could benefit greatly from GPU parallelization. The disadvantage of a naive MC approach is that there is a tradeoff between accuracy and dimensionality.

A potential application is the multivariate probit model (MVP) which is used in joint species distribution modeling (chapter 4; POLLOCK *et al.*, 2014). The MVP is based on the cumulative multivariate normal distribution which has no closed form for many dimensions. Common approximations of the MVP include Markov-Chain-Monte-Carlo (MCMC) sampling (CHIB and GREENBERG, 1998) or iterative importance sampling (HAJIVASSILOU and RUUD, 1994) – which cannot be parallelized well. CHEN, XUE, and GOMES (2018) proposes a MC approximation instead which can be parallelized on the GPU using DL frameworks. The MC approximation makes the MVP model scalable for high dimensional data (many responses/species) but there is a tradeoff between accuracy and runtime. Therefore, we study how the MC approximation compares to others for high dimensional JSMD with respect to accuracy and runtime (chapter 4).

## 1.4 Research questions

The goal of this work is to assess the potential of ML and DL algorithms for data analysis of structured data. ML and DL algorithms have the potential to improve the inference of complex ecological patterns and could support computational statistics by making statistical models scalable. To this end, we first introduce ML and DL algorithms, review their current applications in E&E, and explore potential applications with a focus on inference and computational statistics. Our questions are:

1. **ML and DL algorithms and their current state in E&E:** What are the principles of ML and DL algorithms and their current applications in E&E? How does their automatic complexity adjustment work? What are the potential applications of ML and DL beyond predictions (chapter 2)?
2. **Inference in E&E based on ML and DL:** Can we use naive predictive ML and DL algorithms to infer ecologically plausible patterns (chapters 3 and 7)? How reliable are these effects? Can ML and DL algorithms separate collinear effects, an important prerequisite for reliable inference with ML and DL algorithms? (chapter 5)?
3. **Leveraging ML and DL for computational statistics:** Can we scale statistical models such as the multivariate probit model for joint species distribution modeling using DL frameworks? What are the tradeoffs (chapters 4)? What are the minimum number of levels required for random effect estimation (chapters 8)?



---

## MACHINE LEARNING AND DEEP LEARNING — A REVIEW FOR ECOLOGISTS

---

**Maximilian Pichler and Florian Hartig**

Published in *Methods in Ecology and Evolution*, 2023, 10.1111/2041-210X.14061

### **Abstract**

1. The popularity of machine learning (ML), deep learning (DL) and artificial intelligence (AI) has risen sharply in recent years. Despite this spike in popularity, the inner workings of ML and DL algorithms are often perceived as opaque, and their relationship to classical data analysis tools remains debated.
2. Although it is often assumed that ML and DL excel primarily at making predictions, ML and DL can also be used for analytical tasks traditionally addressed with statistical models. Moreover, most recent discussions and reviews on ML focus mainly on DL, failing to synthesise the wealth of ML algorithms with different advantages and general principles.
3. Here, we provide a comprehensive overview of the field of ML and DL, starting by summarizing its historical developments, existing algorithm families, differences to traditional statistical tools, and universal ML principles. We then discuss why and when ML and DL models excel at prediction tasks and where they could offer alternatives to traditional statistical methods for inference, highlighting current and emerging applications for ecological problems. Finally, we summarize emerging trends such as scientific and causal ML, explainable AI, and responsible AI that may significantly impact ecological data analysis in the future.
4. We conclude that ML and DL are powerful new tools for predictive modelling and data analysis. The superior performance of ML and DL algorithms compared to statistical models can be explained by their higher flexibility and automatic data-dependent complexity optimization. However, their use for causal inference is still disputed as the focus of ML and DL methods on predictions creates challenges for the interpretation of these models. Nevertheless, we expect ML and DL to become an indispensable tool in ecology and evolution, comparable to other traditional statistical tools.

**Keywords:** Artificial intelligence, machine learning, deep learning, big data, causal inference

## 2.1 Introduction

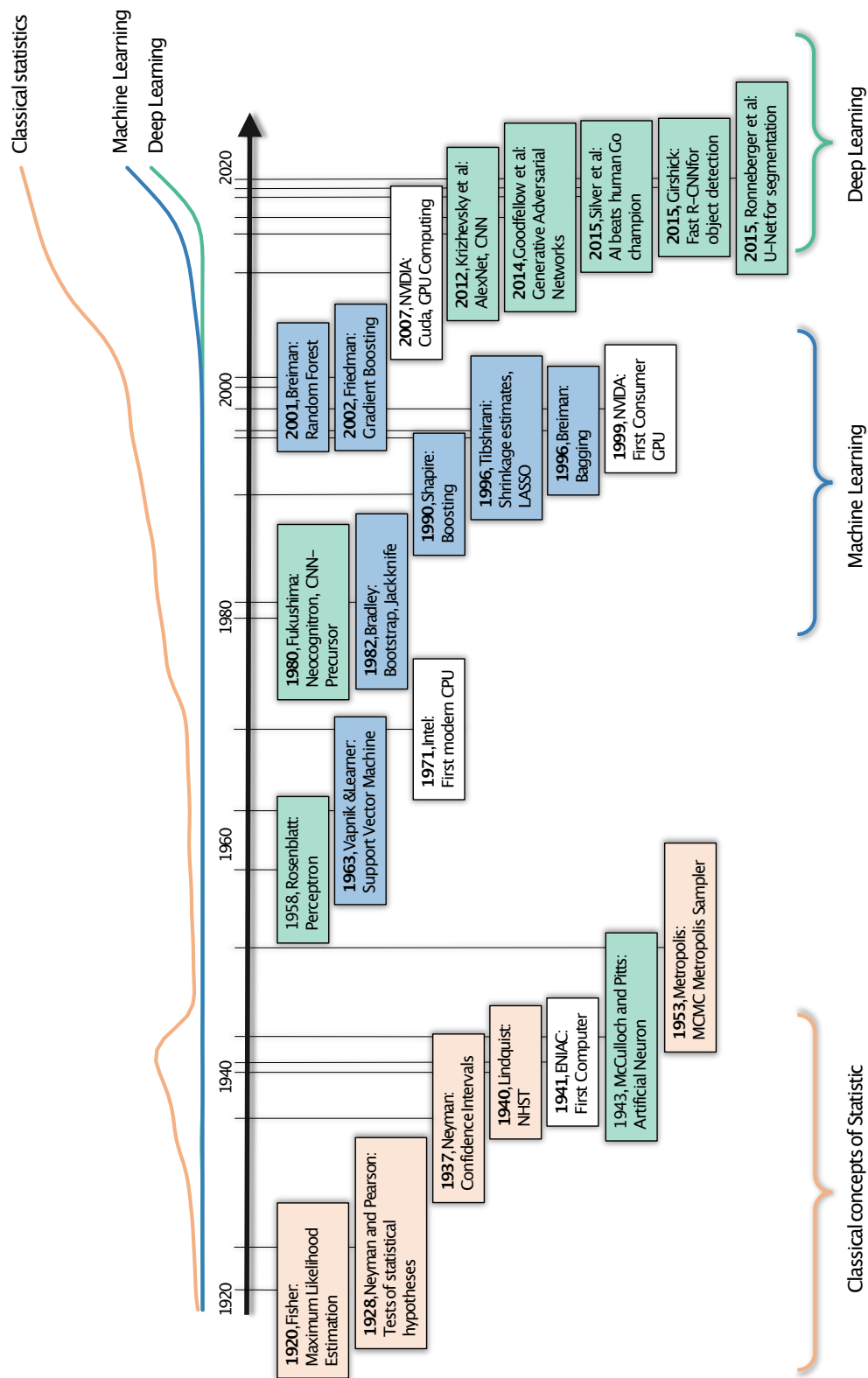
In recent years, machine learning (ML), artificial intelligence (AI) and deep learning (DL) have revolutionized almost all areas of science (JORDAN and MITCHELL, 2015). Early ML algorithms emerged together with the first computers in the '50s, and co-evolved with advances in computing power ever since. During the '90s, the ML field experienced its first bloom, when a wave of fundamental concepts and algorithms such as boosting, bagging, shrinkage estimation and random forest (RF) were discovered. These algorithms challenged, for the first time, the supremacy of classical probability-based statistical models for data analysis and predictions. In the last decade, a second revolution occurred with the rediscovery of deep neural networks, fueled by the availability of graphics processing units (GPUs; 'graphic cards') which made applying these large neural networks practical for the first time. Famous breakthroughs of DL include playing Go (AlphaZero, SILVER *et al.*, 2017), natural language processing (NLP, e.g. GPT-2, RADFORD *et al.*, 2019), detecting and identifying objects in images (Mask R-CNN, HE, GKIOXARI, *et al.*, 2017), short time weather forecasts RAVURI *et al.*, 2021, and predicting protein structures (AlphaFold, JUMPER *et al.*, 2021).

Research in ecology and evolution (E&E) has eagerly adopted both waves of innovation. Several reviews have highlighted the potential of recent advances in DL (BOROWIEC, FRANDSEN, *et al.*, 2021; CHRISTIN, HERVET, and LECOMTE, 2019; TUIA *et al.*, 2022; WÄLDCHEN and MÄDER, 2018), particularly for processing ecological data such as species recognition from video and audio analysis (AODHA *et al.*, 2018; FRITZLER, KOITKA, and FRIEDRICH, 2017; GRAY *et al.*, 2019; GUIRADO *et al.*, 2018) or for extracting trait or behavioural information (DUNKER *et al.*, 2020; GRAVING *et al.*, 2019; MATHIS *et al.*, 2018; OTT and LAUTENSCHLAGER, 2022; PEREIRA *et al.*, 2019). A second area where both traditional ML and DL approaches are already widely used in E&E is predictive modelling. Examples include filling missing links in ecological networks (e.g. DESJARDINS-PROULX *et al.*, 2017), as part of or in conjunction with traditional mechanistic models (RAMMER and SEIDL, 2019; REICHSTEIN *et al.*, 2019), for approximating differential equations (CHEN, RUBANOVA, *et al.*, 2019; RACKAUCKAS *et al.*, 2021), or for species distribution models (CHEN, XUE, and GOMES, 2018; ELITH and LEATHWICK, 2009; HARRIS, TAYLOR, and WHITE, 2018; WILKINSON *et al.*, 2019).

However, despite the rising popularity and attention, the principles and inner workings of ML and DL algorithms are often still perceived as opaque, and their relationship to more classical tools of data analysis, in particular statistical models, remains debated. Trained ML and DL models are often described as a "black box" because their complexity makes it difficult to understand what they have learned. Explainable AI (xAI) methods address this problem and try to understand how trained ML or DL models make predictions (RIBEIRO, SINGH, and GUESTRIN, 2016; RYO *et al.*, 2021). Moreover, a pervasive concern is that ML models are trained for prediction, but the best predictive model does not necessarily correspond to the causal model (BREIMAN, 2001b; PEARL, 2021; PEARL, 2019, Box 2.2). Many researchers thus assume that ML and DL are unable to generate ecological understanding and can only be used as predictive tools (but see ZHAO and HASTIE, 2021, postfix). This view, however, neglects that there is active research to expand ML and DL methods also to causal inference (CHERNOZHUKOV *et al.*, 2018; SCHÖLKOPF, 2019; ZHAO and HASTIE, 2021), which is the classical domain of inferential (causal inference, confirmatory and similar) statistics.

A second reason for confusion about the field is the wealth of algorithms that have been developed in recent years. Most recent reviews on ML have exclusively focused on DL (BOROWIEC, FRANDSEN, *et al.*, 2021; CHRISTIN, HERVET, and LECOMTE, 2019; WÄLDCHEN and MÄDER, 2018). These algorithms differ considerably from simpler, more traditional ML algorithms such as k-nearest-neighbour or boosted regression trees (BRT), and not all statements that are made with respect to DL algorithms apply across the field of ML algorithms in general. For example, image





**FIGURE 2.1:** The three eras of statistical learning. The classical concept of statistics, such as the maximum likelihood estimation, null hypothesis significance testing (NHST), or the Markov-chain-Monte-Carlo (MCMC) metropolis sampler were developed in the 1920s–1940s. Common machine learning algorithms or techniques such as boosting, random forest, or the LASSO were discovered between 1980 and the early 2000s. While the theoretical foundation for Deep learning was postulated in the ‘60s, it has only gained popularity in recent years. The trend lines above the timeline correspond to the frequency of classical statistics (orange), machine learning (blue), and deep learning (green) terms in the scientific literature (see Appendix S1.1 for more details). CNN, convolutional neural network; GPU, graphics processing unit.

based tasks such as automatic species identification (e.g. FERREIRA *et al.*, 2020; TABAK *et al.*, 2019) profit from the use of DL algorithms because they can process spatial patterns better than other ML algorithms (LECUN, BENGIO, and HINTON, 2015), whereas traditional ML algorithms often cope better with lower number of observations (e.g. PICHLER, BOREUX, *et al.*, 2020) or structured (tabular) data (cf. ARIK and PFISTER, 2020).

Third, a too narrow focus on specific algorithms often prevents researchers from appreciating the general principles that apply across all ML and DL algorithms. For example, the general principles of regularization via shrinkage and model averaging form the backbone of nearly all ML and DL algorithms. Other principles must be relearned when moving from classical ML to DL. For example, the bias-variance tradeoff classically predicts that increasing model complexity reduces systematic model error (bias) at the cost of increasing stochastic error (variance) of the parameters (Box 2.5). For DL models, however, it was shown that beyond a certain point, variance decreases again with model complexity, thus helping very large networks to achieve low generalization error (FRANKLE and CARBIN, 2019; HUH *et al.*, 2021; ZHANG, WANG, LIU, *et al.*, 2021). The question of why deep neural networks do not suffer from overparameterization, but instead even depend on it for making accurate predictions is still heavily debated (SEJNOWSKI, 2020), and we will comment on this later.

In the remainder of the review, we will expand on these ideas. Our aim is to provide a comprehensive overview of the principles of ML and DL, starting with the historical development of this field, how ML and DL algorithms differ from traditional statistical tools and how they can be applied for predictive and explanatory modelling (for recent reviews on specific methods, see e.g. DL: BOROWIEC, FRANDSEN, *et al.*, 2021; CHRISTIN, HERVET, and LECOMTE, 2019; WÄLDCHEN and MÄDER, 2018; Computer vision: LÜRIG *et al.*, 2021 or for specific application areas of ML, see e.g. TUIA *et al.*, 2022 (wildlife management)). After discussing the algorithmic ideas, history and general properties of ML and DL algorithms, we focus on understanding the mechanisms that make ML and DL excel in certain predictive and analytical tasks and how this can be used by ecologists. We also discuss the current limitations of ML and DL and where traditional statistical methods are preferable and highlight current and emerging applications for ecological problems.

## 2.2 History of ML and DL and its relation to statistics

### 2.2.1 Statistics as the starting point for ML

The roots of ML and DL go back a long way, and the development of this field is tightly linked to the development of modern statistics. Apart from Bayesian statistics, many foundational statistical principles such as the maximum likelihood estimation (MLE) or null hypothesis significance testing (NHST) were established in the first half of the 20th century (Figure 2.1, left). The core of these classical parametric methods is the idealization of a data-generating model, which allows the calculation of the probability of making certain observations, given the model assumptions and parameters. Based on this, eminent statisticians such as Fisher, Neyman and Pearson developed the theory and practice of estimating model parameters with confidence intervals (CI) and calculating p-values that has dominated data analysis in E&E to this day (but see DUSHOFF, KAIN, and BOLKER, 2019; GELMAN and LOKEN, 2014; HARTIG and BARRAQUAND, 2022; MUFF *et al.*, 2022).

Initially, the data-generating model underlying these methods had to be relatively simple to make the calculations of the involved probabilities trackable. The emergence of the first computers (Figure 1), supported by the discovery of new numerical algorithms (e.g. Markov-chain-Monte-Carlo (MCMC), METROPOLIS *et al.*, 1953), allowed for a substantial increase in the complexity

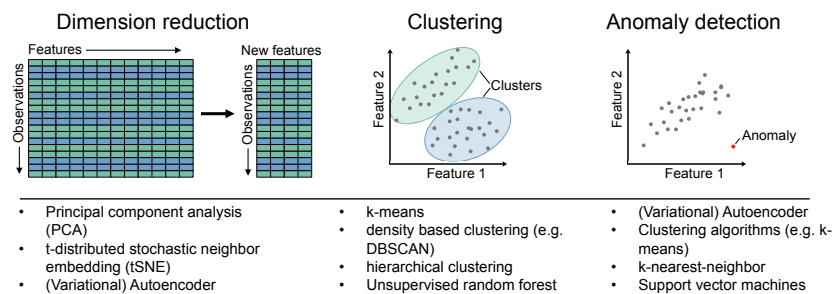
of parametric statistical models, a development reflected in ecological analyses (CLARK, 2005). Even so, when considering the known complexity of the natural world (GRIMM *et al.*, 2005), statistical models tend to be rather simple and rigid, due to the mathematical difficulties involved in calculating likelihoods for more complex or flexible models, and it remains an important caveat of traditional statistical methods that the quality of their inference is conditional on those simplified model assumptions (BREIMAN, 2001b).

## 2.2.2 Machine Learning

The rising availability of computers around the 1980s allowed not only more refined numerical solutions for classical statistical methods but also the development of alternative modelling approaches for data analysis and predictions that we collectively refer to as “machine learning”. Although these approaches differ in their details, we see their commonality in that they abandoned the idea of a probabilistic data-generating model (associated with the ability to calculate p-values, CIs, and all that) in favour of generic algorithmic structures that are trained to perform certain tasks (for general ML principles, see Box 2.3) with the goal of minimizing a general loss function that is not necessarily tied to the probability of the observations (BREIMAN, 2001b; SHMUELI, 2010). Examples of early ML algorithms include neural networks (McCULLOCH and PITTS, 1943), RF (BREIMAN, 2001a), and BRT (FRIEDMAN, 2001; more on these in the Section 2.3).

### Box 3.2: Unsupervised learning in E&E

Unsupervised learning algorithms (for definitions, see Box 2.3) also have interesting applications in E&E. In most cases, the goal is to find patterns in the feature space, for example to reduce the dimensionality of the data, to find clusters of similar data, or to detect anomalies (Figure 2.2).



**FIGURE 2.2:** The three main tasks and their algorithms in unsupervised learning. Dimension reduction techniques reduce the dimensionality of the data by discarding redundant or non-task relevant information. Clustering algorithms try to identify patterns in the data which correspond to mechanistic processes. Anomaly detection is used to identify observations that may have originated from a different data generation process.

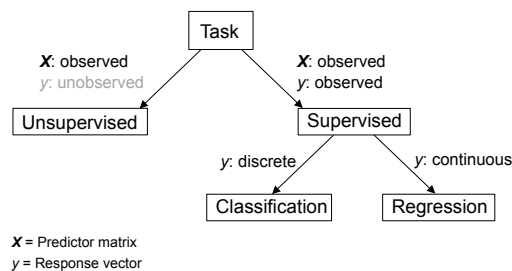
Examples of algorithms that perform dimension reductions that are well-known to ecologists include ordination methods such as principal component analysis (PCA) or t-distributed stochastic neighbour embeddings (tSNE), but also DL algorithms such as variational autoencoders. The latter also works on more complex data such as images. The same is true for clustering and anomaly detection tasks: in addition to simple methods such as k-means, which should be familiar to many ecologists, there are now deep learning methods available that generally have advantages when the data is highly structured, such as in images.

The algorithmic nature of the new ML models lacked the necessary distributional assumptions for calculating p-values and CIs and fueled the development of non-parametric approaches for

### Box 3.1: Basics of ML

#### General objective of ML

The objective of ML is to build a good predictive model. By “good”, we mean that the model should predict well for new data. Sometimes ML models make almost no errors on the data they were trained on, but fail for new data (we say the model overfits). A more complex and flexible model has a higher risk of overfitting. The tradeoff between complexity and flexibility can be depicted by the bias-variance tradeoff (Box 2.5). The general ideal of ML algorithms is thus to take a certain algorithmic structure and then adjust their parameters to the data (training), while simultaneously adapting its complexity by optimizing the bias-variance tradeoff so that the fitted model generalizes well to new data.



**FIGURE 2.3:** Decision tree to assist in task identification. Given feature matrix  $X$  and a response vector  $y$ , the first decision is to choose between unsupervised (outcome  $y$  is unobserved) and supervised (outcome  $y$  is observed) learning. In the case of supervised learning, if  $y$  is discrete (e.g., species classes), it is a classification task, and if  $y$  is a continuous variable (e.g., biomass), it is a regression task.

#### Tasks and learning situations

In ML, the different use cases for the algorithms are called tasks. In supervised learning, examples of the ‘correct’ execution of the task are presented to the algorithm, and the model is trained to minimize the differences between its own actions and the “correct” actions. Common supervised tasks are classification (e.g., labeling of images) and regression (predicting a numerical variable). In contrast to that, unsupervised learning refers to tasks where no examples are supplied, and the algorithms optimize some general loss function (e.g., genomic species delimitation, see DERKARBETIAN *et al.*, 2019). Finally, in reinforcement learning, the ML algorithm is trained by interacting with a (virtual) environment. Reinforcement learning is used in tasks where the learning depends on executed actions and their produced consequences, for instance, playing strategy computer games such as DOTA (BERNER *et al.*, 2019) or Starcraft (VINYALS *et al.*, 2019).

#### Model classes and architectures

In principle, any algorithm that makes predictions for a given task can be used for ML. In practice, for supervised learning, the most commonly used model classes and architectures can be broadly divided into neural networks, which mimic the functioning of a brain, regression and classification trees, and distance-based method (see Section 2.3). In unsupervised learning, model classes can be broadly divided into agglomerative hierarchical methods and methods where the number of clusters must be specified a priori (e.g., k-means; 2.2).

#### Training the models

In supervised and reinforcement learning, training a model consists of two steps. The first step is to define a loss function that measures the current score (performance) of the algorithm in solving a certain task. The loss function differs for classification and regression tasks (e.g., mean squared error and categorical cross-entropy are common loss functions for regression and classification tasks). The second component is the optimizer, which updates the parameters of the algorithm with the goal to improve its performance. In unsupervised learning a common approach is to use similarities between observations to decide whether to group observations together.

estimating model uncertainty. A famous example is the bootstrap (EFRON, 1992), a resampling technique that is often used for estimating CIs on the parameters and predictions of statistical or ML models. Another example is cross-validation, where a part of the data is used to train the model and the other part of the data is used to evaluate the error (STONE, 1974; see ROBERTS *et al.*, 2017 for cross-validation strategies for structured ecological data). Since either of these methods require repeated evaluations of the model, their application would be unimageable without computers and even today, they can be computationally challenging for complex models (more on this in Section 2.4).

### 2.2.3 Deep learning

The co-evolution of computational resources and ML algorithms reached a final peak with the emergence of DL algorithms in the last decade. DL algorithms are neural networks (McCULLOCH and PITTTS, 1943) that differ from classical artificial neural networks (ANNs) mainly by their size. While many algorithms and network architectures that are used today were already described in the '80s and '90s (e.g. FUKUSHIMA, 1980; LECUN *et al.*, 1998), their practical application was prevented by the lack of computing power at the time. This changed with the emergence of GPUs in the '90s (Figure 2.1). Although GPUs were originally developed for computer games or other graphical rendering tasks, it was quickly realized that they are often far more efficient than CPUs for certain numerical and linear algebra tasks. KRIZHEVSKY, SUTSKEVER, and HINTON (2012) ushered in the new era of DL when they demonstrated that their competition-winning neural network could be trained on a GPU within hours instead of days or weeks on a CPU. Today, large DL models trained on GPUs with hundreds of millions parameters dominate the competition for many complex ML tasks, and their behaviour is often markedly different from that of simple ML algorithms (see Section 2.4).

## 2.3 Important ML algorithms in more detail

Considering that ML branched off from classical statistical models with the goal of increasing model flexibility and complexity while abandoning the idea of a probabilistic model, it seems obvious to discuss the advantages and disadvantages of this decision. We will do so in Section 2.4.

Before that, however, it will be useful to explain the most important ML algorithm in more detail. In the main text, we focus on algorithms for supervised learning (see Box 2.3 for definitions of ML tasks) but we also provide a short overview about unsupervised learning in Box 2.2. Note that classical statistical models such as linear and logistic regression models can also be used for supervised regression and binary classification tasks, respectively. Arguably, they provide a baseline that ML models should be able to beat. However, because we assume that ecologists are aware of these models, and because our very aim here is to understand why ML algorithms can beat these models, we do not describe them in this section. R, Python, and Julia code examples for all ML and DL algorithms that are discussed (Table 2.1) are available in Appendix S1.2 or at <https://maximilianpi.github.io/Pichler-and-Hartig-2022>.

### 2.3.1 Support vector machines

A support-vector machine (SVM) is a binary classifier (which can be extended to multiclass and regression tasks, Table 1) that separates the available classes by a hyperplane in the feature (predictor) space (see SVM in 2.1). A predecessor of the SVM, the generalized portrait algorithm, was proposed already by VAPNIK and LERNER (1963). The generalized portrait algorithm was computationally cheap (which was important at the time), but as the perceptron, the initial

predecessor of ANNs, it was unable to solve non-linear tasks. BOSER, GUYON, and VAPNIK (1992) overcame this obstacle by using a non-linear feature space transformation (the kernel-trick) to make the task linearly separable (the modern form of the generalized portrait algorithm, the SVM). Because of their computational efficiency for dealing with high dimensional data and relatively low data requirements (compared to DL), SVMs were the most common method for image classification in E&E, particularly in remote sensing (e.g. GUALTIERI and CROMP, 1999; MELGANI and BRUZZONE, 2004; MOUNTRAKIS, IM, and OGOLE, 2011), prior to the success of DL.

### 2.3.2 The emergence of ensembles models

Apart from SVMs, ensemble models are the other central ML paradigm that emerged in the 90s: SCHAPIRE (1990) showed that ensembles of weak learners (typically simple models such as linear regression models or classification and regression trees, see FRIEDMAN, 2001) often have a low prediction error when their predictions are averaged, even if each individual model has large prediction errors. This principle of generating ensembles of “weak learners” gave rise to two prominent ML techniques: boosting and bagging.

Boosting is an ensemble modelling approach in which weak models are trained sequentially, either by training the next model to correct the errors of the previous model (high-weighting of misclassified observations, AdaBoost, see FREUND and SCHAPIRE, 1997), or by sequentially optimizing a general differentiable objective (cost) function, gradient boosting (FRIEDMAN, 2001) with the latter being the state-of-the-art today (e.g. BRT for species distribution models, see ELITH, LEATHWICK, and HASTIE, 2008; ELITH and LEATHWICK, 2009; Table 2.1).

In bagging (bootstrap aggregation), an ensemble of independent weak models is created by training models on bootstrap samples (BREIMAN, 1996). A famous representative is the RF algorithm which additionally subsamples the features in each node of the decision trees (Table 2.1; BREIMAN, 2001a).

Ensemble models are based on an important ML principle that is still valid today: simple algorithms or statistical models can be transformed into more complex algorithms by creating ensembles, which are more difficult to interpret, but often have low prediction errors (see Section 2.4). BRT and RF are still widely applied, mostly for structured tabular data (Table 2.1), also because they cope better with smaller datasets than comparable DL models. Examples of recent applications of ensemble models in E&E include predictions in ecological networks (PICHLER, BOREUX, *et al.*, 2020), linking gene variation to phenotypes (BRIEUC *et al.*, 2018), species distribution models (ELITH and LEATHWICK, 2009), and various applications in remote sensing (BELGIU and DRĂGUȚ, 2016).

### 2.3.3 Neural networks

Artificial neural network (ANN), inspired by the architecture of our brains, are arguably the most iconic ML architecture, reflecting the long-held dream of building intelligence into a computer. The first fully functional ANN was described by ROSENBLATT (1958). This “perceptron algorithm” was a binary classifier that connected the input neurons (one for each input variable = feature) to an output neuron (response). If the signal in the output crossed a certain threshold (activation function), the predicted class changed (e.g., from ‘0’ to ‘1’). However, because of its limited flexibility and particularly its inability to represent nonlinear relationships, the perceptron fell into oblivion for many years until it was discovered that additional layers between the input and output neurons (so-called ‘hidden’ layers) made it possible to approximate any functional form (see subSection 2.3.4). The potential of ANNs for ecological applications was recognized early (e.g. FOODY, 1995; SIMPSON *et al.*, 1992; FRENCH and RECKNAGEL, 1970), although to date they have

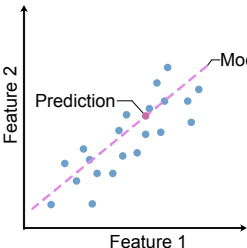
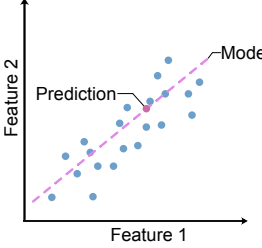
largely been replaced in E&E by the more advanced Deep Neural Networks.

### 2.3.4 Deep learning architectures

Deep learning models represent the latest methodological advance in ML (Figure 2.4). DL algorithms are neural networks which differ from simple ANN in the larger number of hidden layers (BOROWIEC, DIKOW, *et al.*, 2022; LECUN, BENGIO, and HINTON, 2015) and the often more complicated connection between the neurons (=architecture). Complex task-specific architectures, often with millions of parameters and specific structures, evolved over the years (for example residual neural networks, HE, ZHANG, *et al.*, 2015, see also Table 2.1).

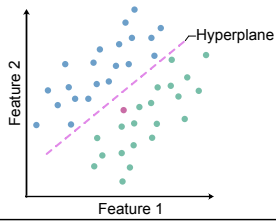
Although DL is based on the same ideas and principles as all other ML algorithms, it is commonly treated as a new field because of its distinct principles (see Section 2.4) and the task-specific architectures that do not resemble traditional ML models. For example image-based tasks (e.g., species identification, see BOROWIEC, DIKOW, *et al.*, 2022) are typically handled by convolutional neural networks (CNNs), a special architecture that uses kernels (convolution matrices) to detect certain shapes and which is used, for example, in identifying species (FERREIRA *et al.*, 2020), automatic monitoring of species (NOROUZZADEH, NGUYEN, *et al.*, 2018; TUIA *et al.*, 2022), or landscape classification (STUPARIU *et al.*, 2022). Recurrent neural networks (RNNs) are another architecture that is applied for time series tasks (Table 2.1; CHRISTIN, HERVET, and LECOMTE, 2019; LECUN, BENGIO, and HINTON, 2015). In ecology, for example, RNNs have been used to predict population dynamics (JOSEPH, 2020a) or animal movements (REW *et al.*, 2019; see BOROWIEC, DIKOW, *et al.*, 2022 for more details on different DL algorithms). DL algorithms have also been used to synthesize taxonomic information from literature (LE GUILLARME and THUILLER, 2022), to predict species interactions in ecological networks (STRYDOM *et al.*, 2021), or to predict species distributions (DENEU *et al.*, 2021). In the following, we treat DL as a subfield of ML and only mention DL when relevant differences to classical ML algorithms are involved.

**TABLE 2.1:** Overview of common supervised machine learning algorithms and their most common application areas. Word clouds were created by searching abstracts and titles in the ecology and evolution literature within the specific machine learning algorithms for ecological keywords, the size of the words corresponds to their frequency (see Appendix S1.1).

Machine learning model	Description	Data Type	Application areas
	Lasso, Ridge Regression: Regression models with regularized coefficients (Appendix S1.2.1): + highly interpretable + few observations - Not very flexible	Tabular data: - Classification - Regression	

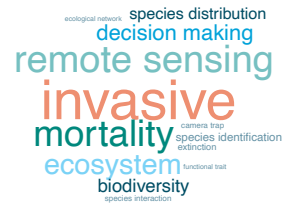
*continues on next page*

Support vector machines:

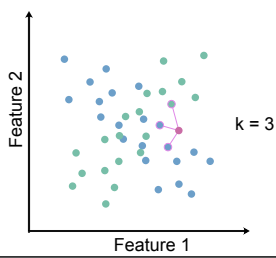


Hyperplane is optimized to separate response classes (Appendix S1.2.2):  
 + fast and memory efficient  
 + high dimensional data  
 - kernel dependent  
 - no probabilities

Tabular data:  
 - Classification  
 - Regression

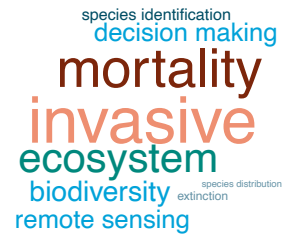


k-nearest-neighbor:

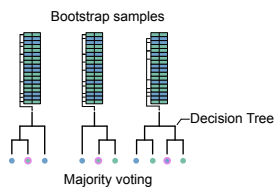


K nearest neighbors in feature space decide response (e.g., by majority voting) (Appendix S1.2.3):  
 + simple  
 + no training  
 - scales poorly  
 - high dimensionality

Tabular data:  
 - Classification  
 - Regression

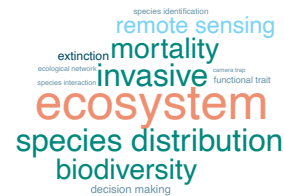


Random Forest:

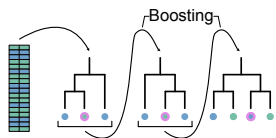


N decision (regression) trees are fitted on bootstrap samples. Split variable is selected from random subset of variables (Appendix S1.2.4):  
 + flexible  
 + robust (e.g., outliers)  
 + few hyperparameters  
 (+) variable importance  
 - scales poorly

Tabular data:  
 - Classification  
 - Regression

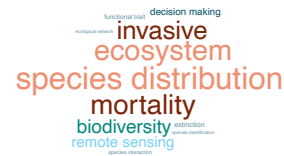


Boosted Regression Trees:

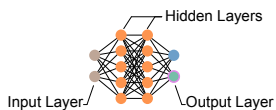


N trees are fitted sequentially to minimize an overall loss function (Appendix S1.2.5):  
 + flexible  
 (+) variable importance  
 - many hyperparameters  
 - high complexity

Tabular data:  
 - Classification  
 - Regression

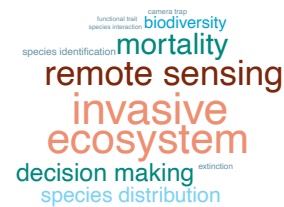


Deep neural networks:



Input (features) are passed through many hidden layers. Last layer maps into response space (Appendix S1.2.6):  
 + flexible  
 + adaptive to different tasks  
 - many hyperparameters  
 - computationally expensive

Tabular data:  
 - Classification  
 - Regression



*continues on next page*



<p>Convolutional Neural Networks:</p>	<p>Neural</p>	<p>Small kernels (filters) processes images before passing it to fully connected layers (Appendix S1.2.7):                  + flexible                  + detecting shapes and edges                  - many hyperparameters                  - computationally expensive</p>	<p>Tabular data:                  - Classification                  - Regression</p>	
<p>Recurrent Neural Networks:</p>	<p>Neural Net-</p>	<p>RNN-Cells (e.g. Long short term memory cells) process the input sequence and hidden states are re-cycled (Appendix S1.2.8):                  + flexible                  - long-term dependencies are difficult to learn                  - many hyperparameters                  - computationally expensive</p>	<p>Tabular data:                  - Classification                  - Regression</p>	
<p>Graph Neural Networks:</p>		<p>GNNs operate directly on the edges and nodes of graphs. They can be used for a variety of different tasks such as node or edge classifications (Appendix S1.2.9):                  + flexible                  + non-Euclidean data                  - many hyperparameters                  - computationally expensive                  - complex data type</p>	<p>Tabular data:                  - Classification                  - Regression</p>	

TABLE 2.2: Common ML and DL libraries and frameworks.

Name	Description	Language	Link
ranger	Random Forest algorithm.	R	<a href="https://github.com/imbs-hl/ranger">https://github.com/imbs-hl/ranger</a>
xgboost	Boosted Machine framework.	R, Python	<a href="https://github.com/dmlc/xgboost">https://github.com/dmlc/xgboost</a>
lightGBM	Boosted Machine framework.	R, Python	<a href="https://github.com/microsoft/LightGBM">https://github.com/microsoft/LightGBM</a>

*continues on next page*

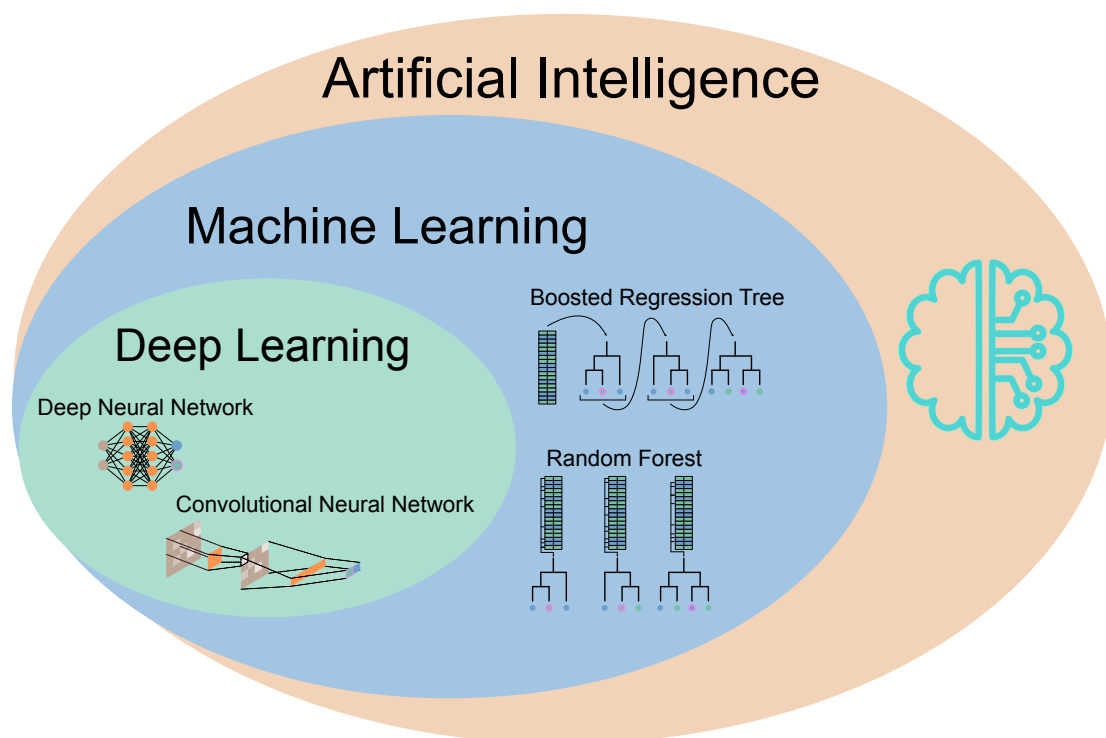
caret	ML framework for hyperparameter tuning and cross-validation . Supports different ML algorithms.	R	<a href="https://topepo.github.io/caret">https://topepo.github.io/caret</a>
mlr3	ML framework for hyperparameter tuning and cross-validation . Supports different ML algorithms.	R	<a href="https://mlr3.mlr-org.com">https://mlr3.mlr-org.com</a>
tidymodels	(ML) framework for hyperparameter tuning and cross-validation . Supports different (ML) algorithms.	R	<a href="https://www.tidymodels.org">https://www.tidymodels.org</a>
Scikit-learn	ML framework for hyperparameter tuning and cross-validation . Supports different ML algorithms.	Python	<a href="https://scikit-learn.org">https://scikit-learn.org</a>
TensorFlow	Deep Learning framework.	R, Python	<a href="https://www.tensorflow.org">https://www.tensorflow.org</a>
Keras	Higher-level deep learning framework.	R, Python	<a href="https://keras.io">https://keras.io</a>
PyTorch	Deep Learning framework.	R, Python	<a href="https://pytorch.org">https://pytorch.org</a>
PyTorch Geometric	Graph Neural Network (GNN) framework. Supports different GNN algorithms.	Python	<a href="https://github.com/pyg-team/pytorch_geometric">https://github.com/pyg-team/pytorch_geometric</a>
Flux	Deep learning framework.	Julia	<a href="https://fluxml.ai/Flux.jl/stable">https://fluxml.ai/Flux.jl/stable</a>
MLJ	ML framework. Supports different ML algorithms.	Julia	<a href="https://alan-turing-institute.github.io/MLJ.jl/stable">https://alan-turing-institute.github.io/MLJ.jl/stable</a>

## 2.4 Why does ML work?

When considering current DL algorithms with millions of parameters, researchers trained in classical statistics often struggle to understand why they should work at all. A statistical model with a similar number of parameters could likely not even be fit (e.g., in a linear regression model, if the number of parameters is greater than the number of observations, there are no degrees of freedom and the equation system is underdetermined). And even if it were possible to fit the model, the bias-variance tradeoff that is fundamental to both statistics and ML (Box 2.5; Figure 2.5a) predicts that the optimal compromise between systematic model error (bias) and

error due to variance (parameter uncertainty) is at intermediate model complexity (Boxes 2.5 and 2.6). Excessively large models should therefore overfit the data and generalize badly.

Despite that, the practical experience shows that ML models converge and generalize well to new data, suggesting that they do not overfit (at least for in-distribution predictions; extrapolation beyond the data domain is as challenging for ML as for other approaches). Even more surprisingly, it was observed that for deep neural networks, the bias-variance tradeoff actually reverts, following a double-descent curve (Box 2.5; Figure 2.5b; see BELKIN *et al.*, 2019; NAKKIRAN *et al.*, 2019): beyond a certain model size, making deep neural networks even deeper and wider decreases generalization error, suggesting that model size, contrary to our general expectation, can actually be beneficial for reducing total prediction error (ARORA *et al.*, 2019; HUH *et al.*, 2021; NOVAK *et al.*, 2019; SHWARTZ-ZIV and ALEMI, 2020; NAKKIRAN *et al.*, 2019).



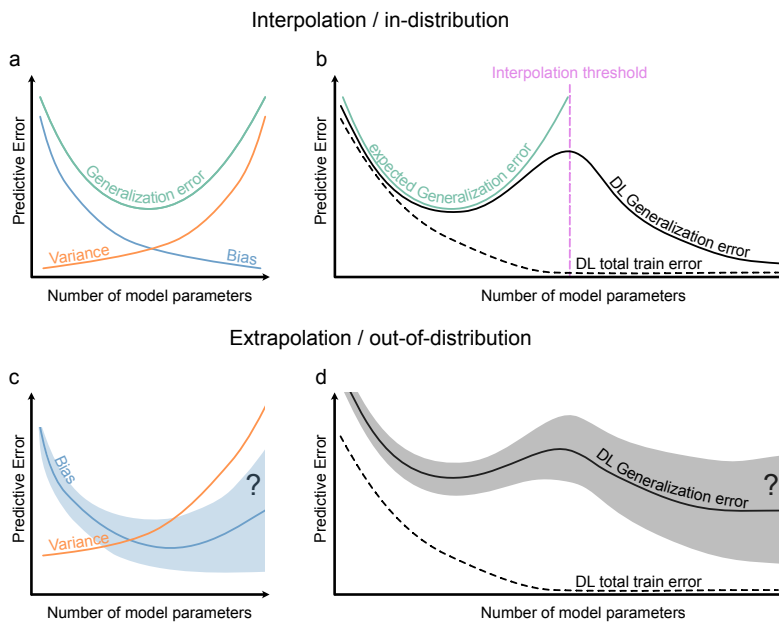
**FIGURE 2.4:** Relationship between artificial intelligence (AI), machine learning (ML), and deep learning (DL). AI refers to algorithms that are capable of achieving similar to human-like performance in specific decision or recognition tasks. This is sometimes contrasted with the pursuit of Artificial general intelligence, which refers to AI algorithms that can perform a wide range of tasks and may display human-like abilities in cognitive tasks such as reasoning, logic or common sense. ML algorithms serve as a tool for AI systems to learn from data and make a decision based on data. There are many different ML algorithms such as boosted regression trees or random forest. Within ML, a family of ML algorithms based on artificial neural networks emerged in recent years. Due to their similarities in the way they work and their backbone, DL is considered as a family of its own.

The reason for this superficially perplexing behaviour is that practically all ML approaches, despite formally having a very high number of parameters and the associated ability to model complex input–output relationships, perform implicit complexity adjustments that limit their flexibility and avoid overfitting. As a result, especially for DL models, the number of parameters is a poor measure of effective model complexity (BIRDAL *et al.*, 2021), which is confirmed by more appropriate complexity measurements (Box 2.5; Figure 2.5b; see BIRDAL *et al.*, 2021; NAKKIRAN

### Box 3.3: Generalization error and the bias-variance tradeoff

By making models more complex, one can make the prediction error on the training data (*in-sample error*) arbitrarily small. What we really care about, however, is the ability of a model to predict new (out-of-sample data). The discrepancy between model predictions and observations on independent data (e.g., generated by an appropriate cross-validation, see ROBERTS *et al.*, 2017) is called the *generalization error*.

When minimizing the generalization error, there exists a fundamental tradeoff between variance (parameter uncertainty) and bias. More complex models have higher variance, but also lower bias (Figure 2.5a). The two counteracting errors usually lead to a sweet spot of the generalization error at intermediate model complexity. Interestingly, DL models show a double sloping curve of generalization error (Figure 2.5b), suggesting that the variance of deep neural networks does not increase for very wide and deep networks. The reasons for this are still being discussed in the literature.



**FIGURE 2.5:** Typical bias-variance tradeoffs in classical machine learning (left panels) and deep learning (right panels) models for interpolation (in-distribution) and extrapolation (out-of-distribution) tasks. In contrast to the classical bias-variance tradeoff in panel (a), the bias-variance tradeoff for DL in panel (b) shows that after the interpolation threshold (pink dotted line) the training loss is constant (i.e., bias is not improved by increasing model complexity), but the test loss (and thus variance) can be still reduced by increasing the model size. Note also that the total generalization error in extrapolation tasks (panels c and d) is usually higher and the optimal model complexity lower, as the bias will not go to zero with increasing model complexity (depending on the similarity between training and test data).

*Overfitting/Underfitting* describes a situation where the generalization error is higher than necessary or expected. In interpolation (in-distribution, Figure 2.5a, b) tasks, overfitting/underfitting is usually associated with too high/low model complexity, which leads to a poor compromise between bias and variance. In extrapolation (out-of-sample, see Figure 2.5c,d) tasks, the reasons for overfitting are often more rooted in bias problems, meaning that the patterns learned in the training data do not generalize to the test data (see, e.g. YANG *et al.*, 2020 for an example in vision tasks).

*et al.*, 2019). We divide the underlying mechanisms that adjust model complexity in ML algorithms into two categories: internal (algorithmic) and external (optimization) based approaches.

### 2.4.1 Internal (algorithmic) complexity optimization

By internal (algorithmic) complexity optimization, we understand algorithmic structures that lead to a self-adaptation of model complexity. One basic mechanism for generating this behaviour is that many ML algorithms implicitly or explicitly generate ensemble predictions. An ensemble model may formally include many parameters, but its effective complexity is by no means the sum of the complexity of each ensemble member. Rather, the complexity of an ensemble model is typically related to the average complexity of its ensemble members and the differences between them, which in turn affect error and variance of the ensemble estimator (BERNARDO and SMITH, 2009; DIETTERICH, 2000; DORMANN, CALABRESE, *et al.*, 2018; GANAIE *et al.*, 2021).

Because the ensemble members are fit to the data, the data can influence the difference between the ensemble members and thus the complexity of the entire ensemble estimator. To support this behaviour, many ML algorithms include (tunable) mechanisms to increase heterogeneity in the ensemble. For example, bagging decreases the similarity between ensemble members by bootstrapping the data (SAGI and ROKACH, 2018). RF goes one step further by using a random subset of the features in each node, which further diversifies and decorrelates the individual models (BREIMAN, 2001a) and reduces the variance of the ensemble (BREIMAN, 2001a). In gradient boosting (see FRIEDMAN, 2001), the subsequently trained models depend on the previous model but they are uncorrelated because the following members are forced to compensate for the errors of the previous model (SAGI and ROKACH, 2018).

### 2.4.2 External (optimization) adaptation of model complexity

On top of internal mechanisms to adopt model complexity, most practical ML pipelines apply an additional optimization step where hyperparameters of the model are optimized under cross-validation (or simply into training, evaluation, and test splits for large DL models). Hyperparameters are parameters that do not directly control predictions, but rather the architecture (e.g., number of nodes in a hidden layer of a neural network or the number of trees in a RF) or the learning behaviour of ML algorithms. Some ML algorithms have few (e.g., RF), others many (BRT or DL) hyperparameters. Hyperparameters are usually tuned via a nested cross-validation setup, that is, an outer cross-validation to estimate generally the prediction error of the model and an inner cross-validation to control the tuning (see Table 2.2 for ML frameworks).

A particularly important class of hyperparameters are regularization parameters, which control the flexibility of the algorithms. In general, regularization means imposing constraints on an algorithm to limit its flexibility. The type and strength of the regularization depends on the task, the data and the algorithm but the most common regularization type is a so-called shrinkage penalty which biases parameter estimates to a certain value, typically zero. For example, L1 (LASSO, TIBSHIRANI, 1996) and L2 (Ridge; HOERL and KENNARD, 1970), or elastic-net when combined (ZOU and HASTIE, 2005), intentionally biases the estimates to zero. Shrinkage penalties were originally developed to estimate complex statistical models (e.g., when number of observations  $\ll$  number of predictors) such as linear or logistic regression models but have since been adopted in ML models. In tree-based methods (e.g. RF), hyperparameters such as the depth of the trees have regularizing effects, whereas in DL a range of regularization techniques is used, such as L1 or L2 on weights and dropout (where random parts of the network are set to zero during training; see SRIVASTAVA *et al.*, 2014).

### 2.4.3 Open questions regarding model complexity in DL

While the principles of internal and external complexity adoption are central to both classical ML and DL algorithms, it is often conjectured that, they alone are not sufficient to explain the success of the highly complex DL algorithms, in particular the puzzling double-decent behaviour where generalization loss improves with model size even after the training loss has reached a value close to zero (Figure 2.4b), a behaviour that is not observed in simpler ML algorithms.

One hypothesis to explain the discrepancy between simple and deep neural networks is that overparameterization combined with stochastic training of the networks (stochastic gradient descent) leads to an implicit regularization (ARORA *et al.*, 2019; HUH *et al.*, 2021; LI, LUO, and LYU, 2021). This would explain why deep neural networks display a bias towards simpler functions (DE PALMA, KIANI, and LLOYD, 2019; VALLE-PÉREZ, CAMARGO, and LOUIS, 2019) that increases with the depth of the networks (HUH *et al.*, 2021). It was also observed that often over 90% of the trained networks' parameters can be set to zero with little or no loss of generalization accuracy (FRANKLE and CARBIN, 2019), suggesting that there is a considerable amount of redundancy and possibly ensemble behaviour in deep neural networks. Such a pruning can reduce the computational cost of the model and reduce the generalization error (BARTOLDSON *et al.*, 2020) or identify robust models (KUHN, LYLE, *et al.*, 2021). It was also suggested that the random initialization of a large DNN is more likely to create a good subnetwork (FRANKLE and CARBIN, 2019; ZHANG, WANG, LIU, *et al.*, 2021), which is then identified by training, regularization or pruning (ZHANG, WANG, LIU, *et al.*, 2021). Moreover, most modern DL models consist of a mix of different architectures and techniques which can even include common ML concepts such as boosting or bagging. For example, dropout training can be interpreted as generating a large number of subnetworks, similar to an ensemble model (SRIVASTAVA *et al.*, 2014), and deep residual networks (HE, ZHANG, *et al.*, 2015) for image-based tasks resemble boosting and thus ensemble models (VEIT, WILBER, and BELONGIE, 2016). None of this fully answers the question of DL's superiority but it does at least provide conjectures that need to be followed up by future research.

## 2.5 Emerging trends in ML (in E&E)

In the last Section of this paper, we will look at the current practice and emerging trends in ML and speculate on how they will impact the field of E&E.

### 2.5.1 Trends in algorithm use in E&E

As a basis for this discussion, we performed a text analysis of the E&E literature over the last decades (for details see Appendix S1.1). Our results show that the use of both ML and DL methods in E&E increased sharply over the last decade (Figure 3). Classical ML methods still dominate in practical applications. Of those, SVMs were the most popular algorithm in the early 2000s, but lost their dominance since then. BRTs became popular in the mid 2010s (Figure 2.6b), and more recently, neural networks (including DL) are rising in popularity (Figure 2.6b). The increase in publications using DL approaches explains why these algorithms receive so much attention in recent reviews, but our analysis (Figure 2.6b) also highlights that classical ML methods still account for a proportionally larger share of all applications.

We anticipate that classical ML will remain important in the future, as many tasks in E&E are more naturally approached with simpler ML algorithms. In particular, there is little evidence that DL can outperform classical ML algorithms in supervised learning tasks with limited structured (tabular) data (cf. STRYDOM *et al.*, 2021). The higher flexibility of DL algorithms tends to be advantageous

only when the data is large and complex enough. One would therefore expect that classical ML algorithms will continue to be used for tasks such as species distribution modelling (BEERY, COLE, *et al.*, 2021; ELITH and LEATHWICK, 2009), with subsequent applications for identifying conservable or restorable areas (MORADI *et al.*, 2019; CHENG, AUGUSTIN, *et al.*, 2018; KWOK, 2019; DUHART *et al.*, 2019), ecosystem service management (DIETTERICH *et al.*, 2012; SOCOLAR *et al.*, 2016), wildlife management (HUMPHRIES, MAGNESS, and HUETTMANN, 2018) and conservation (see TUIA *et al.*, 2022), assessing the risk of invasive species (BARBET-MASSIN *et al.*, 2018; JENSEN *et al.*, 2020), and biodiversity assessments (DISTLER *et al.*, 2015). Other tasks where classical ML will likely remain competitive include filling (knowledge) gaps in datasets (PENONE *et al.*, 2014) or in ecological networks (e.g., food webs, DESJARDINS-PROULX *et al.*, 2017; plant-pollinator networks, PICHLER, BOREUX, *et al.*, 2020; host-parasite networks, DALLAS, PARK, and DRAKE, 2017; DALLAS, RYAN, *et al.*, 2021), and predicting potential wildlife hosts of zoonotic diseases (ALBERY *et al.*, 2021; BECKER *et al.*, 2022; HAN *et al.*, 2015; WARDEH, BAYLIS, and BLAGROVE, 2021).

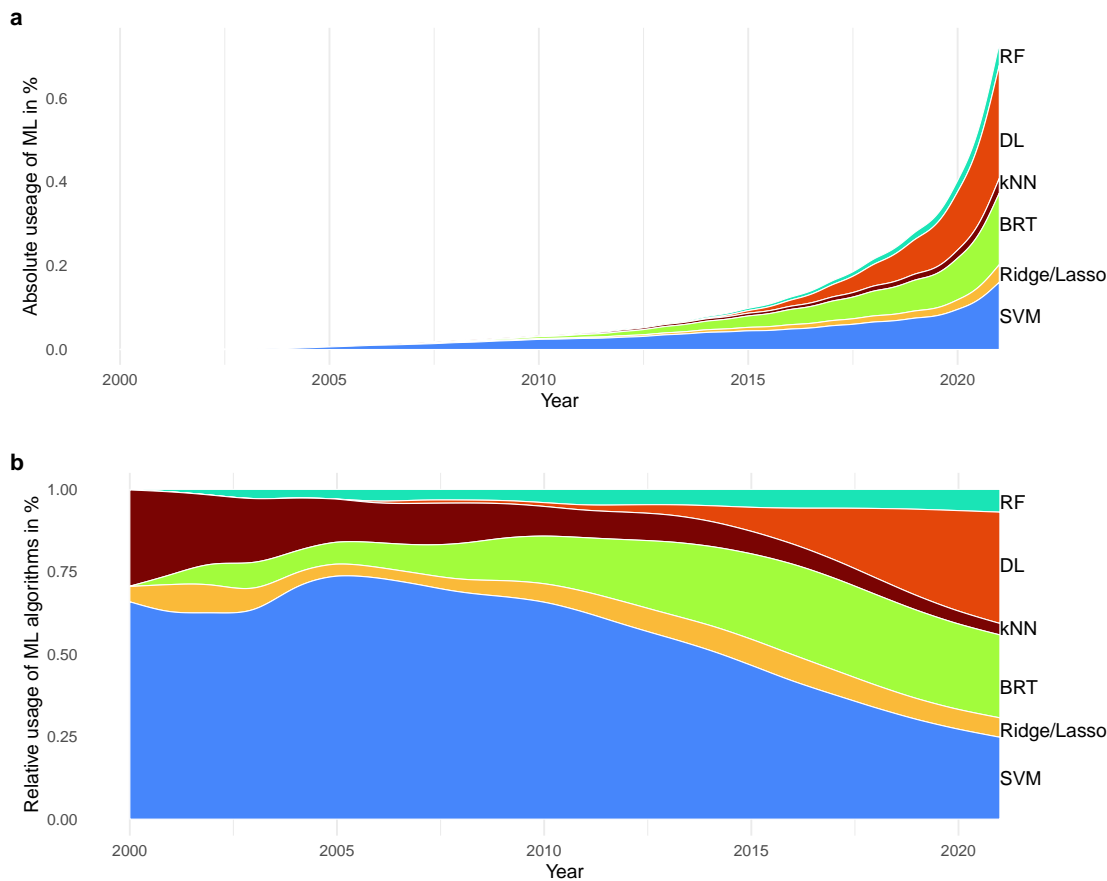
Deep learning algorithms, on the other hand, will likely continue to gain popularity for analyzing complicated and unstructured data in E&E, such as species identification in aerial images (FERREIRA *et al.*, 2020; GRAY *et al.*, 2019; GUIRADO *et al.*, 2018; TORNEY *et al.*, 2019) or camera (trap) images (FERREIRA *et al.*, 2020; MÄDER *et al.*, 2021; TABAK *et al.*, 2019; WILLI *et al.*, 2019; FRITZLER, KOITKA, and FRIEDRICH, 2017; LASSECK, 2018; STOWELL *et al.*, 2018; AODHA *et al.*, 2018; FAIRBRASS *et al.*, 2018; BEERY, WU, *et al.*, 2020; NOROUZZADEH, MORRIS, *et al.*, 2021; VAN HORN *et al.*, 2018).

For clustering and ordination tasks, which have a long tradition in ecology and ML algorithms for unsupervised learning tasks (Box 2.2), classical ML algorithms such as k-means or t-distributed stochastic neighbour embedding algorithms are and will remain important, for example for species delimitation (DERKARABETIAN *et al.*, 2019), outlier detection, identification of eco-provinces (SONNEWALD *et al.*, 2020) or operational taxonomic units (OTUs) in metabarcoding (DEINER *et al.*, 2017). DL-based approaches (e.g., based on (variational) autoencoders), on the other hand, are gaining popularity into certain data-dependent tasks, such as image-based tasks in remote sensing (ZERROUKI *et al.*, 2020) or (genomic) sequences (WANG and GU, 2018).

### 2.5.2 New new applications for ML in E&E

Apart from improving the quality of classical prediction and classification tasks, there are many novel applications that could be addressed in particular by the more advanced DL algorithms. For example, DAVIES *et al.* (2021) demonstrated that DL can aid researchers by generating new hypotheses which were tested afterwards, or it was shown that modern DL models can achieve human-like performance in text generation (BROWN, MANN, *et al.*, 2020). Generative models may play an increasing role in the coming years; however, it is currently difficult to predict where they will be used in ecological research, for example, whether they will help in data-collection or in subsequently answering the research questions themselves (e.g., by generating new hypotheses).

Another interesting field for ML is simulations and simulation-based inference. For stochastic simulations or big process-based models, likelihoods are often intractable or computationally expensive to evaluate (e.g., phylogenetic analyses). ML and DL algorithms can support simulation-based inference by generating new summary statistics (e.g. HAUENSTEIN *et al.*, 2019), by being incorporated into process-based models for computational gains (e.g. RAMMER and SEIDL, 2019), or by emulating them (WANG, FAN, *et al.*, 2019). Moreover, ML can also be used to predict the parameters of complex stochastic models (VOZNICA *et al.*, 2021; ROY, FABLET, and BERTRAND, 2022), and thus act as a likelihood-free calibration method, similar to approximate Bayesian computation (HARTIG, CALABRESE, *et al.*, 2011).



**FIGURE 2.6:** Development usage of ML algorithms (RF, DL, kNN, BRT, Ridge/ Lasso, SVM) in literature from the E&E field (see Appendix S1.1 for more details about the trend analysis). Panel (a) shows the absolute change in their usage in percent and panel (b) shows the relative change in their usage in percent. The overall usage of ML algorithms increased strongly over the last 20 years and especially DL attracted a lot of attention in the last 10 years. BRT, boosted regression trees; DL, deep Learning; E&E, ecology and evolution; kNN, k-nearest neighbour; ML, machine learning; RF, random forest; Ridge/Lasso, ridge or lasso or elastic-net (ridge and lasso) regression; SVM, support vector machines.

In addition, in the era of cheap sensors and other data collection sources, the dimensionality of the data is often difficult to handle with traditional methods. Unsupervised learning algorithms can help to reduce the dimensionality of the data and detect patterns and trends, for example, before the data is used in downstream supervised learning tasks (STRYDOM *et al.*, 2021; ZERROUKI *et al.*, 2020), to handle the data itself (ALVES DE OLIVEIRA *et al.*, 2021), or to detect anomalies in the data (ZHANG, XU, *et al.*, 2021).

### 2.5.3 Rethinking the data collection process in the light of the new methods

The wide availability of DL algorithms could also have a strong impact on data collection in E&E. Image recognition methods can reduce labor costs and thus help to generate much larger datasets. DL can identify species in different data types (FERREIRA *et al.*, 2020; GRAY *et al.*, 2019; GUIRADO *et al.*, 2018; TORNEY *et al.*, 2019; MÄDER *et al.*, 2021; TABAK *et al.*, 2019; WILLI *et al.*, 2019) or extract information such as traits from raw data (DUNKER *et al.*, 2020). Moreover, technical advances in eDNA and other sensor data allow the collection of much larger datasets (e.g., see PIMM *et al.*, 2015) which can then be processed and combined by ML and DL algorithms (see TUIA *et al.*, 2022).



One of the advantages of establishing such machine-assisted data collection and processing pipelines is that they can be reused by many researchers, similar to the development in sequencing technology. For example, once an image-based species recognition pipeline is established, it can be reused without requiring the time of taxonomic experts for data analysis. So far, there are few examples of such ready-to-use pipelines for realistic data collection tasks, and those that exist do not always perform well and generalize well to new situations. However, we believe that the field should develop ML models for data collection that are available to everyone (MCINTIRE *et al.*, 2022) and do not need to be retrained by experts. The NLP community demonstrate how this could be done: Model hubs with many different pre-trained models and a simple and common interface that can be used by everyone (e.g. WOLF *et al.*, 2019; MÄDER *et al.*, 2021, cf. OTT and LAUTENSCHLAGER, 2022).

#### 2.5.4 Making ML work with small datasets

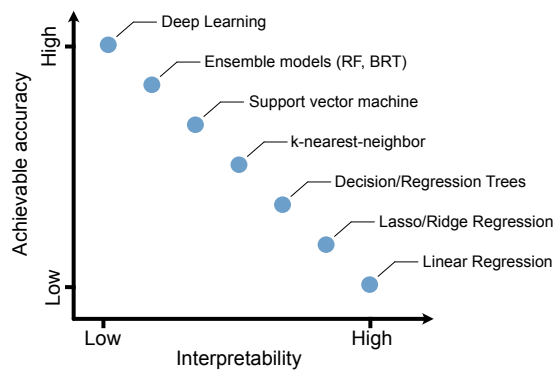
A pervasive problem in the application of many modern ML algorithms for E&E is that their training requires data sizes that are rarely available. New DL techniques such as few-shot learning (see. WANG, YAO, *et al.*, 2020) or transfer learning can greatly reduce the necessary amount of data, and are thus of particular interest to ecologists. As an example, most DL-based image classifiers consist of two stages: first, they identify edges and shapes in the images, and second, they classify the shapes (LECUN, BENGIO, and HINTON, 2015). The first stage makes up a major part of the model and is both data and resource intensive. Research has shown that this part of the network is relatively generic, and usually only the second stage needs to be retrained when a network is adopted for a new task (WEISS, KHOSHGOFTAAR, and WANG, 2016; ZHUANG *et al.*, 2021). Thus, many large model architectures can be downloaded pre-trained and can then be fine-tuned for a new task (transfer learning).

Options such as transfer learning, however, are mainly applicable to vision-based tasks and DL, and not to classical structured tabular data. In the latter, the response is often explained by specific relationships with a particular feature, which rarely generalize to other tasks. In such a situation, there is little to be gained by applying transfer learning, which may also partly explain why DL rarely outperforms traditional ML algorithms on small classical structured tabular data (PICHLER, BOREUX, *et al.*, 2020; SHWARTZ-ZIV and ALEMI, 2020, cf. ARIK and PFISTER, 2020).

Small datasets are common in E&E because observations are often difficult to obtain (e.g., for ecological networks, see MAGLIANESI *et al.*, 2014; STRYDOM *et al.*, 2021); and due to the change in ecological patterns across scales (POISOT, STOUFFER, and GRAVEL, 2015), datasets are difficult to combine. Here, E&E researchers can benefit from the wide range of different ML algorithms (Figure 2.7): SVMs or kNN can handle sparse datasets well (COMO *et al.*, 2017; DRAKE, RANDIN, and GUIAN, 2006) and LASSO, Ridge, and elastic-net regressions are well suited for datasets with more features than observations (ZOU and HASTIE, 2005). On top of the data dimensions, the nature of the signal and the interpretability can influence the choice of the modelling approach (e.g. PICHLER, BOREUX, *et al.*, 2020). However, these tradeoffs are difficult to predict a priori, which explains the common practice of comparing different algorithms for a given task (FAISAL *et al.*, 2010; NORBERG *et al.*, 2019; PICHLER, BOREUX, *et al.*, 2020).

#### 2.5.5 Transparency and bias of decision based on ML and DL models

Predictions and research in E&E often intersect with policy and decision-making (GROOT *et al.*, 2010; SOFAER *et al.*, 2019). As ML models are increasingly used in this context, for example for conservation planning (HUETTMANN, 2018), management decisions (HUMPHRIES, MAGNESS, and



**FIGURE 2.7:** Conceptual illustration of the tradeoff between achievable accuracy and interpretability of machine learning and deep Learning (DL) algorithms. DL can achieve the highest accuracy but shows the lowest interpretability. BRT, boosted regression trees; RF, random forest.

HUETTMANN, 2018), agricultural management (e.g., crop management; LIAKOS *et al.*, 2018), and disease control (e.g. ROMERO *et al.*, 2021), we anticipate that problems of bias and transparency will emerge, as they have in other fields (e.g. HARDT *et al.*, 2016; VAYENA, BLASIMME, and COHEN, 2018).

Transparency refers to the problem that stakeholders may question why certain predictions are being made by the algorithm and whether they can be trusted. Without satisfactory answers to these questions, ML decisions may be subject to legal challenges. While it is not impossible to answer these questions for ML models (see the next subsection on xAI), it is undoubtedly more challenging to understand and communicate the logic of ML decisions, compared to simple statistical models (Figure 2.7).

This lack of transparency also makes it difficult to understand if an algorithm exhibits bias. In the context of ML and AI, bias is understood more broadly than in statistics, and includes both the use of non-representative or socially undesirable training data and the use of features that should not normally be used in decisions (e.g., gender, race). The former occurs when training data was disproportionately collected in different groups or regions (ZOU and SCHIEBINGER, 2018) and is not representative of what the algorithm should learn. A common example from the social sciences is that language models trained on classical literature or texts often learn gender-biased word associations, such as a preference for doctors to be male. Biased data may be a significant problem for E&E, as geographic (MARTIN, BLOSSEY, and ELLIS, 2012; MEYER, WEIGELT, and KREFT, 2016) and taxonomic (PYŠEK *et al.*, 2008; TRIMBLE and AARDE, 2012) sampling biases are common. The use of undesired features describes situations where the data may be representative, but certain features should not be used for ethical reasons. The challenge here is that these features may be implicitly encoded by other features and thus be used in ML and DL algorithms, even if they are not explicitly provided in the data (e.g., ethnic background can be inferred from the people's urban districts, FEUERRIEGEL, DOLATA, and SCHWABE, 2020; HARDT *et al.*, 2016; CALISKAN, BRYSON, and NARAYANAN, 2017).

To understand these issues and to find solutions, the field of responsible and trustworthy AI has formed at the intersection between AI and social science disciplines (sociology, psychology, law). The focus of responsible AI is on creating fair and sustainable ML and DL models and avoiding their misuse or misinterpretation (e.g. BARREDO ARRIETA *et al.*, 2020; WEARN, FREEMAN, and JACOBY, 2019). For example, it is possible to algorithmically detect biases (CIRILLO *et al.*, 2020) and

correct the models accordingly (e.g. ALVI, ZISSERMAN, and NELLAAKER, 2018; KIM *et al.*, 2019) by using fairness metrics to guide training so that underrepresented groups are not neglected (e.g. LIU and VICENTE, 2021).

### 2.5.6 Explainable AI as peephole in black-box models

The previously mentioned transparency issues are amplified by the fact that ML algorithms become increasingly difficult to interpret as model complexity increases (Figure 2.7, BREIMAN, 2001b). Although some algorithms provide metrics for feature importance (e.g. BREIMAN, 2001a; FRIEDMAN, 2001), ML algorithms usually do not provide simple effect estimates, nor do they provide measures of certainty, such as CIs or p-values. This poses a problem not only for ethical transparency but also for researchers that want to understand why predictions are made for scientific reasons.

To address this problem, the field of explainable AI (xAI) has emerged that develops tools to understand how ML models make their predictions. Most xAI methods are post-hoc, meaning that the model is trained first and then investigated (BARREDO ARRIETA *et al.*, 2020; LUCAS, 2020). xAI differs from other similar-sounding approaches such as identifying predictive trait profiles (domain expertise is used to group features and by including and excluding them from the model, their predictive capabilities are estimated (e.g. HAN *et al.*, 2015)), in that the goal is to understand the model itself. Global xAI methods try to summarize the models by generating variable importance (similar to the natural variable importances from RF and BRT; FISHER, RUDIN, and DOMINICI, 2018) or simplify the models by approximating the original model with interpretable models (MOLNAR, 2020). Local xAI methods attempt to explain individual predictions (RIBEIRO, SINGH, and GUESTRIN, 2016). In E&E, for example, xAI methods are already being used to assess trust of predictions from SDMs (RYO *et al.*, 2021). Although there are still many questions about the reliability of these methods, especially under collinearity of features (HOOKER and MENTCH, 2019; YU, JI, PRIHODKO, *et al.*, 2021, but see APLEY, 2016), xAI is becoming an indispensable tool for working with ML models, and presumably there will be specialized xAI methods for ecological applications.

### 2.5.7 Causal inference with Machine Learning

Finally, it is not uncommon for models to give us the right answer for the wrong reason. Sometimes, it is even easier to get good predictions for the wrong reasons. An example is severe feature collinearity, where including all features increases the uncertainties of the estimates (GREENLAND, 2003; LEDERER *et al.*, 2018), whereas removing features, even if causally connected to the response, can improve predictive performance (DORMANN, ELITH, *et al.*, 2013; HOERL and KENNARD, 1970; ARIF and MACNEIL, 2022a). Interestingly, some of the regularization techniques now widely used in ML were originally developed to reduce collinearity problems (HOERL and KENNARD, 1970; LEDERER *et al.*, 2018), although that does not mean that they necessarily always ‘select’ the causal one from two collinear features (ZOU and HASTIE, 2005).

Optimizing ML algorithms for predictive performance means that, generally, we should not assume that ML algorithms will learn the correct causal dependencies between the input features and the response (Box 2.6). It also means that we should not interpret xAI metrics as ‘effect estimates’ – if a certain feature is strongly used by the ML model, it could be because this feature has a biological or ecological effect on the response, but it could also easily be because it correlates with other features (e.g. GENUER, POGGI, and TULEAU-MALOT, 2010).

Nevertheless, there are interesting ideas for exploiting ML for (causal) data analysis. For example, if

we have a high dimensional dataset that would be difficult to analyse with conventional statistical tools, we could use ML and xAI to identify interesting patterns or features (LUCAS, 2020; PICHLER, BOREUX, *et al.*, 2020), and test the latter in a confirmatory analysis.

Moreover, based on a causal analysis, we could pre-select features that are consistent with ideas of causal inference. When selected in such a way, ML algorithm can achieve more exact control for confounders due to their greater flexibility (e.g. TANK *et al.*, 2021; WEIN *et al.*, 2021; ZEČEVIĆ *et al.*, 2021), and they can be used to estimate causal effects of treatments (CHERNOZHUKOV *et al.*, 2018; WAGER and ATHEY, 2018). Another interesting approach is to combine statistical models with ML and DL algorithms (JOSEPH, 2020a; MASAHIRO and RILLIG MATTHIAS, 2017; TANK *et al.*, 2021).

A third idea is to incorporate physical laws as constraints in the learning of neural networks. Such physics induced (or informed) neural networks (PINN, see KARNIADAKIS *et al.*, 2021) were originally developed to improve predictions but could also mark an important milestone in combining ML and causal inference by forcing the models to adhere to known physical or biological laws. This could be of interest for a field as E&E (WESSELKAMP *et al.*, 2022) that has acquired a lot of knowledge about various ecological systems over the years.

Finally, there is active research to develop ML methods that directly achieve causal discovery. For example, DL research has shown promising progress in symbolic regression where equations for systems are automatically inferred (CARDOSO *et al.*, 2020; D'ASCOLI *et al.*, 2022).

In summary, causal inference or causal discovery with ML is challenging, but there are various ways to combine ML methods with causal inference and based on the interest in this topic in the ML and DL field, but also the interest of ecologists to understand causal relationships in their data, we believe that the importance of this topic will increase in the coming years.

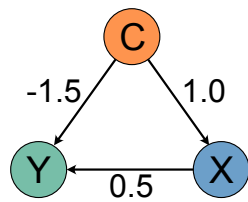
## 2.6 Conclusion

ML and DL algorithms are powerful and very general tools for predictive modelling and data analysis. Currently, DL algorithms have conquered image-based and similar tasks on complex, unstructured data, while classical ML algorithms such as RF and BRT still excel on structured data. The superior performance of ML and DL algorithms compared to statistical models can be explained by their higher flexibility and automatic data-dependent complexity optimization. For example, ML algorithms such as RF and BRT balance complexity by combining uncorrelated weak models into an ensemble, and DL uses a combination of indirect regularization through overparameterization stochastic gradient descent. Compared to classical statistical tools in E&E, ML methods are rather optimized for prediction, and caution is advised with their causal interpretations (Box 2.6).

A common challenge for both statistical models and ML alike is making predictions that extrapolate outside the feature space used to train the model (out-of-distribution predictions, see BEERY, VAN HORN, and PERONA, 2018; KOH *et al.*, 2021; Box 2.5). The reason why such predictions fail is that predictive models often learn to use non-causal proxies but also that relationships do not necessarily remain linear outside the area of data. Efforts have been made to identify such potentially unreliable predictions a priori (e.g. MEYER and PEBESMA, 2021) or to correct for them (e.g. TSENG, KERNER, and ROLNICK, 2022). Possibly, extending causal principles to ML (ZHAO and HASTIE, 2021), or incorporating ecological or mechanistic understanding, for example as additional model constraints, could help to improve model generalizability.

### Box 3.4: Predictive versus causal model building strategies

The best predictive model need not be the true causal model. To demonstrate this, we created a simulated dataset, where the response variable  $Y$  is affected by the feature  $X$  with a causal effect of 0.5 and by a second feature  $C$  with the causal effect of -1.5 (Figure 2.8).  $C$  also has a causal effect of 1.0 on  $X$ .  $X$  and  $C$  are thus highly correlated ( $>0.9$  Pearson correlation factor). We fitted different models (full model, model with only  $X$  or  $C$ , and full model with a ridge regularization ( $\lambda = 0.01$ ), Table 2.4) on 20 observations and estimated the prediction error (root-mean-square-error, RMSE) on 480 observations of the holdout. We repeated the simulation and the model fitting 10,000 times.



**FIGURE 2.8:** Small example. We are interested in how  $X$  affects  $Y$ .  $C$  is a confounder, i.e., affecting  $X$  and  $Y$ .  $C$  and  $X$  are highly correlated ( $> 0.9$  Pearson correlation factor). Numbers show the true effect estimates.

Model	X-estimate	C-estimate	RMSE on holdout
$Y \sim X + C$	0.5	-1.5	0.109
$Y \sim X$	-1.0	.	0.107
$Y \sim C$	.	-1.0	0.106
Ridge	-0.32	-0.59	0.106

**TABLE 2.4:** Results of different model specifications. We simulated 10,000 times from our small example (Figure 2.8) and fitted four different models on 20 observations: (i) The full model with the confounder ( $C$ ) and our variable of interest ( $X$ ), (ii) only our variable of interest, (iii) only the confounder, and (iv) the full model with a ridge regularization ( $\lambda = 0.01$ ). We evaluated the predictive error of the models by calculating the root-mean-squared error of the predictions for the holdout (480 observations).

*In causal inference*, the objective is to obtain correct effect estimates, which means that sometimes otherwise uninteresting collinear features must be included to control for confounding (variable  $C$ , Figure 2.4), while other structures, such as collider, must be excluded to obtain correct estimates. Thus, in causal analysis, the focus is to establish a correct hypothesis about the causal structure to obtain correct effect estimates (first model, Table 2.8). The causal structure can be based on logical considerations or causal discovery algorithms. Importantly, minimizing predictive error is not the primary goal of the analysis, and controlling confounders often increases uncertainties of the parameter estimates that propagate through the model and negatively affect the predictive error (note that the true causal model shows the highest RMSE, Figure 2.8).

*In predictive modelling*, our goal is to minimize the prediction error of our model. A common strategy is to provide the model with all the variables and use methods such as regularization or AIC selection to reduce model complexity and find the sweet spot of the bias variance tradeoff (see Figure 2.4 last model). In such an approach, collinear features are often removed because they increase uncertainties while contributing relatively little to the prediction, given that their effects can be ‘emulated’ by other features. In our simulation (Figure 2.8), we see that the true causal model has the highest prediction error, but correct effect estimates. The other models, have incorrect estimates but smaller prediction errors (Figure 2.8).

---

Much research in recent years has focused on making ML algorithms more transparent and bridging the gap between the properties of classical statistical tools and ML tools (e.g., xAI). New methods such as Bayesian neural networks have paved the way to obtain uncertainty and prediction intervals for DL models, bringing ML algorithms closer to statistical models (ASHUKHA *et al.*, 2021; LOQUERCIO, SEGU, and SCARAMUZZA, 2020). Despite that, their use for (causal) statistical inference is still controversial, not least because it is still unclear if ML can separate causal effects under feature collinearity, which is a basic requirement for causal inference.

Finally, despite the well-deserved attention, ML does not offer a free lunch. We have seen that the focus of ML methods on minimizing prediction comes at a cost elsewhere, such as in data requirements, interpretability or runtime. Even more strongly than statistical models, ML depends on the quality and the quantity of the data. Because of this, we should carefully consider whether the application of ML or even DL is necessary or promising for a task when simpler models with the advantages of higher interpretability, higher statistical power, and lower computational costs (and thus a better  $CO_2$  footprint, SCHWARTZ *et al.*, 2020) could do the job (MIGNAN and BROCCARDO, 2019). Nevertheless, we expect ML to become an indispensable tool in E&E, comparable to other traditional statistical tools such as linear regression models or analysis of variance models that have been used for many years.

**Acknowledgements** Maximilian Pichler received funding from the Bavarian Ministry of Science and the Arts in the Context of Bavarian Climate Research Network (bayklif). We thank Daniel Rettelbach and Tankred Ott, and two anonymous reviewers for their valuable comments and suggestions. Open Access funding enabled and organized by Projekt DEAL.

**Data availability statements** Code chunks for the different ML algorithms for different programming languages (R, Python, and Julia) can be found in the Supporting Information S1. Trend analysis (including the data) and scripts for reproducing the simulations and figures can be found in MAXIMILIANPI (2022).

---

## MACHINE LEARNING ALGORITHMS TO INFER TRAIT-MATCHING AND PREDICT SPECIES INTERACTIONS IN ECOLOGICAL NETWORKS

---

**Maximilian Pichler, Virginie Boreux, Alexandra-Maria Klein, Matthias Schleuning, Florian Hartig**

Published in *Methods in Ecology and Evolution*, 2020, 2, 11, 10.1111/2041-210X.13329

### Abstract

1. Ecologists have long suspected that species are more likely to interact if their traits match in a particular way. For example, a pollination interaction may be more likely if the proportions of a bee's tongue fit a plant's flower shape. Empirical estimates of the importance of trait-matching for determining species interactions, however, vary significantly among different types of ecological networks.
2. Here, we show that ambiguity among empirical trait-matching studies may have arisen at least in parts from using overly simple statistical models. Using simulated and real data, we contrast conventional generalized linear models (GLM) with more flexible Machine Learning (ML) models (Random Forest, Boosted Regression Trees, Deep Neural Networks, Convolutional Neural Networks, Support Vector Machines, naive Bayes, and k-Nearest-Neighbor), testing their ability to predict species interactions based on traits, and infer trait combinations causally responsible for species interactions.
3. We found that the best ML models can successfully predict species interactions in plant-pollinator networks, outperforming GLMs by a substantial margin. Our results also demonstrate that ML models can better identify the causally responsible trait-matching combinations than GLMs. In two case studies, the best ML models successfully predicted species interactions in a global plant-pollinator database and inferred ecologically plausible trait-matching rules for a plant-hummingbird network from Costa Rica, without any prior assumptions about the system.
4. We conclude that flexible ML models offer many advantages over traditional regression models for understanding interaction networks. We anticipate that these results extrapolate to other ecological network types. More generally, our results highlight the potential of machine learning and artificial intelligence for inference in ecology, beyond standard tasks such as image or pattern recognition.

**Keywords:** bipartite networks, causal inference, deep learning, hummingbirds, insect pollinators, machine learning, pollination syndromes, predictive modelling

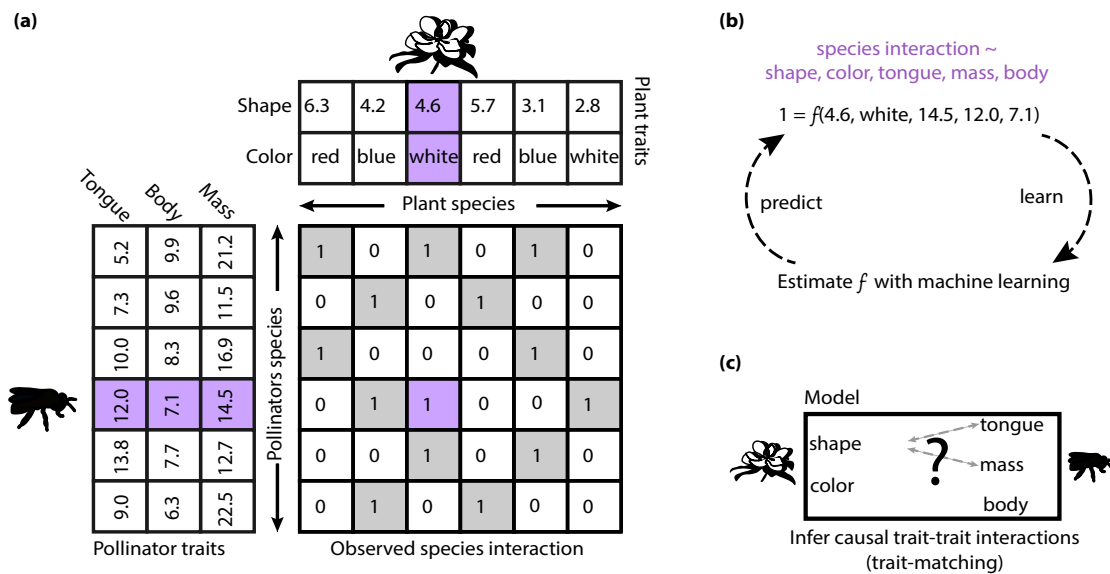
### 3.1 Introduction

The understanding and analysis of species interactions in ecological networks has become a central building block of modern ecology. Research in this field, however, has concentrated in particular on analyzing observed network structures (e.g. GALIANA *et al.*, 2018; GONZÁLEZ, DALSGAARD, and OLESEN, 2010; MORA *et al.*, 2018; POISOT, STOUFFER, and GRAVEL, 2015). Our understanding of why particular species interact, and others not, is comparatively less developed (cf. BARBEDO, 2018; POISOT, STOUFFER, and GRAVEL, 2015). A key hypothesis regarding this question is that species interact when their functional properties (traits) make an interaction possible (e.g. EKLÖF *et al.*, 2013; JORDANO, BASCOMPTE, and OLESEN, 2003). In plant–pollinator networks, for example, one would imagine that an interaction is easier to achieve when the tongue or body of the bee matches with the shape and size of the flower (GARIBALDI *et al.*, 2015; STANG, KLINKHAMER, and MEIJDEN, 2007). The idea that interactions will occur when traits are compatible is known as trait-matching (e.g. SCHLEUNING, FRÜND, and GARCÍA, 2015, see also Figure 3.1).

The assumption that trait-matching is important for species interactions is engraved in many other ecological ideas and hypotheses. For example, trait-matching is a prerequisite for the idea of pollination syndromes (i.e., the hypothesis that flower and pollinator traits co-evolve, FAEGRI and PIJL, 1979; see also FENSTER *et al.*, 2004; OLLERTON *et al.*, 2009; ROSAS-GUERRERO *et al.*, 2014b). Moreover, it has been suggested that trait-matching occurs also in other mutualistic ecological networks, for example fruit-frugivore interactions (e.g. DEHLING, TÖPFER, *et al.*, 2014), or antagonistic ecological networks, for example host-predator or host-parasitoid networks (GRAVEL *et al.*, 2013; EKLÖF *et al.*, 2013; see also EKLÖF *et al.*, 2013). Trait-matching between species has ample consequences for fundamental research, such as the identification and prediction of species interactions (BARTOMEUS *et al.*, 2016, see VALDOVINOS, 2019), but also impacts ecosystem management. For example, it could be used for identifying effective pollinators to optimize production of pollinator-dependent crops (GARIBALDI *et al.*, 2015; BAILES *et al.*, 2015, see POTTS *et al.*, 2016). Finally, explaining and predicting links between interaction partners from information about their properties has applications far beyond ecology. An example is molecular medicine, where analogue concepts are used to study gene association (e.g. LAARHOVEN and MARCHIORI, 2013; MENDEN *et al.*, 2013; YAMANISHI *et al.*, 2008; ZHANG, WANG, XI, *et al.*, 2018) or harmful drug–drug interactions (e.g. CHENG and ZHAO, 2014; TARI *et al.*, 2010).

While the idea of trait-matching itself is intuitive, it is less clear how important this mechanism is for determining species interactions (e.g. BARTOMEUS *et al.*, 2016; EKLÖF *et al.*, 2013). On the one hand, recent findings in plant–pollinator networks support the concept of pollination syndromes (ROSAS-GUERRERO *et al.*, 2014b) and the utility of syndromes for predicting or understanding species interactions (DANIELI-SILVA *et al.*, 2012; MURÚA and ESPÍNDOLA, 2015; FENSTER *et al.*, 2004, see GARIBALDI *et al.*, 2015). Recent studies also demonstrate that species interactions can be reasonably well predicted with phylogenetic predictors (BROUSSEAU, GRAVEL, and HANDA, 2018a; PEARSE and ALTERMATT, 2013; POMERANZ *et al.*, 2019), which supports the idea of trait-matching when assuming that traits are phylogenetically conserved. Similarly, studies of mutualistic pollination and seed-dispersal networks have accumulated evidence for strong signals of trait-matching, in particular in diverse tropical ecosystems (DEHLING, JORDANO, *et al.*, 2016; MAGLIANESI *et al.*, 2015). On the other hand, many ecological networks show low to moderate levels of specialization (BLÜTHGEN *et al.*, 2007) and high flexibility in partner choice (BENDER *et al.*, 2017), questioning the idea of strong co-evolutionary feedback loops between plants and animals (JANZEN, 1985; OLLERTON *et al.*, 2009). Moreover, while there is some direct evidence for trait-trait relationships as predictors for trophic interactions in simple prey-predator networks (GRAVEL *et al.*, 2013),





**FIGURE 3.1:** An illustration of the trait-matching concept. (a) Two classes of organisms, each with their own traits, interact in a bipartite network. (b) The goal of the statistical algorithm is to predict the probability of a plant-pollinator interaction, based on their trait values and (c) to infer the trait-trait interaction structure (trait-matching) that is causally responsible for those interactions

recent models that relied solely on trait-trait predictors (without phylogenetic predictors) showed only moderate performance in predicting species interactions (BROUSSEAU, GRAVEL, and HANDA, 2018a; POMERANZ *et al.*, 2019).

Progress on these questions is complicated by the fact that, until very recently, analyses of empirical networks relied almost exclusively on conventional regression models and phylogenetic predictors (BROUSSEAU, GRAVEL, and HANDA, 2018a; PEARSE and ALTERMATT, 2013; POMERANZ *et al.*, 2019), or on simple regression trees (e.g. BERLOW *et al.*, 2009). Reasonable doubts exist as to whether these models are flexible enough to capture the way traits give rise to interactions (see e.g. MAYFIELD and STOUFFER, 2017). Machine Learning (ML) models could be a solution to this problem. Modern ML models can flexibly detect interactions between predictors (trait-trait interactions), depend on fewer a-priori assumptions and usually achieve higher predictive performance than traditional regression techniques (e.g. BREIMAN, 2001b). State-of-the-art deep learning algorithms can detect complex pattern (e.g. LECUN, BENGIO, and HINTON, 2015) and excel in tasks such as image or species recognition (e.g. GRAY *et al.*, 2019; TABAK *et al.*, 2019). In food webs, recent findings demonstrate the potential of ML models in predicting species interactions. For example, DESJARDINS-PROULX *et al.* (2017) report that both a k-nearest neighbor and random forest (based on phylogenetic relationships and traits) can successfully predict food web interactions. It therefore seems promising to further explore the performance of machine learning algorithms for predicting species interactions from measurable traits, and whether those more flexible models change our view on the importance of trait matching for plant-pollinator interactions.

When assessing the suitability of ML algorithms for this problem, it is important to note that, while ML models tend to excel in predictive performance, their interpretation is often challenging (e.g. RIBEIRO, SINGH, and GUESTIN, 2016). Ecologists, however, would likely not be satisfied with predicting species interactions, but would also want to know which traits are causally

responsible for those interactions, for instance due to their importance as essential biodiversity variables (see KISSLING, WALLS, *et al.*, 2018). Unlike for statistical models, however, fitted ML models typically provide no direct information about how they generate their predictions. In recent years, also in response to issues such as fairness and discrimination (see OLHEDE and WOLFE, 2018), techniques aiming at interpreting fitted ML models have emerged (e.g. GUIDOTTI *et al.*, 2018). For example, permutation techniques (FISHER, RUDIN, and DOMINICI, 2018) allow estimating the importance of predictors for any kind of model, similar to the variable importance in tree-based models (BREIMAN, 2001a). In this case, however, we are not primarily interested in the effects of a single predictor, but we want to know how interactions between predictors (trait-trait matching) influence interaction probabilities. A suggested solution to this problem is the H-statistic (FRIEDMAN and POPESCU, 2008), which uses partial dependencies to estimate feature-feature (trait-trait) interactions from fitted ML models. Assuming that networks emerge due to a few important trait-trait interactions (EKLÖF *et al.*, 2013), the H-statistic should be able to identify those from a fitted ML, but to our knowledge, the efficacy of this or similar techniques in inferring causal traits has not yet been demonstrated.

The purpose of this study is to (a) systematically assess the predictive performance of different ML models for the identification of trait-matching in plant-pollinator networks and (b) to investigate if causal traits can be extracted from the fitted models with the H-statistics. We consider the most common ML models (k-nearest neighbor, random forest, boosted regression trees, deep neural networks, support vector machine, naive Bayes, and convolutional neural networks), with standard generalized linear model (GLM) as a benchmark. We apply all models to simulated and empirical plant-pollinator networks to establish how networks properties influence their predictive performance, and to test if the causally responsible trait-trait interactions be inferred from the fitted models. We ask the following questions: (1) Which algorithms display the highest predictive performance for simulated plant-pollinator networks, varying network sizes, observation times, and species abundances? (2) Can we retrieve the true underlying trait-trait interaction structure (trait-matching) in the simulated plant-pollinator networks from the fitted ML models? We demonstrate the practical utility of the developed methods by predicting interactions in a global crop-pollinator interaction database, and by inferring the causal trait-trait interaction structure in a Costa Rican plant-hummingbird network.

## 3.2 Material and Methods

### 3.2.1 Machine learning models for predicting species interactions from trait-matching

Throughout this paper, we consider that empirical observations of species interactions may be available either as binary (presence-absence) or weighted (counts, intensity, interaction frequencies) data. The objective for the models is to predict those plant-pollinator interactions based on the species' traits. We selected seven classes of ML models, either because they were previously used for trait-matching, or because the general ML literature suggests that they should perform well for this task (Table 3.1). For more details on the respective models, see the column 'Design principle' and the cited literature in Table 3.1, and the Supporting Information S2 in the Appendix.

Each of the ML models in Table 3.1 includes model-specific tuning parameters (so-called hyperparameters, for instance to control the model's learning behaviour) that can be adjusted by hand or optimized. To factor out idiosyncrasies due to the choice of these parameters, we optimized each models' hyperparameters with a random search in 30 (20 for empirical data) steps (see also

BERGSTRA and BENGIO, 2012), with nested cross-validations to avoid overfitting (for details see Appendix S3). Furthermore, ML models often perform poorly with imbalanced classes (proportion of plant-pollinator interactions to no plant-pollinator interactions is extremely low/high, KRAWCZYK, 2016). To address this, we applied the standard approach of oversampling observed plant-pollinator interactions when their proportion (compared to plant-pollinator pairs without an interaction) was lower than 20%. To compare ML with traditional regression models, we fitted GLMs (binomial GLM for presence-absence plant-pollinator interactions and Poisson GLM for plant-pollinator interaction counts), using all traits and all their possible two-way interactions as predictors and plant-pollinator interactions as response. Analyses were conducted with the statistical software R (R CORE TEAM, 2019). The r package mlr (BISCHL *et al.*, 2016, version 2.12) was used for hyperparameter tuning and cross validation of our ML models.

**TABLE 3.1:** Machine learning models and their usage for trait-matching

ML models	Type	Design principle	Applied with trait-matching
random forest (RF)	tree-based	Ensemble of a finite number of regression trees (see Breiman, 2001a).	DESJARDINS-PROULX <i>et al.</i> 2017; MASAHIRO and RILLIG MATTHIAS 2017; HU <i>et al.</i> 2016
boosted regression trees (BRT)	tree-based	After fitting the first weak regression tree to the response, subsequent regression trees are fitted on the previous residuals (see FRIEDMAN, 2001).	HE, HEIDEMEYER, <i>et al.</i> 2017; RAYHAN <i>et al.</i> 2017
k-nearest-neighbor (kNN)	distance-based	Given new point X, nearest k neighbors determine response.	DESJARDINS-PROULX <i>et al.</i> 2017 (as recommender system); RODGERS <i>et al.</i> 2010
support vector machines (SVM)	distance-based	In the n-dimensional feature space, a hyperplane to separate the classes is fitted (see CRISTIANINI, SHAWE-TAYLOR, <i>et al.</i> , 2000).	FANG <i>et al.</i> 2013
deep neural networks (DNN)	neural networks	By learning to represent the input over several hidden layers, they are able to identify the patterns in the data for the task	WEN <i>et al.</i> 2017
convolutional neural networks (CNN)	neural networks	Topological patterns in the input space (images, sequences) are preserved and processed by a number of kernels to extract features (see LECUN, BENGIO, and HINTON, 2015).	LIU, TANG, <i>et al.</i> 2016
naive Bayes	probabilistic classifier	The model learns the probability belonging to a class given a specific input vector.	FANG <i>et al.</i> 2013
GLM	parametric	A specific theory or model is fitted to the data	POMERANZ <i>et al.</i> 2019

### 3.2.2 Simulating plant-pollinator interactions

To assess predictive and inferential performance of the models, we created a minimal simulation model plant-pollinator interactions. The model assumes that the interaction probability between individuals of plants (group A) and pollinators (group B) arises from a Gaussian niche, matching the logarithmic ratio of the plant and pollinator traits. The logarithmic ratio ensures a symmetrically shaped interaction niche, see Figure S2.1. The niche value is multiplied by a weight to allow modifying the interaction strength independent of the niche width, and thus to control the overall trait-matching effect signal. Plant and pollinator abundances can either be drawn from an exponential distribution or a uniform distribution, to examine the effects of uneven abundance distributions and rare species. The expected number of observed interactions (i.e., their probability,  $P_{interaction}$ ) was then calculated as the interaction probability times the interaction partner's abundances times the observation time. Observation times were adjusted to standardize the proportion of plant-pollinator interactions to no plant-pollinator interactions. To create the final interaction counts, we sampled from a Poisson distribution with  $\lambda = P_{interaction}$ . For presence-absence species interactions ( $1 = interaction$ ,  $0 = nointeraction$ ), we set all *counts*  $> 0$  to 1.

Our default simulation scenario used 50\*100 (plants\*pollinators) for the simulated plant-pollinator

networks. To remove obstacles such as class imbalance, we adjusted the observation duration to have a class proportion of  $\approx 40\%$  for plant-pollinator interactions to no plant-pollinator interactions. The absence of interactions cannot be observed explicitly, and we speculate that most empirical datasets consist of observed species interactions (and possible non-interactions are inferred afterwards), thus we removed species with no observed plant-pollinator interaction.

### 3.2.3 Comparison of predictive performance

#### Predicting species interactions in simulated plant-pollinator networks

To assess predictive performance, we simulated reference data with six traits for each plant and pollinator. A possible issue with measuring predictive performance is that hidden correlations or structure in the data can lead to seemingly higher-than-random predictive performance even on random data (e.g. ROBERTS *et al.*, 2017). To check that this is not the case, we created a first baseline scenario, consisting of equal species abundances and no trait-trait interactions (no trait-matching, the latter was achieved by setting the trait-trait interaction niche extremely wide). A second issue is that interactions of rare species will be less frequent than those of abundant species. As a result, models can achieve higher-than-random performance even without any trait-trait interactions when species abundances are uneven. To ensure that the performance of our models exceeds these trivial performance levels, we created a second baseline scenario with exponential abundance distributions, but without trait-matching.

For the trait-matching scenario, we simulated networks with even abundance distributions and three trait-trait interactions (A1-B1, A2- B2, and A3-B3), each with a weight of 10. The scale parameter controlling the niche width was randomly sampled between 0.5 and 1.2 for simulating varying degrees of specialization in ecological networks (cf. BLÜTHGEN *et al.*, 2007). The even abundance distributions assumed here are unrealistic to some extent, but allow a better contrast between the models (because abundance effects are removed). In the case studies, we consider real abundance distributions. Other than that, the trait-matching scenario used the same parameter settings as the baseline scenarios (network size  $50 * 100$ ,  $\approx 40\%$  class balance). To test additionally for the effect of network sizes and observation time, we also varied network size to  $25 * 50$  and  $100 * 200$  (plants\*pollinators) setting and proportions plant-pollinator interactions to  $\approx 10\%$ ,  $\approx 25\%$ , and  $\approx 40\%$  one-factor-at-a-time from the base setting.

#### Case study 1 – Predicting plant-pollinator interactions

Our first case study uses data from a global database of crop-pollinator interactions, assembled from 1607 published studies from 77 countries worldwide (details see Data availability statement). Of these, we selected only crops that appeared at least two times at different geographical locations, resulting in 80 crops with 256 entries for pollinators.

The database lists five pollinator traits: guild (bumblebees, butterflies etc.), tongue length, body size, sociality (yes or no), and feeding behaviour (oligolectic, polylectic, or parasitic). In case of sexual dimorphism, the female measures were taken. Plants are described by 10 traits: type of plant (arboreous or herbaceous), flowering season, flower diameter, corolla shape (open, campanulate, or tubular), flower colour, nectar (yes or no), bloom system (type of pollination: insects, insects/wind, or insects/birds), self-pollination (yes or no), inflorescence (yes, solitary, solitary/pairs, solitary/clusters), and composite flowers (yes or no). Flower diameter, body size, and tongue length were provided as continuous traits (see Tables S2.1 and S2.2 for detailed information). When traits for a species were available from different sources, they were averaged. We filled missing trait values with a multiple imputation algorithm based on random forest (STEKHOVEN

and BÜHLMANN, 2012). We used all available traits as predictors in our models.

### **Measures of predictive performance**

To assess the models' predictive performance on the simulated plant-pollinator networks, we used the area under the receiver operating characteristic curve (AUC, measures how well the models are able to distinguish between plant-pollinator interaction and no plant-pollinator interaction regardless of classification threshold) and true skill statistic (TSS, which assess the predictive performance under a specific classification threshold, see ALLOUCHE, TSOAR, and KADMON, 2006) for presence-absence, and spearman's correlation for interaction frequencies. Because the TSS for the empirical plant-pollinator database (case study 1) was similar, we additionally calculated classification threshold-dependent performance measurements: accuracy (proportion of correct predicted labels), sensitivity (recall), precision, and specificity (true negative rate). Classification thresholds were optimized with TSS. The interpretation of these statistics is as follows: if our focus is to detect plant-pollinator interactions, we want to achieve a high true positive rate (sensitivity) with an acceptable rate of false positives in the as true predicted labels (precision). Specificity estimates the rate of true negatives of all predicted negatives (no plant-pollinator interaction).

#### **3.2.4 Measuring accuracy for inferring causal traits**

##### **H-statistics for inferring causal traits**

We used the H-statistic (FRIEDMAN and POPESCU, 2008) to infer causally responsible trait-trait interactions from the fitted ML models. The idea of this algorithm is similar to the principle of partial dependence plots. The H-statistic estimates the variance of the model's response caused by two traits separately (main effects) compared to the variance caused by the two traits combined partial function (trait-trait interaction). The H-statistic is scaled to [0,1]. A high value indicates that the interaction is the main reason for the variance in the response (probability for plant-pollinator interactions and counts for plant-pollinator interaction counts).

##### **Inferential performance in simulated plant-pollinator networks**

To assess the accuracy with which causal trait combinations can be identified from the fitted models via Friedman's H-statistic, we considered 25 \* 50 (plants\*pollinators) species networks with one, two, three and four trait-trait interactions (always six traits for each group, but varying number of trait-trait interactions that correspond to traitmatching), and equal interaction strength. We replicated the simulated plant-pollinator networks eight to ten times. The reason for choosing a smaller network size than for the predictive analysis was the computational cost of the H-statistics, which made applying a large number of replicates to larger networks computationally prohibitive.

The resulting networks had a 'real' observed size of 800–1,200 data points (we removed two networks with four true trait-trait interactions, because they had under 20 remaining samples after removing species with no plant-pollinator interactions at all). We fitted RF, BRT, DNN and kNN (the top predictive models) on the 76 simulated networks, 38 for presence-absence plant-pollinator interactions and 38 for plant-pollinator interaction counts (with uniform species abundances). For each sample, we calculated the H-statistic for all possible trait-trait interactions between the two species' groups. We calculated for each, the averaged true positive rate (true trait-trait interaction in found interactions with highest H-statistic) over the eight/ten repetitions. In a second step, based on our interim results (see results), we repeated the procedure with BRT

and DNN for 50 \* 100 (plants\*pollinators) simulated networks (see Appendix S3.1 for details regarding model fitting).

For GLMs, we selected the  $n$  ( $n$  = number of true trait-trait interactions) predictors with lowest p-value to calculate the true positive rate.

### Case study 2—Inferring trait-matching in a plant-hummingbird network

As a case study for inferring causally responsible traits, we used a dataset of plant-hummingbird interactions from Costa Rica. Plant-hummingbird networks are characterized by particularly strong signals of trait-matching (VIZENTIN-BUGONI, MARUYAMA, and SAZIMA, 2014). MAGLIANESI *et al.* (2014) filmed and analyzed plant-hummingbird interactions at three elevations in Costa Rica (700 hr of observations on 50 m a.s.l; 695 hr of observation on 1,000 m a.s.l; 727 hr of observations on 2000 m a.s.l). The resulting network consisted of 21 \* 8, 24 \* 8 and 20 \* 9 plant and hummingbird species, respectively. To predict plant-pollinator interactions, we used bill length, bill curvature, body mass, wing length, and tail length of hummingbirds, and corolla length, corolla curvature, inner corolla diameter width, and external corolla diameter width of plants. Flower volume was calculated by corolla length and external diameter (MAGLIANESI *et al.*, 2014). We used all available traits because the ML models should automatically learn trait-trait interactions.

We fitted the BRT with a Poisson maximum likelihood estimator and RF with a root mean squared error (RMSE) objective function (we did not log count data). We optimized DNNs with Poisson and negative binomial likelihood loss functions. We trained models on each elevation and on combined elevations (e.g. Low, Mid, High, Low-Mid-High, for details see Appendix S3.1). We calculated for the Low, Mid, High and Low-Mid-High models interaction strengths (H-statistics) for all possible trait-trait interactions (with trait-trait interactions within hummingbird/plant group). We checked the eight trait-trait interactions with highest interaction strengths for their biological plausibility by reviewing relevant literature.

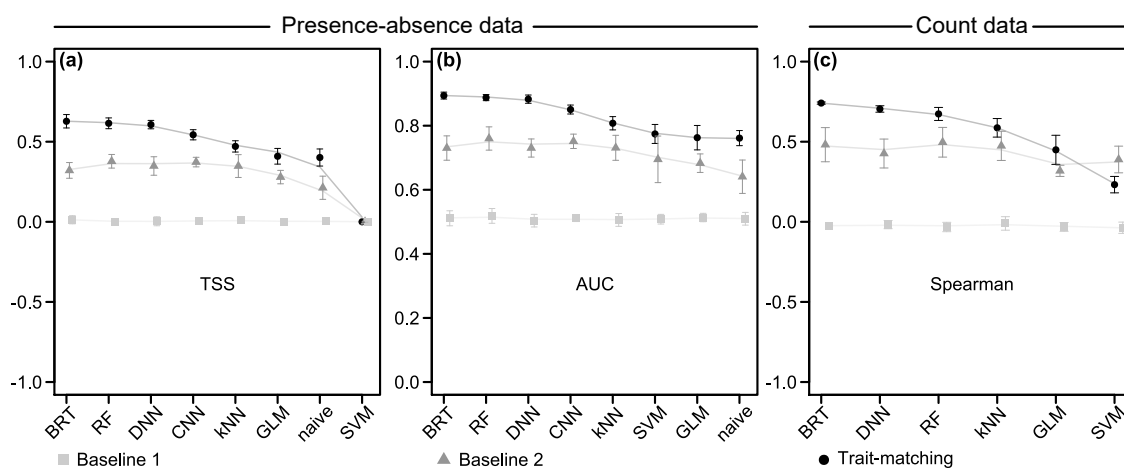
## 3.3 Results

### 3.3.1 Predictive performance

#### Predictive performance in simulated plant-pollinator networks

In the first baseline scenario (no trait-matching and equal species abundances), all models performed as expected for random plant-pollinator interactions, with AUC  $\approx$  0.5, TSS  $\approx$  0.0, and Spearman Rho factor  $\approx$  0.0 for both for presence-absence data and count data (Figure 3.2), indicating that our cross-validation setup is accurate. In the second baseline scenario (no trait-matching and networks with uneven species abundances), models achieved a TSS between 0.0–0.38, AUC between 0.64–0.76, and Spearman Rho factor of between 0.26–0.5 (Figure 3.2). The latter provides an indication, also with respect to existing literature, of what performance values can be achieved through imbalance of the data alone, even if there is no trait-matching. AUC, and Spearman Rho than for the baseline scenarios (Figure 2). Moreover, DNN, RF, and BRT achieved a higher TSS (0.61–0.63) than GLMs (0.41). SVM, naive Bayes, kNN were around GLM's performance or lower (Figure 3.2).

While all models improved their predictive performance with increasing network sizes with count data (Figure S2.2c), only DNN, RF, and BRT improved their performance with increasing network sizes with presence-absence plant-pollinator interactions (Figure S2.2a,b). Prolonging



**FIGURE 3.2:** Predictive performance of kNN, CNN, DNN, RF, BRT, naive Bayes, GLM and SVM with simulated plant-pollinator networks (50 plants \* 100 pollinators) for baseline scenarios with random interactions and even (baseline 1, squares) or uneven species abundances (baseline2, triangles, respectively), and trait-based interactions with even species abundances (circles). Predictive performance was measured by TSS (a) and AUC (b) for binary interaction data; and Spearman Rho factor (c) for interaction counts. Lowest predictive performance corresponds to zero for TSS, AUC, and Spearman Rho factor

the observation time (i.e., creating more plant-pollinator interactions and thus reducing data imbalance) generally increased the models' performances (Figure S2.2d-f).

### Predicting species interactions in a global crop-pollination database

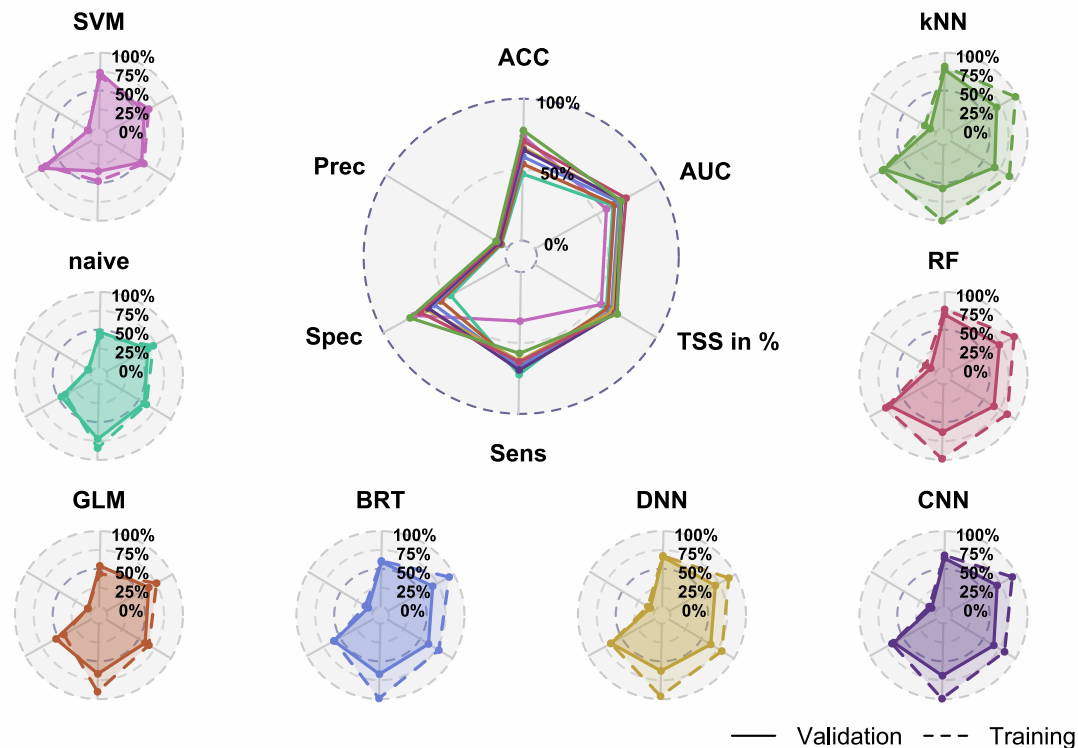
After fitting the models to real data from a global crop-pollination database, we calculated AUC, TSS and additional performance measures (Figure 3.3, Table S2.4) on the left-out samples. kNN achieved the highest TSS (0.36), RF achieved the highest AUC (0.73), and naive Bayes achieved highest TPR, followed by CNN. Overall, RF achieved the overall best predictive performance with highest AUC and second highest TSS (Figure 3.3, Table S2.4).

#### 3.3.2 Inference of causal trait-trait interactions

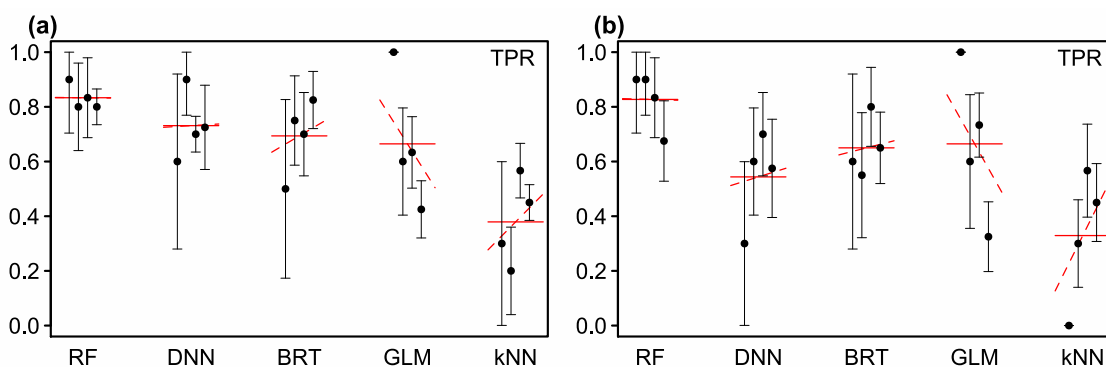
##### Inference of causal trait-trait interactions in simulated networks

In the second analysis step, we tested the ability of the H-statistics to infer the trait-trait interactions causally responsible for plant-pollinator interactions from the fitted models. In simulated networks, RF and BRT achieved highest true positive rates (Figure 3.4). For presence-absence plant-pollinator interactions, RF, DNN and BRT exceeded GLM performance with an averaged true positive rate of 70% to 80% over one to four true trait-trait matches (Figure 3.4a, the models were able to identify most of the true trait-trait interactions). For plant-pollinator interaction count data, only RF achieved a higher true positive rate than GLM (Figure 3.4b). However, it should be noted that the good GLM performance hinged on simulations with 1–3 trait-trait interactions and decreased most strongly of all algorithms with the number of trait-trait interactions (Figure 3.4).

When increasing network size (from 25 \* 50 to 50 \* 100), DNN and BRT improved their overall performance to 70%–95% and 87%–98% for presence-absence networks (Figure S2.3a), but showed a lower TPR for count data (Figure S2.3b).

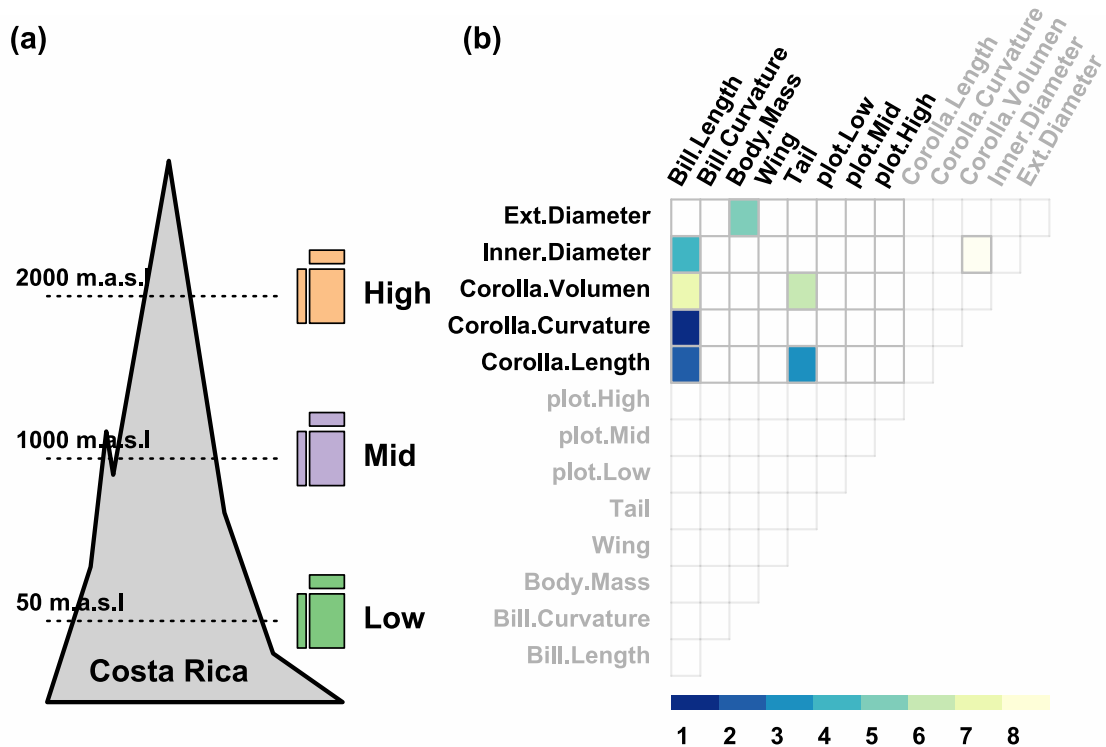


**FIGURE 3.3:** Predictive performance of different ML methods (naive Bayes, SVM, BRT, kNN, DNN, CNN, RF) and GLM in a global database of plant-pollinator interactions. Dotted lines depict training and solid lines validation performances. Models were sorted from left to right with increasing true skill statistic. The central figure compares directly the models' performances. Sen = Sensitivity (recall, true positive rate); Spec = Specificity (true negative rate); Prec = Precision; Acc = Accuracy; AUC = Area under the receiver operating characteristic curve (AUC); TSS in % = True skill statistic rescaled to 0–1



**FIGURE 3.4:** Comparison of the top predictive models' (RF, DNN, BRT, kNN, and GLM) abilities to infer the causal trait-trait interaction structure in simulated networks, using presence-absence data (a) and count data (b). The four values associated with each algorithm represent the mean true positive rate (TPR, dot) and its standard error (error bar) for the four interaction scenarios (one to four true trait-trait interactions in the simulations). The values were calculated based on 8–10 replicate simulations each. Solid red lines display the mean TPR across all four scenarios, dotted red lines show a linear regression estimate of TPR against the number of true trait-trait interactions





**FIGURE 3.5:** (a) Elevation profile for the three plant-hummingbird networks in Costa Rica (details see MAGLIANESI *et al.*, 2014). (b) The eight strongest trait- trait interactions (blue-yellow gradient) inferred with the H-statistic from RF models fitted to the combined plant- hummingbird network (colors code the ranking of strengths). Corolla length-bill length and corolla curvature-bill length had the highest interaction strength

### Inference of causal trait-trait interactions in a plant-hummingbird network

In a second case study, we computed interaction strength (H-statistic) for all possible trait-trait interactions in plant-hummingbird networks (Figure S2.5). The seven trait-trait interactions with highest interaction strength were identified by RF (Figure 3.5b). These interactions also achieved highest predictive performance (Figure S2.4). The four trait- trait interactions with highest interaction strength identified by BRT were in accordance with the ones that RF identified (Figure S2.5). RF and BRT identified corolla length-bill length, corolla curvature- bill length, inner diameter-bill length, and external diameter- body mass as the most important trait-trait interactions (Figure 3.5b, Figure S2.5). The models identified varying trait-trait interactions for networks at different elevations, but corolla and bill associations tended to be most important across elevations (Figure S2.5).

## 3.4 Discussion

We assessed the ability of seven ML models, plus GLM as a reference, to predict plant-pollinator interactions based on their traits. In a second step, we tested whether it is possible to identify the causally responsible trait-trait interaction structure (trait-matching) from the fitted models. Our main results are that the best ML models (RF, BRT, and DNN) outperform GLMs to a substantial degree in predicting plant-pollinator interactions from their traits, and that it is possible to identify the trait-trait interactions causally responsible for plant-pollinator interactions from the fitted

models with satisfying accuracy. The best ML models outperformed the simpler GLMs particularly for more complex trait-trait interaction structures, for which GLM performance dropped sharply.

### 3.4.1 Comparison of performance in predicting species interactions

In our analysis of predictive performance, we found that ML models such as RF, BRT and DNNs exceeded GLM performance for predicting plant-pollinator interactions from trait-matching data. They also worked surprisingly well with small network sizes ( $25 * 50$ , Figure S3.2a), such that performance did not increase substantially for larger networks ( $50 * 100$ ,  $100 * 200$ , Figure S2.2a-c).

An important point, also for comparing our performance indicators to the literature, is that all algorithms can achieve higher than naive random performance (e.g. AUC of 0.5) when species distributions are uneven, even when plant-pollinator interactions are not tied to traits (Figure 3.2). These results, in line with earlier findings (ADERHOLD *et al.*, 2012; CANARD *et al.*, 2014), highlight the importance of considering abundance when analyzing network structures: frequent species tend to have more observed interactions, and this effect might interfere with the trait-matching signal (OLITO and FOX, 2015). While the trait-matching effect may influence which plant-pollinator interaction is feasible, the species abundance effect determine the actual observed plant-pollinator interactions. Without adjusting observed plant-pollinator interactions for species abundances, it is difficult to separate the contributions of abundance and trait-matching to predictive performance (OLITO and FOX, 2015).

Observation time and type are further critical factors in ecological network analysis. Short observation times often lead to sparse networks with many unobserved plant-pollinator interactions, potentially creating biases in the analysis. Moreover, few plant-pollinator interactions result in data with imbalanced class distributions, presenting challenges for many ML methods (KRAWCZYK, 2016), which is also reflected in our results (Figure S2.2d-f). On the other hand, too long observation times could also negatively affect predictive performance, in particular when using binary links. The reason is that, given sufficient time, even weak links will be included in the network, potentially reducing the models' ability to identify the essential traits. Count data are more robust to these problems, and as our approach is equally applicable with count data, this data type seems generally preferable (DORMANN and STRAUSS, 2014).

While the ML models detected the important trait-trait interactions automatically, GLMs were pre-specified with all possible two-way trait-trait interactions. To check that the resulting high complexity did not disadvantage them unduly, we additionally confirmed that AIC selection on their interaction structure did not increase their predictive performance. We therefore believe that their lower performance is either explained by the fact that GLMs are not flexible enough to capture the complex form of the trait-matching structures (see MAYFIELD and STOUFFER, 2017), or that ML methods are more successful than AIC variable selection in addressing overfitting induced by the high combinatorial number of possible trait-trait interactions. These results mirror findings in the literature: while a few studies showed that GLM can predict species interactions based on trait-matching (e.g. GRAVEL *et al.*, 2013), most studies struggled in predicting species interactions with the trait-matching signal alone (e.g. BROUSSEAU, GRAVEL, and HANDA, 2018a; PEARSE and ALTERMATT, 2013; POMERANZ *et al.*, 2019). We speculate based on our results that previous studies based on GLMs may have underestimated the importance of trait-matching considerably, unless a very small number of trait-trait interactions (1–2) is dominantly responsible for the structure of the networks.

Previous studies often showed improved performance in predicting species interactions by using

phylogenetic predictors, serving as proxies for unobserved traits (see MORALES-CASTILLA *et al.*, 2015). The drawback, however, is that such phylogenetic proxies can be hard to interpret in the context of specific ecological hypothesis of why species interact (DÍAZ *et al.*, 2013). For example, a phylogenetic signal could arise both as a result of trait-matching (because traits tend to be phylogenetically conserved), or as a result other genetically coded preferences for particular interactions that are not accessible as traits. Based on our results, we expect that the relative importance of phylogenetic proxies will decrease when using appropriate ML models, which could help to better explore to what extent species interactions are determined by measurable functional traits.

We found that the models' predictive performance was lower for the empirical plant-pollinator database than for the simulated networks. There are several plausible reasons for this. Firstly, trait-matching rules may change over scales (POISOT, STOUFFER, and GRAVEL, 2015). As the database consists of globally observed plant-pollinator interactions, this may complicate the identification of a common trait-matching signal. Secondly, the high share of discrete predictors and high-class imbalance is likely to negatively affect the predictive performance. Despite these obstacles, kNN, RF, and CNN achieved  $>0.3$  TSS, and CNN and RF  $>70\%$  AUC (Figure 3.2, S2.4), much higher than null expectation, and consistent with results from the simulated networks. While it may be possible to improve GLM performance by manual selection of predictors, we also find that the case study highlights that algorithms such as RF and BRT are more parsimonious and robust in their use than a GLM which further suffered convergence problems.

### 3.4.2 Causal inference of trait-matching

To infer trait-trait interactions causally responsible for species interactions, we used the H-statistics. We found that this method, coupled with RF, DNN and BRT, could identify around 90% of the true trait-trait interactions in simulated plant-pollinator networks (Figure 3.4, Figure S2.5). Increasing the network size improved the detection accuracy of true trait-matches for BRT and DNN (Figure S2.3). When increasing the number of trait-trait interactions, the approach outperformed GLMs (Figure 3.2).

Our results demonstrate that identifying trait-matching from fitted models with the H-statistic works, but it also comes with drawbacks. The H-statistic depends on partial dependencies (FRIEDMAN and POPESCU, 2008) and is therefore sensitive to collinearity (see APLEY, 2016). Other alternative approaches (e.g. APLEY, 2016) might overcome this limitation. Moreover, the H-statistic is extremely computationally expensive, which is the reason why we tested it only on small network sizes ( $25 * 50$  species). Neither of these issues, however, would change the balance in favour of GLMs, which are prone to collinearity issues, too. To make sure that GLMs are not unjustly disfavoured, we additionally tested if AIC selection or choosing causal traits based on regression estimates instead of p-values would change the results, but neither improved inferential performance. In summary, we think that ML models are the better choice, not only for predictions, but also for causal inference in this setting. Future research should, however, focus on testing and advancing methods for the causal analysis of fitted models.

Analyzing plant-hummingbird networks with RF, we highlighted the seven trait-trait interactions with highest interaction strength (Figure 5b). The inferred trait-trait interactions are highly plausible for the following reasons: (a) RF showed high accuracy with low consistent errors in the simulated networks (Figure 4). (b) The identified trait-trait interactions are ecologically plausible (Figure 5b): Trait matches with highest interaction strength (corolla length-bill length and corolla curvature-bill length) are in line with previous findings that emphasize their importance in

plant-hummingbird networks (TEMELES *et al.*, 2009; MAGLIANESI *et al.*, 2014; VIZENTIN-BUGONI, MARUYAMA, and SAZIMA, 2014; WEINSTEIN and GRAHAM, 2017). Collinearity of traits likely explains other matches. For instance, body mass is positively correlated with tail length, explaining why corolla volume was associated with tail length. These results further support the view that it is possible to infer trait-matching with ML in ecologically realistic settings, without a priori assumptions.

Estimated trait-trait interactions in the plant-hummingbird networks differed for the three elevations, but the match of corolla length-bill length was generally most important (Figure S2.5). MAGLIANESI *et al.* (2014) and MAGLIANESI *et al.* (2015) reported similarly varying trait-trait interactions in plant-hummingbird networks across elevations, consistent with our results. While interactions in ecological networks vary over scales (POISOT, STOUFFER, and GRAVEL, 2015), a common backbone is assumed (MORA *et al.*, 2018). With corolla length-bill length, identified by RF and BRT with highest interaction strength (Figure 3.5b, Figure S2.5), we speculate that we identified with ML the central trait-matching phenomenon in plant-hummingbird networks.

### 3.5 Conclusion

In conclusion, our study demonstrates that RF, BRT, and DNN exceeded GLM performance in predicting plant-pollinator interactions from trait information. ML models could also identify causally responsible trait-trait interactions with a higher accuracy than GLMs. The ability to automatically extract species interactions from observed networks and traits, and causally interpreting the underlying trait-trait interactions, makes our approach, which we provide in an R package, a powerful new tool for ecologists.

While we considered only plant-pollinator networks in this study, our method could be applied to other types of species interaction networks such as any mutualistic and antagonistic interactions in complex food webs (this is also supported by DESJARDINS-PROULX *et al.*, 2017). In either of these ecological network types, there are ample opportunities for further analyses, for example how species interactions will change under global change or how species interactions will rewire in novel communities with reshuffled species and trait composition (BAILES *et al.*, 2015, see KISSLING and SCHLEUNING, 2015). By identifying crucial rules of trait-matching between species, our approach can give insights into how biotic interactions shape community assembly and also contribute to the identification of Essential Biodiversity Variables in the context of global change (KISSLING, WALLS, *et al.*, 2018).

**Acknowledgements** Maria A. Maglianesi recorded interaction and trait data of plants and hummingbirds in Costa Rica. We would like to thank Johannes Oberpriller and Lukas Heiland, as well as Roozbeh Valavi and an anonymous reviewer for their valuable comments and suggestions. V.B. acknowledges funding for the assembly of the global plant-pollinator database (case study 1) by Bayer CropScience.

**Data availability statements** The plant-hummingbird data associated with this study is available at <https://doi.org/10.6084/m9.figshare.3560895.v1>. The global plant-pollinator database used with this study is available at <https://doi.org/10.6084/m9.figshare.9980471.v1>. The analysis and the Trait-matching R package is available at <https://doi.org/10.5281/zenodo.3522854> (<https://github.com/TheoreticalEcology/Pichler-et-al-2019>).

---

## A NEW JOINT SPECIES DISTRIBUTION MODEL FOR FASTER AND MORE ACCURATE INFERENCE OF SPECIES ASSOCIATIONS FROM BIG COMMUNITY DATA

---

**Maximilian Pichler and Florian Hartig**

Published in *Methods in Ecology and Evolution*, 2021, 11, 12, 10.1111/2041-210X.13687

### **Abstract**

1. Joint species distribution models (JSDMs) explain spatial variation in community composition by contributions of the environment, biotic associations and possibly spatially structured residual covariance. They show great promise as a general analytical framework for community ecology and macroecology, but current JSDMs, even when approximated by latent variables, scale poorly on large datasets, limiting their usefulness for currently emerging big (e.g., metabarcoding and metagenomics) community datasets.
2. Here, we present a novel, more scalable JSDM (sjSDM) that circumvents the need to use latent variables by using a Monte Carlo integration of the joint JSDM likelihood together with flexible elastic net regularization on all model components. We implemented sjSDM in PyTorch, a modern machine learning framework, which allows making use of both CPU and GPU calculations. Using simulated communities with known species-species associations and different number of species and sites, we compare sjSDM with state-of-the-art JSDM implementations to determine computational runtimes and accuracy of the inferred species-species and species-environment associations.
3. We find that sjSDM is orders of magnitude faster than existing JSDM algorithms (even when run on the CPU) and can be scaled to very large datasets. Despite the dramatically improved speed, sjSDM produces more accurate estimates of species association structures than alternative JSDM implementations. We demonstrate the applicability of sjSDM to big community data using eDNA case study with thousands of fungi operational taxonomic units (OTU).
4. Our sjSDM approach makes the analysis of JSDMs to large community datasets with hundreds or thousands of species possible, substantially extending the applicability of JSDMs in ecology. We provide our method in an R package to facilitate its applicability for practical data analysis.

**Keywords:** big data, co-occurrence, machine learning, metacommunity, regularization, statistics

## 4.1 Introduction

Understanding the structure and assembly of ecological communities is a central concern for ecology, biogeography and macroecology (VELLEND, 2010). The question is tightly connected to important research programs of the field, including coexistence theory (see CHESSON, 2000, e.g. LEVINE *et al.*, 2017), the emergence of diversity patterns (e.g. PONTARP *et al.*, 2019) or understanding ecosystem responses to global change (URBAN *et al.*, 2016).

The statistical analysis of spatial community data is currently dominated by two major ecological frameworks: metacommunity theory (see LEIBOLD, HOLYOAK, *et al.*, 2004) and species distribution models (SDMs ELITH and LEATHWICK, 2009). Metacommunity theory formed in the last two decades as the study of the spatial processes that give rise to regional community assembly (e.g. LEIBOLD and CHASE, 2017; LEIBOLD, HOLYOAK, *et al.*, 2004). Most current metacommunity analyses rely on ordination (LEIBOLD and MIKKELSON, 2002, e.g.) or variation partitioning (e.g. COTTENIE, 2005) techniques, which disentangle abiotic and spatial contributions to community assembly (see LEIBOLD and CHASE, 2017). SDMs are statistical models that link abiotic covariates to species occurrences. They are widely used in spatial ecology, for example to study invading species (GALLIEN *et al.*, 2012; MAINALI *et al.*, 2015) or species responses to climate change (THUILLER *et al.*, 2006).

A key limitation of both variation partitioning and SDMs, noted in countless studies, is that they do not account for species interactions. Both approaches essentially assume that species depend only on space and the environment (COTTENIE, 2005; DORMANN, SCHYMANSKI, *et al.*, 2012; PERES-NETO and LEGENDRE, 2010; WISZ *et al.*, 2013), whereas we know that in reality, species can also influence each other through competition, predation, facilitation and other processes (GILBERT and BENNETT, 2010; VAN DER PUTTEN, MACEL, and VISSER, 2010, see MITTELBACH and SCHEMSKE, 2015, see LEIBOLD, HOLYOAK, *et al.*, 2004).

Joint species distribution models (JSDM) recently emerged as a novel analytical framework promising to integrate species interactions into metacommunity and macroecology (LEIBOLD, RUDOLPH, *et al.*, 2022). JSDMs are similar to SDMs in that they describe species occurrence as a function of the environment, but additionally consider the possibility of species-species associations. By an association, we mean that two species tend to appear together more or less often than expected from their environmental responses alone. Whether those association originate from biotic interactions (e.g., competition, predation, parasitism, mutualism) or other reasons (e.g., unmeasured environmental predictors) needs to be carefully considered (see BLANCHET, CAZELLES, and GRAVEL, 2020; DORMANN, BOBROWSKI, *et al.*, 2018; KÖNIG *et al.*, 2021). Still, when appropriately interpreted, JSDMs combine the essential processes believed to be responsible for the assembly of ecological communities—environment, space and biotic interactions—and they could be applied to large scale as well as for regional metacommunity analyses (e.g. GILBERT and BENNETT, 2010; LEIBOLD, HOLYOAK, *et al.*, 2004; MITTELBACH and SCHEMSKE, 2015).

Recent interest in JSDMs was further fuelled by the emergence of high-throughput technologies that are currently revolutionizing our capacities for observing community data (e.g. PIMM *et al.*, 2015). We can now detect hundreds or even thousands of species from environmental DNA (eDNA) or bulk-sampled DNA (YU, JI, EMERSON, *et al.*, 2012; BOHMANN *et al.*, 2014; CRISTESCU, 2014; DEINER *et al.*, 2017; BÁLINT *et al.*, 2018; BARSOU *et al.*, 2019; HUMPHREYS *et al.*, 2019; TIKHONOV, DUAN, *et al.*, 2019) in a given sample, and next generation sequencing (NGS) has become cheap enough that this process could be replicated at scale. Other emerging technologies will likely also produce large amounts of community data, such as automatic species recognition

(GUIRADO *et al.*, 2018, e.g. TABAK *et al.*, 2019) from acoustic recordings. Recent studies have used these methods to generate community inventories of fish (see (DESJONQUÈRES, GIFFORD, and LINKE, 2019), e.g. PICCIULIN *et al.*, 2019), forest wildlife (e.g. WREGE *et al.*, 2017), bird communities (FRITZLER, KOITKA, and FRIEDRICH, 2017; LASSECK, 2018; WOOD, GUTIÉRREZ, and PEERY, 2019) or bats (e.g. MAC AODHA *et al.*, 2018). Jointly, these developments mean that large spatial community datasets will become available in the near future, and ecologists have to consider how to best analyse them.

Joint species distribution models would seem the natural analytical approach for these emerging new data, given their ability to separate the essential processes for spatial community assembly. Current JSDM software, however, has severe limitations for processing such large (and/or wide) datasets. Early JSDMs were based on the multivariate probit (MVP) model (CHIB and GREENBERG, 1998), which describes species-species associations via a covariance matrix (e.g. CLARK, GELFAND, *et al.*, 2014; GOLDING and HARRIS, 2015; HUI, 2016; OVASKAINEN, HOTTOLA, and SIITONEN, 2010; POLLOCK *et al.*, 2014). The limitation of the MVP approach is that it scales poorly for species-rich data, as the number of parameters in the species-species covariance matrix increases quadratically with the number of species (see WARTON *et al.*, 2015).

The current solution to this problem is latent variable models (LVMs), which replace the covariance matrix with a small number of latent variables (see WARTON *et al.*, 2015). The LVM reparameterization makes the estimation of MVP models computationally more efficient (see NIKU, HUI, *et al.*, 2019; NORBERG *et al.*, 2019; OVASKAINEN, TIKHONOV, *et al.*, 2017; TIKHONOV, ABREGO, *et al.*, 2017; TIKHONOV, OPEDAL, *et al.*, 2020; WARTON *et al.*, 2015). That, however, does not mean that simultaneously estimating species' abiotic preferences and species-species associations with LVMs is fast. Integrating out the latent variables requires MCMC sampling or numerical approximations (e.g. Laplace, variational inference, see NIKU, HUI, *et al.*, 2019), which is computationally costly and can fail to converge. For communities with hundreds of species, computational runtimes of current LVMs can still exceed hours or days (e.g. TIKHONOV, OPEDAL, *et al.*, 2020; WILKINSON *et al.*, 2019). This poses severe limitations for analysing eDNA data, which can include thousands of species or operational taxonomic units (OTUs, e.g. FRØSLEV *et al.*, 2019). Moreover, LVMs also scale disadvantageously with the number of sites, because each site introduces additional parameters in the latent variables (BARTHOLOMEW, KNOTT, and MOUSTAKI, 2011; SKRONDAL and RABE-HESKETH, 2004). Thus, the advantage of the LVM over the full-MVP model decreases with increasing numbers of sites (on the order of thousands). An important challenge for the field is therefore to make JSDMs fast enough for big datasets (KRAPU and BORSUK, 2020).

A second question for JSDM development is the accuracy of inferred species associations. Surprisingly little is known about this question. Most existing JSDM assessments (NORBERG *et al.*, 2019, e.g. TOBLER *et al.*, 2019, (WILKINSON *et al.*, 2019)) concentrate on runtime, predictive performance or on aggregated measures of accuracy that do not necessarily capture the error of the estimated species-species association structure (but see ZURELL, POLLOCK, and THUILLER, 2018). From a statistical perspective, however, it is clear that estimating a large species covariance matrix with limited data must have considerable error, and it would be desirable to better understand the dependence on this error on the structure of the data and the chosen modelling approach.

The LVM approach, specifically, not only makes the models faster, but also reduces the number of free parameters (WARTON *et al.*, 2015), which should theoretically reduce the variance (and thus the error) of the species-species covariance estimates, but possibly at the cost of a certain bias. The trade-off between bias and variance is controlled by the number of latent variables—when

the number of latent variables is similar to the number of species, the LVM will be as flexible (and unbiased) as the full-MVP model. The fewer latent variables are used, the stronger the reduction in variance and the potential increase in bias. In practice, the number of latent variables is usually chosen much smaller than the number of species (the highest value we saw was 32 with hundreds of species in TIKHONOV, DUAN, *et al.*, 2019), which means that JSDMs fitted currently by LVMs could show biases due to the regularization induced by the LVM structure (STEIN, 2014).

While trading off some bias against a reduction in variance is fundamental to all regularization approaches, and no concern as such, it seems important to understand the nature of the bias that is created by the LVM structure and examine if alternative regularization structures are more appropriate. Similar to LVMs, spatial models for large data often use a low-rank approximation of the covariance matrix (e.g. STEIN, 2007; STEIN, 2014, e.g. SANG, JUN, and HUANG, 2011). For Gaussian process models, it has been shown that this approximation captures the overall structure well (in the sense that the magnitude of covariances is captured well), but at the costs of larger errors in local structures (see STEIN, 2014). We conjecture that LVMs with a small number of latent variables behave analogous—with a few latent variables, it will be difficult to model a specific covariance structure without unintentionally introducing other covariances elsewhere, but it could be possible to generate a good approximation of the overall correlation level between species.

Here, we propose a new method for estimating JSDMs, called scalable JSDM, that addresses many of the above-mentioned problems. By using a Monte Carlo approach [originally proposed by CHEN, XUE, and GOMES (2018)] that can be outsourced to graphical processing units (GPUs), sjSDM is able to fit JSDMs with a full covariance structure extremely fast, without having to resort to latent variables. To address the issue of overfitting due to the increased number of parameters compared to state-of-the-art latent variable JSDMs, we introduce a new regularization approach, which directly targets the covariance matrix of the full-MVP model. Additionally, we propose a method for optimizing the regularization strength based on tuning the parameter under a  $k$ -fold cross-validation.

To demonstrate the beneficial properties of the new model structure, we assess: (a) its computational runtime on GPUs and CPUs, (b) the accuracy of inferred species-species associations and species' environmental responses and (c) its predictive performance. We compare the performance of sjSDM to several state-of-the-art JSDM software packages (Hmsc, gllvm and BayesComm; see also HARRIS, 2015; CLARK, NEMERGUT, *et al.*, 2017; VIEILLEDENT and CLÉMENT, 2019), as well as results from a recent JSDM comparison (WILKINSON *et al.*, 2019). Finally, to illustrate the applicability of our approach to wide community data, we additionally applied our model to a community eDNA dataset containing 3,649 fungi OTUs over 125 sites.

## 4.2 Material and Methods

### 4.2.1 The structure of the JSDM problem

Species-environment associations are classically addressed by SDMs, which estimate the expected probability of the presence of a species as a function of the environmental predictors. The functional response to the environment can be expressed by GLMs, or by more flexible (i.e., nonlinear and/or nonparametric) approaches such as generalized additive models, boosted regression trees or Random Forest (ELITH and LEATHWICK, 2009; INGRAM, VUKCEVIC, and GOLDING, 2020).

A JSDM generalizes this approach by including the possibility of residual species-species corre-



lations (in the literature usually called species-species associations). The most common JSDM structure is the MVP model, which describes the site by species matrix  $Y_{ij}$  ( $Y_{ij} = 1$  if species  $j = 1, \dots, J$  is present at site  $i = 1, \dots, I$  or  $Y_{ij} = 0$  if species  $j$  is absent) as a function of the environmental covariates  $X_{in}$  ( $n = 1, \dots, N$  covariates), and the covariance matrix (species associations)  $\Sigma$  accounts for correlations in  $e_{ij}$ :

$$\begin{aligned} Z_{ij} &= \beta_{j0} + \sum_{n=1}^N X_{in}\beta_{nj} + e_{ij}, \\ Y_{ij} &= 1(Z_{ij} > 0), \\ \mathbf{e}_i &\sim MVN(0, \Sigma) \end{aligned} \tag{4.1}$$

For the results, we normalized the fitted species-species covariance matrix  $\Sigma$  to a correlation matrix.

#### 4.2.2 Current approaches to fit the JSDM structure

The model structure described in Equation 1 can be fitted directly using the probit link, and the first JSDMs used this approach (CHIB and GREENBERG, 1998, POLLOCK *et al.*, 2014, see WILKINSON *et al.*, 2019). Fitting the full-MVP model directly, however, has two drawbacks: first, calculating likelihoods for large covariance matrices is computationally costly. Second, the number of parameters in the covariance matrix for  $j$  species increases quadratically as  $j \cdot (j - 1)/2$ . For 50 species, for example, we would have to estimate 2,250 covariance parameters.

Because of these problems, a series of papers (OVASKAINEN, ABREGO, *et al.*, 2016; WARTON *et al.*, 2015) introduced the LVM (see SKRONDAL and RABE-HESKETH, 2004) to the JSDM problem. The latent variable JSDM approximates the species-species covariance by introducing a number of latent covariates (=latent variables), which act exactly like real environmental covariates, except that their values are estimated as well. Species that react (via their factor loadings) similarly or differently to the latent variables thus show positive or negative associations respectively (see OVASKAINEN, TIKHONOV, *et al.*, 2017; WARTON *et al.*, 2015; WILKINSON *et al.*, 2019, for details). The factor loadings can be translated into a species-species covariance matrix:  $\Sigma = \lambda \cdot \lambda^T$  ( $\lambda$  = matrix of factor loadings). The latent variables can be interpreted as unobserved environmental predictors, but they can also be viewed as a purely technical approach to regularized low-rank reparameterization of the covariance matrix. One advantage of the LVM is that the latent variables can be used for constrained (LVM with environmental predictors) and unconstrained ordination (LVM without environmental predictors; WARTON *et al.*, 2015). The complexity of the association structure can be set via the number of latent variables (usually to a low number, see WARTON *et al.*, 2015).

#### 4.2.3 An alternative approach to fit the JSDM structure

Because LVMs still have computational limitations, and because of the need for flexible regularization discussed in the Introduction, we propose a different approach to fit the model structure in Equation 4.1. The full-MVP assumes that the observed binary occurrence vector  $Y_i \in (0, 1)^J$  arises as the sign of the latent Gaussian variable  $Y_i^* \sim N(\mathbf{X}_i\beta, \Sigma)$ :

$$Y_{ij} = \mathbb{1}(Y_{ij}^*) \tag{4.2}$$

where  $\beta$  is the environmental coefficient matrix and  $\Sigma$  the covariance matrix. Then the probability to observe  $\mathbf{Y}_i$  is:

$$\Pr(\mathbf{Y}_i | \mathbf{X}_i \beta, \Sigma) = \int_{A_{i1}} \cdots \int_{A_{iJ}} \phi_J(\mathbf{Y}_i^*; \mathbf{X}_i \beta, \Sigma) d\mathbf{Y}_{i1}^* d\mathbf{Y}_{iJ}^* \quad (4.3)$$

with the interval  $A_{ij}$  defined as:

$$D_{it} = \begin{cases} (-\inf, 0) & \text{if } Y_{ij} = 0 \\ [0, +\inf) & \text{if } Y_{ij} = 1 \end{cases} \quad (4.4)$$

and  $\phi$  being the density function of the multivariate normal distribution. The main computational issue of the full-MVP (Equation 4.3) is that calculating the probability of  $\mathbf{Y}_i$  requires to integrate over  $\mathbf{Y}_i^*$ , which has no closed analytical expression for more than two species ( $J > 2$ ). This makes the evaluation of the likelihood computationally costly when  $J \gg 1$  and motivates the search for an efficient numerical approximation of Equation 4.3.

To see how this approximation can be achieved, note that Equation 4.3 can be expressed more generally as:

$$\mathcal{L}(\beta, \Sigma; \mathbf{Y}_i, \mathbf{X}_i) = \int_{\Omega} \prod_{j=1}^J \Pr(Y_{ij} | \mathbf{X}_i \beta + \xi) \Pr(\xi | \Sigma) d\xi \quad (4.5)$$

In sjSDM, we approximate this integral by  $M$  Monte Carlo samples from the multivariate normal species-species covariance. With the covariance term being integrated out, we can calculate the remaining part of the likelihood as in a univariate case, and use the average of the  $M$  samples to get an approximation of Equation 4.5:

$$\mathcal{L}(\beta, \Sigma; \mathbf{Y}_i, \mathbf{X}_i) \approx \frac{1}{M} \sum_{m=1}^M \prod_{j=1}^J \Pr(Y_{ij} | \mathbf{X}_i \beta + \xi_m) \quad (4.6)$$

$\xi_m \sim MVN(0, \Sigma)$

This approximation of the MVP was first proposed by CHEN, XUE, and GOMES (2018) in the context of fitting deep neural networks with an MVP response structure. Its most notable computational advantage over other existing approximations to the MVP problem, such as the Geweke-Hajivassiliou-Keane (GHK) algorithm (HAJIVASSILIOU and RUUD, 1994), is that the Monte Carlo sampling in Equation 4.6 can be parallelized. This is especially efficient when performing calculations on GPUs rather than CPUs, due to their much higher number of cores (see also GOLDING, 2019, who similarly uses GPUs to improve an expensive computational tasks in ecology). The GHK algorithm, on the other hand, is based on a recursive and thus non-independent importance sampling procedure, which means that the sampling cannot be parallelized.

For sjSDM, we implemented this approximation, which was previously only used in the deep learning literature, to the generalized linear MVP, which means that we conform to the model structure typically used in this field and can profit from all benefits associated with parametric

models. The only difference to a standard MVP is that we use an approximation of the probit link, which we found to be numerically more stable than the analytical probit link (see Supporting Information S3 for details).

We implemented the method in an R package (<https://github.com/TheoreticalEcology/s-jSDM>), using the Python package PyTorch, which is designed for deep learning (PASZKE, GROSS, MASSA, *et al.*, 2019), and the R package reticulate, which allows us to run PyTorch from within R (USHEY, ALLAIRE, and TANG, 2019). This setup allows us to leverage various sophisticated numerical algorithms from PyTorch, including the possibility to switch between efficiently parallelized CPU and GPU calculations, and the ability to obtain analytical gradients (via automatic derivatives) of the MVP likelihood with the latent covariance structure marginalized out via the Monte Carlo ensemble. The combination of efficient parallelization and analytical derivatives of the Monte Carlo approximated likelihood makes finding the maximum likelihood estimate (MLE) for the full-MVP model extremely fast, despite the large number of parameters to optimize.

Outsourcing the Monte Carlo approach to a GPU solves the issue of computational speed (as we show below), but it does not yet solve the problem that the covariance matrix has a very large number of parameters, which raises the problem of overfitting when the method is used on small datasets. To address this, we penalized the actual covariances in the species-species covariance matrix, as well as the environmental predictors, with a combination of ridge and lasso penalty (elastic net, see ZOU and HASTIE, 2005, more details below). Our R package includes a function to tune the strength of the penalty for each model component separately via cross-validation.

The here-tested implementation of sjSDM only considers binary (presence-absence) data, but there are several ways to extend the approach to count and proportional data as well. To a large extent, these are already implemented in our R package. Currently supported are count (Poisson distribution with log-link), presence-absence (binomial distribution with logit and probit links) and normal data (multivariate normal distribution). Moreover, the model-based ordination that is popular for latent variable JSDMs is currently not implemented in sjSDM and probably challenging to achieve, since the model is fit without latent environmental variables. However, new ordination techniques with a focus on co-occurrence patterns (e.g. POPOVIC *et al.*, 2019) could complement sjSDM in practical analyses.

#### 4.2.4 Benchmarking our method against state-of-the-art JSDM implementations

To benchmark our approach, we fit sjSDM to six datasets from a recent JSDM benchmark study by WILKINSON *et al.* (2019) (Table S3.1). Covariates were centred and standardized. To be able to compare our results to theirs across different hardware, we also reran BayesComm, the fastest JSDM in their study, with the same parameters as in the study by WILKINSON *et al.* (2019) on our hardware.

Additionally, we simulated new data from an MVP (Equation 4.2), varying the number of sites from 50 to 500 (50, 70, 100, 140, 180, 260, 320, 400, 500) and the number of species as a percentage (10%, 30% and 50%) of the sites (e.g., the scenario with 100 sites and 10% results in 10 species). In all simulations, the species' environmental preference was described for five environmental covariates (beta), which was randomly selected. Each scenario was sampled five times. Here, all species had species-species associations, that is the species-species covariance matrices were not sparse (for details, see Supporting Information S3).

To compare our model to existing JSDM software packages, we selected BayesComm (version 0.1-2

GOLDING and HARRIS, 2015), the fastest MVP-based JSJM according to the study by WILKINSON *et al.* (2019), and two state-of-the-art latent-variable JSJM implementations: Hmsc (version 3.0-4 TIKHONOV, OVASKAINEN, *et al.*, 2019), which uses MCMC sampling, and gllvm (version 1.2.1 NIKU, BROOKS, *et al.*, 2020), which uses variational Bayes and Laplace approximation to fit the model. We used the default parameter settings for all three methods which were in line with other recent JSJM benchmarks (details see Supporting Information S3).

Since GPUs might be not commonly available, we calculated sjSDM results both on the CPU and on the GPU. To estimate the influence of the number of Monte Carlo samples on the error of the MVP approximation, we used 100 Monte Carlo samples for each species when run on the CPU and 1,000 Monte Carlo samples for each species when run on the GPU for sjSDM. In the following, we will refer to sjSDM when run on the GPU as GPU-sjSDM, and when run on the CPU as CPU-sjSDM.

To assess the predictive performance of the models, we calculated the average area under the curve (AUC) over all species and five independent replicates for each scenario of a hold-out dataset (same size as the dataset used for fitting the model). The AUC measures the capability of the model to distinguish between absence and presence of species. To calculate the accuracy of the estimated species associations and environmental coefficients, we used root mean squared error and the accuracy of the coefficients' signs, again averaged over all species and replicates.

To additionally explore the ability of sjSDM to infer community assembly processes from more realistic ecological data, we simulated communities from the process-based ecological model used by LEIBOLD, RUDOLPH, *et al.* (2022) and compared the inferred species-species association structures with the true structures for sjSDM, BayesComm, Hmsc and gllvm. For details, see Supporting Information S3.

#### 4.2.5 Regularization to infer sparse species-species associations

For the benchmark described above, we simulated data under the assumption that all species interact. While this assumption may or may not be realistic, it is generally desirable for a method to work well also when there is only a small number of associations, that is when the species-species covariance matrix is sparse. We were particularly interested in this question, because we conjectured that the LVM approach imposes correlations on the species-species associations that makes it difficult for LVMs to fit arbitrarily sparse covariance structures.

We therefore simulated data under the same scenarios as before, but with 95% sparsity in the species-species associations. To adjust our model to such a sparse structure, we applied an elastic net shrinkage (ZOU and HASTIE, 2005) to all off-diagonals of the covariance matrix. Following ZOU and HASTIE (2005), the parameters lambda (the regularization strength) and alpha (the weighting between LASSO and ridge) of the elastic net were tuned via fivefolded cross-validation in 40 random steps. As species are correlated within sites, we blocked the CV in sites. For real data, one could additionally consider a spatial blocking (ROBERTS *et al.*, 2017) to account for correlations between sites (e.g., by using the blockCV package, VALAVI *et al.*, 2019).

For the cross-validation, we used 2,000 samples for the MVP approximation in sjSDM, because we found that the approximation error can introduce stochasticity in the tuning process. For BayesComm, Hmsc, and gllvm, we used the default settings (see details and additional comments in Supporting Information S3). For Hmsc, following TIKHONOV, OPEDAL, *et al.* (2020) associations with >95% posterior probability being positive or negative were set to zero.

To measure the accuracy of inferred species-species associations for this benchmark, we normalized the covariance matrices to correlation matrices and calculate the true skill statistic ( $TSS = Sensitivity + Specificity - 1$ , ALLOUCHE, TSOAR, and KADMON, 2006) by transforming the true and predicted associations into two classes: all absolute associations smaller than 0.01 were assigned to class ‘0’ and all absolute associations  $>0.01$  were assigned to class ‘1’. That way, a two-class classification problem was obtained and the TSS was calculated.

#### 4.2.6 Case study – Inference of species-species associations from eDNA

To demonstrate the practical applicability of our approach, we fitted our model to an eDNA community dataset from a published study that sampled 130 sites across Denmark (for details on the study design, see BRUNBJERG *et al.*, 2017; for data and bioinformatics, see FRØSLEV *et al.*, 2019). On each site, eight environmental variables were recorded: precipitation, soil pH, soil organic matter, soil carbon content, soil phosphorous content and mean Ellenberg’s indicator values (light condition, nutrient status and soil moisture) based on the plant community. FRØSLEV *et al.* (2019) identified 10,490 OTUs by eDNA sequencing (81 samples per site). We followed FRØSLEV *et al.* (2019) and removed five sites with  $<4$  OTU presences (low species richness). We used only OTUs occurring at least three times over the remaining 125 sites, which reduced the overall number of OTUs from 10,490 to 3,649 OTU. All eight environmental variables were used in our analysis as main effects on the linear scale. The final dataset consisted of 3,649 OTU co-occurrences over 125 sites with eight environmental variables.

For this analysis, we set the regularization for the z-transformed environmental predictors to  $\lambda = 0.1$  and  $\alpha = 0.5$  (equal weighting of ridge and LASSO regularization). The regularization for the covariances of the species-species associations was tuned over 40 random steps (independent samples from the hyper-parameter space) and with leave-one-out cross-validation. For each of the resulting  $40 \cdot 125 = 5,000$  evaluations, we fitted a GPU-sjSDM in 150 iterations (with a batch size of 12 and 125 site, one iteration consists of 100 optimization steps, see BOTTOU, 2010),  $3,649 \cdot 3,649$  weights for the covariance matrix (see Supporting Information S3 for details about the parametrization of the covariance matrix in sjSDM), with batch size of 8 and learning rate of 0.001 (the size of the update of the parameters in one optimization step).

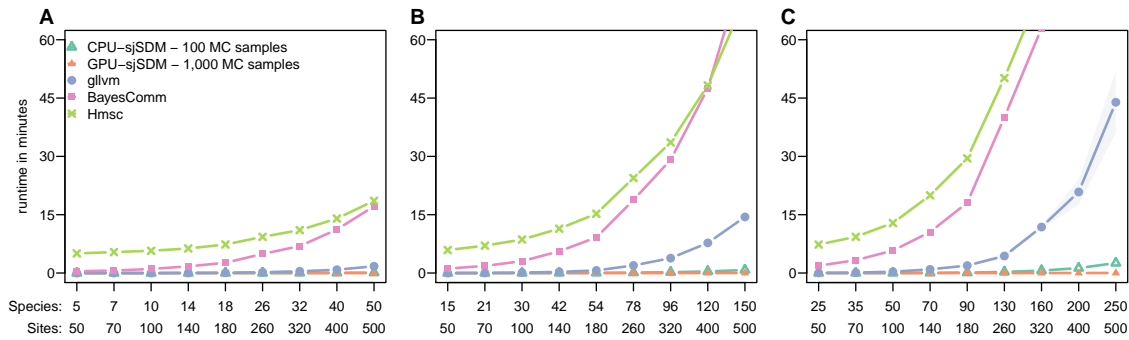
## 4.3 Results

### 4.3.1 Method validation and benchmark against state-of-the-art JSdMs

#### Computational speed

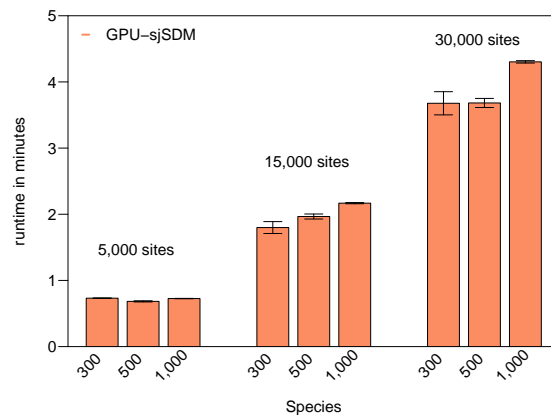
On a GPU, our approach (GPU-sjSDM) required under 3-s runtime for any of our simulated data with 50–500 sites and 5–250 species. When run on CPUs only (CPU-sjSDM), runtimes increased to a maximum of around 2 min (Figure 4.1a; Figure SS3.1). In comparison, Hmsc had a runtime of around 7 min for our smallest scenario and increase in runtime exponentially when the number of species exceeded 40 (Figure 4.1a). BayesComm was slightly faster than Hmsc, but scaled worse than Hmsc to large data sizes (Figure SS3.1). gllvm achieved low runtimes, equivalent and sometimes better than our method for small data ( $<50$  species), but for larger data, runtime started to increase exponentially as well, leading to runtimes of approximately 45 min for our most demanding scenario (Figure 4.1a).

Because of the runtime limitations of the other approaches, we calculated big data benchmarks only for GPU-sjSDM. The overall runtimes for GPU-sjSDM increased from under 1 min for 5,000



**FIGURE 4.1:** Runtime benchmarks for GPU-sjSDM, CPU-sjSDM, gllvm, BayesComm and Hmsc fitted to simulated data with 50–500 sites (dense species-species association matrices) and the number of species set to (a) 0.1, (b) 0.3 and (c) 0.5 times the number of sites. All values are averages from five simulated datasets. To estimate the inference error of the Monte Carlo approximation, GPU-sjSDM was fitted with 1,000 and CPU-sjSDM with 100 MC samples for each species. sjSDM, Scalable joint species distribution model

sites to a maximum of around 4.5 min for 30,000 sites (Figure 2). GPU-sjSDM showed greater runtime increases when increasing numbers of sites, while the numbers of species (300, 500 and 1,000 species in each scenario) had only small effects on runtimes (Figure 4.2).



**FIGURE 4.2:** Benchmark results for sjSDM on big community data. We simulated communities with 5,000, 15,000 and 30,000 sites and for each set of 300, 500 and 1,000 species. sjSDM, scalable joint species Distribution model

For the empirical benchmarking datasets from the study by Wilkinson *et al.* (2019), CPU-sjSDM achieved a 3.8 times lower runtime for the bird dataset and 23 times lower runtime for the butterfly dataset, and GPU-sjSDM achieved a 500 times lower runtime for the bird dataset and a 150 times lower runtime for the butterfly dataset compared to BayesComm, the fastest JSDM in the study by WILKINSON *et al.* (2019) (Table 4.1).

**TABLE 4.1:** Model runtimes in hours. Results for BayesComm against our new approach scalable joint species distribution model (sjSDM) (CPU and GPU version)

Dataset	Wilkinson et al. 2019		Our Approach	
	Size (site * species)	BayesComm	CPU-sjSDM	GPU-sjSDM
Birds (HARRISON, 2015)	2,752 * 370	3.5	0.97	0.007
Butterflies (OVASKAINEN, ABREGO, <i>et al.</i> , 2016)	2,609 * 55	0.15	0.01	0.001
Eucalypts (POLLOCK <i>et al.</i> , 2014)	458 * 12	<0.01	<0.001	<0.001
Frogs (POLLOCK <i>et al.</i> , 2014)	104 * 9	<0.002	<0.001	<0.001
Fungi (OVASKAINEN, HOTTOLA, and SIITONEN, 2010)	800 * 11	<0.02	<0.001	<0.001
Mosquitos GOLDING, 2015	167 * 16	<0.01	<0.001	<0.001

### Accuracy of the inference about species-environment and species-species associations

For simulated data with dense species–species association structures, BayesComm and sjSDM consistently achieved higher accuracy in the inferred species-species associations than the LVMs Hmsc and gllvm (Figure 3a–c). The accuracy of all methods decreased with an increasing proportion of species, to around 70 models (sjSDM and BayesComm) and 60 Even for communities with 300 to 1,000 species, sjSDM achieved accuracies of 69% and higher (Table S3.4).

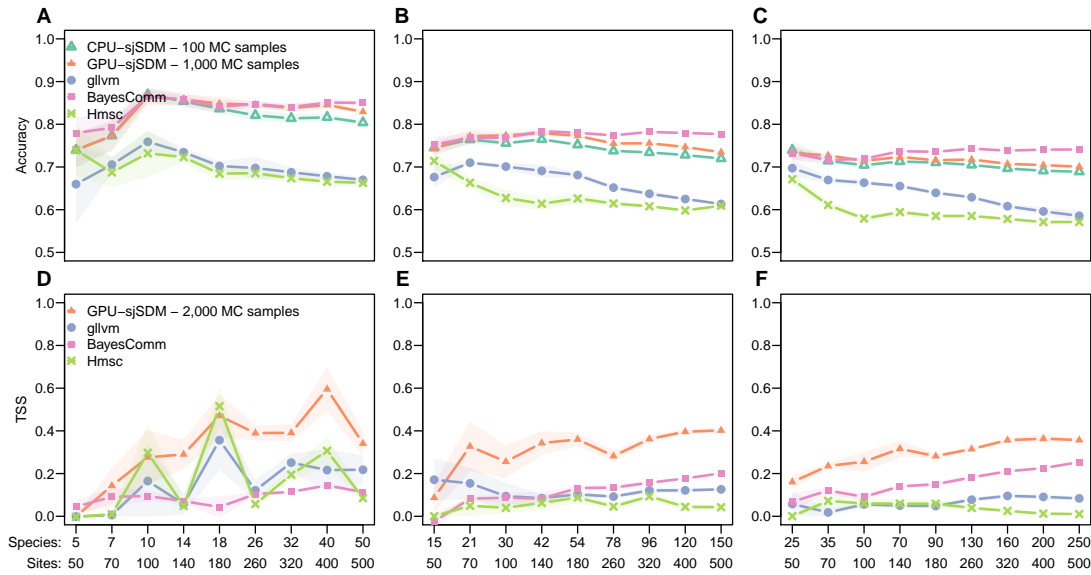
For environmental preferences (measured by RMSE), Hmsc showed slightly higher inferential performance when the number of sites was low (Figure SS3.4a,b) while all models performed approximately equal for a high number of sites (Figure SS3.4a,b).

For simulated data with sparse species-species association structures (95% sparsity), sjSDM achieved the highest TSS (up to 0.35–0.38 with 30% and 50% species, see Figure 4.3d-f). Hmsc showed for 10% species the second highest TSS (Figure 4.3d-f), but for 30% and 50% species together with gllvm the lowest TSS (a maximum of 0.1 TSS for 30% and 50% species). BayesComm showed in average the lowest TSS for 10% species, but for 30% and 50% species the second highest TSS (Figure 4.3d-f). The inferential performance regarding the environmental predictors showed the same pattern as for dense species-species associations. All models improved their environmental accuracy (Figure S3.4c) and reduced RMSE as the number of sites increased (Figure S3.4d).

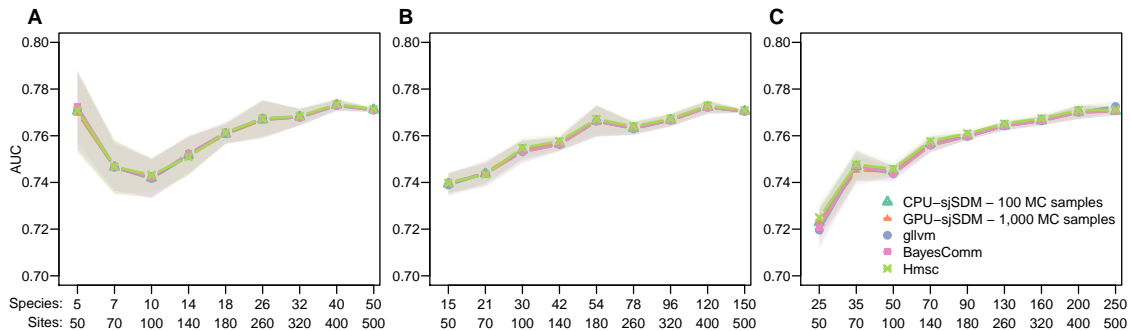
Fitting sjSDM to data simulated with the process-based simulation model used in the study by LEIBOLD, RUDOLPH, *et al.* (2022), we find, similar to LEIBOLD, RUDOLPH, *et al.* (2022), that important signals of the underlying processes, including biotic interactions, can be recovered by sjSDM (Figure S3.10). Our results also hint towards certain advantages of MVP JSDMs over LVMs for this task, although we caution that this question will require further exploration. For details, see Supporting Information S3.

### Predicting species occurrences

All models performed similarly in predicting species occurrences in the simulation scenarios, with predictive accuracies of around 0.75 AUC (Figure 4.4).



**FIGURE 4.3:** Inference performance of the inferred sparse and non-sparse species-species associations. Models were fitted to simulated data with 50 to 500 sites. All values are averages from five simulated datasets. (a-c) The upper row shows the accuracies of matching signs (positive or negative covariance) for the estimated and true dense species-species association matrix. (d-f) The lower row shows the accuracy of inferring non-zero species associations for sparse association matrices (95% sparsity), measured by the true skill statistic (absolute associations smaller than 0.01 were assigned the class ‘0’ and absolute associations  $>0.01$  were assigned the class ‘1’). The number of species for were set to 0.1 (a, d), 0.3 (b, e) and 0.5 (c, f) times the number of sites. To estimate the inference error of the Monte Carlo (MC) approximation, GPU-sjSDM was fitted with 1,000 and CPU-sjSDM with 100 MC samples for each species



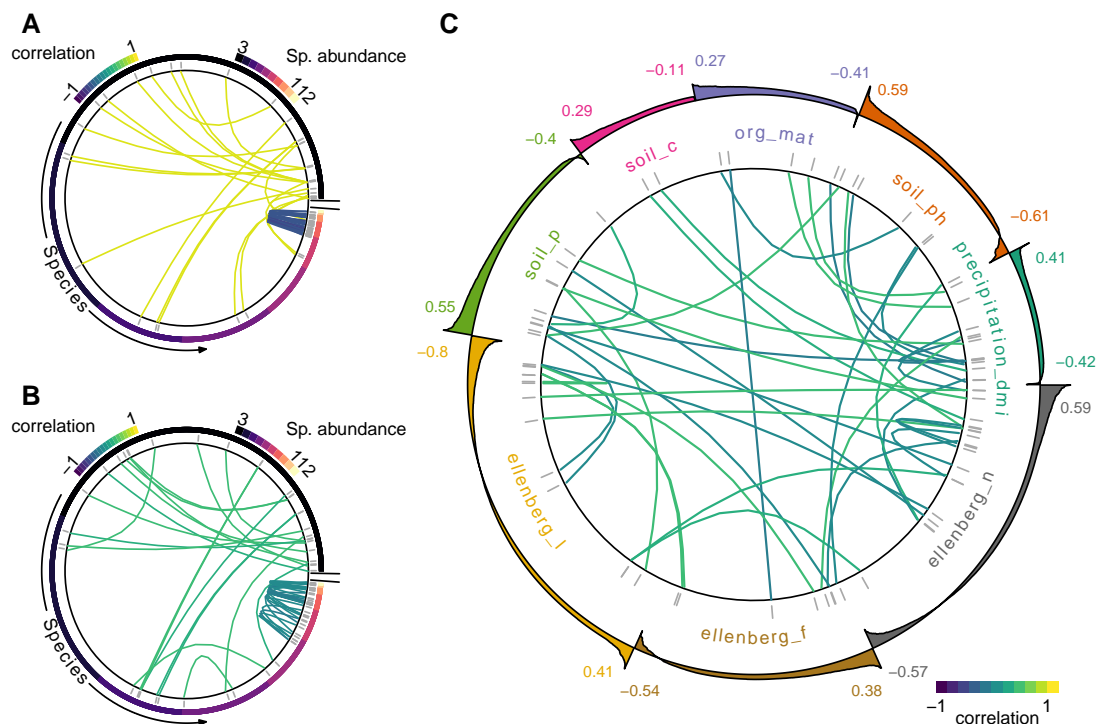
**FIGURE 4.4:** Predictive performance in simulated species distributions for GPU-sjSDM and CPU-sjSDM with gllvm, BayesComm and Hmsc as references. Species distribution scenarios with (a) 50–500 sites and 10%, (b) 30% and (c) 50% species were simulated, on which the models were fitted (training). Models predicted species distributions for additional 50–500 sites (testing). Area under the curve (AUC) was used to evaluate predictive performance on hold-out. sjSDM, scalable joint species distribution model

## Case Study – Inference of species-species associations from eDNA

In our eDNA case study with 3,649 OTUs over 125 sites, we found that without regularization, sjSDM inferred the strongest negative OTU–OTU covariances among the most abundant species and the strongest positive OTU–OTU associations among the rarest OTUs (Figure 4.5a,b). When



optimizing the regularization strength for the OTU–OTU associations via a leave-one-out cross-validation, positive and negative OTU–OTU associations changed somewhat, but the overall pattern stayed qualitatively constant (Figure 4.5a,b). For the environmental covariates (a weak non-optimized regularization was used), we found that most OTUs showed the highest dependency on Ellenberg F (moisture), Ellenberg L (light availability) and Ellenberg N (nitrogen).



**FIGURE 4.5:** Inferred operational taxonomic unit (OTU) associations and environmental preferences for the eDNA community data. The left column (panels a–c) shows OTU–OTU associations for (a) no regularization and (b) tuned regularization, with the 3,649 OTUs sorted according to their summed abundance over 125 sites. The large panel (c) shows the covariance structure of (b), but with OTUs sorted after their most important environmental coefficients (largest absolute environmental effect size; the outer ring shows the environmental effect distribution for the OTUs within the group)

## 4.4 Discussion

Joint species distribution models extend standard species distribution models by also accounting for species–species associations. Current JSDM software, however, exhibits computational limitations for large community matrices, which limits their use for big community data that are created by novel methods such as eDNA studies and metabarcoding. Here, we presented sjSDM, a new numerical approach for fitting JSDMs that uses Monte Carlo integration of the model likelihood, which allows moving calculations to GPUs. We show that this approach is orders of magnitude faster than existing methods (even when run on the CPU) and predicts as well as any of the other JSDM packages that we used as a benchmark. To avoid overfitting, especially when fitting sjSDM to hitherto computationally unrealistic eDNA datasets with thousands of species, we introduced a flexible elastic net regularization on species associations and environmental preferences. sjSDM inferred the signs of full association matrices and identified zero/non-zero entries in sparse species–species associations across a wide range of scenarios better than all tested alternatives. Advantages of BayesComm and sjSDM over LVM-based JSDMs (Hmsc and gllvm) occurred for all species–species associations structures tested, while improvement of sjSDM over BayesComm

was in particular visible for sparse species-species associations.

## Computational performance

Whereas runtimes for Hmsc, BayesComm and gllvm started to increase exponentially when the number of species exceeded around 100, sjSDM scaled close to linearly with the number of species regardless of whether we used GPU or CPU computations (Figure 1a). A further advantage of sjSDM is that, unlike in particular the MCMC algorithms used in BayesComm and Hmsc, it is highly parallelizable, which allows using efficiently the advantages of modern computer hardware such as GPUs. These two properties, scalability and parallelizability, make sjSDM the first and currently only JSDM software package that seems capable of analysing big eDNA datasets HUMPHREYS *et al.*, 2019; TIKHONOV, OPEDAL, *et al.*, 2020; WILKINSON *et al.*, 2019 on standard computers with acceptable runtimes.

We concede that runtimes of the different JSDM implementations may depend on hyperparameters such as the number of MCMC iterations in BayesComm or Hmsc, or the number of MC samples in sjSDM. Changing these parameters could affect results; however, increasing or decreasing MCMC iterations would only linearly shift the runtime curves (Figure 4.1; Figure S3.1). When we compare such a linear shift with the strong nonlinearity scaling of BayesComm and Hmsc, it seems unlikely that changes to the hyperparameters could qualitatively change the results. Moreover, sjSDM uses a Monte Carlo approximation of the likelihood and runtime, thus naturally depends on number of Monte Carlo samples. Yet, all other tested methods use approximations as well to obtain the inference. Neither our inferential results nor other indicators give us reasons to think that the approximation made by sjSDM is worse than that of competing algorithms. Specifically, increasing the number of Monte Carlo samples for each species in sjSDM from 100 to 1,000 increased the inferential performance moderately (Figure 4.3a-c). Also, the excellent inferential accuracy of sjSDM across various tests does not suggest a large approximation error. We are therefore confident that our Monte Carlo approximation is acceptable in general, and not worse than the approximations made in other packages.

State-of-the-art JSDM implementations offer a variety of extensions such as the inclusion of phylogeny, space and traits (e.g. Hmsc, TIKHONOV, OPEDAL, *et al.*, 2020). Here, we used sjSDM only for estimating a simple MVP structure, which is arguably the most generic version of a JSDM that is implemented by all packages. In principle, however, the algorithm used in sjSDM could be extended to include other structures that have been proposed in the literature. The sjSDM package already supports alternative responses and link functions (e.g., normal, Poisson or binomial), and has an option to add spatial model components (e.g., via spatial eigenvectors). Also the option to include traits by using the fourth-corner approach as in gllvm BROWN, WARTON, *et al.*, 2014; NIKU, HUI, *et al.*, 2019 could be added. A crucial question for all these extensions is if they interact beneficially with our MLE approximation, that is if we can optimize the MLE without having to resort to other integration methods (such as MCMC or Laplace approximations) for the added structures, which would negate the speed advantage of sjSDM. For example, we found that the approximation used by sjSDM does not interact well with the addition of conditional autoregressive (CAR) terms in the model structure.

## Inferential performance

All JSDM implementations showed similar performance in correctly inferring environmental responses, but the MVP approaches, sjSDM and BayesComm, achieved significantly higher accuracy in inferring the correct signs of species-species associations (Figure 4.3a-c) and identifying sparse

structures (Figure 4.3d-f). It should be noted here that we tuned the regularization of sjSDM to improve the performance for sparse associations and the other JSDM might also benefit from tuning the regularization. BayesComm and Hmsc allow more restrictive priors to be specified on the covariance matrix (BayesComm) or on the factor loadings (Hmsc). However, the long runtimes of these JSDM implementations place time constraints on testing different prior specification. Moreover, BayesComm already achieved high TSS for sparse associations with default specifications, indicating superiority of highly parametrized JSDM over LVM for sparse structures (Figure 4.3d-f).

We speculate that the LVMs' lower performance for the inferred species associations originates from the constraints imposed by the LVM structure, which creates some bias that showed in particular for dense species association structures (compare Figure 4.2a; Figure S3.2). This is not particularly surprising, as similar phenomena have been found also for other approaches to covariance regularizations, for example in spatial models STEIN, 2014. It is difficult to estimate how important these biases are in practical applications, because we still know too little about the typical structure of species associations in real ecological data OVASKAINEN, TIKHONOV, *et al.*, 2017. One might expect that associations in data generated by high-throughput technologies, which detect species already at very low densities, would be relatively sparse, or consist of a mix between sparse and non-sparse blocks for rare and common species (cf. CALATAYUD *et al.*, 2019). Moreover, one would expect that LVMs would be particularly efficient if species associations follow the structure implemented in the LVMs. To test this, we also simulated data from an LVM structure, and fitted these data with sjSDM and the two LVMs (gllvm and Hmsc). Our results show that the LVMs indeed perform better than for such data than for our previously used general covariance matrices, but not better than sjSDM (Figures S3.7, S3.8, and S3.9).

A slight disadvantage of sjSDM is that it is more complicated to obtain parameter uncertainties, compared to JSDM implementations based on MCMC sampling such as BayesComm and Hmsc. The R implementation of sjSDM calculates Wald confidence intervals for all environmental predictors using PyTorch's automatic differentiation feature. However, we have currently no analytical option to calculate confidence intervals for the species-species associations. If these are needed, we propose using bootstrap samples.

## Implications and outlook for ecological data analysis

The JSDM structure has the potential to become the new default statistical approach for species and community observations that originate from eDNA and similar big community data. However, to fulfil this promise, we need statistical algorithms that scale to big datasets and deliver accurate inference, in particular for a large number of species or operational taxonomic units. Our results show that a combination of a scalable and parallelizable Monte Carlo approximation of the likelihood, together with a shrinkage regularization of the species-species covariance, can achieve both goals.

Our results also suggest that regularization of the species-species covariance is particularly crucial to obtain reasonable inference for such data. In principle, all software packages that we compared could include additional regularization methods, such as the elastic net employed in our approach. Better understanding the use of such statistical approaches is one promising route for further research. Another option would be to impose ecologically motivated structures on the species-species covariance matrix (e.g. BYSTROVA *et al.*, 2021; CLARK, NEMERGUT, *et al.*, 2017; TAYLOR-RODRÍGUEZ *et al.*, 2017).

Another interesting question is how ecologists should use and interpret JSDMs, once they scale to big data. Many recent studies have stressed that JSDMs may improve predictions NORBERG *et al.*, 2019, and indeed, from ecological theory, one would expect that species associations are important for accurate species occurrence predictions DORMANN, SCHYMANSKI, *et al.*, 2012; NORBERG *et al.*, 2019; WISZ *et al.*, 2013. Despite different accuracy in inferring true species associations (Figure 4.3), we found similar predictive performances (Figure 4.4) for all tested JSDMs. It should be noted, however, that the AUC metric we used captures only marginal predictive performance, and a closer relationship between inferential and predictive performance might have arisen when using joint predictive performance measures (WILKINSON *et al.*, 2021).

Another open question in the context of predictions is the relative importance of including the association structure, compared to a more detailed description of the environmental model components. Without systematic benchmarks, where model structures on both biotic and abiotic predictions are flexibly adopted (e.g., via machine learning approaches such as in CHEN, XUE, and GOMES, 2018), and where indicators of joint predictive performance WILKINSON *et al.*, 2021 are used that are sensitive to covariances, it is difficult to examine whether increases in predictive performance of JSDMs are really due to their exploitation of a stable association structure, or simply arise from the higher model complexity of JSDMs, which allows fitting the data more flexibly.

When turning to inference, the new information that JSDMs deliver to ecologists are species-species covariance estimates (LEIBOLD, RUDOLPH, *et al.*, 2022). These could be used, for example, to test if the strength or structure of species associations varies with space or environmental predictors; or if spatial species associations correlate with local trophic or competitive interactions or traits (see generally POISOT, STOUFFER, and GRAVEL, 2015). For regional studies, there is the prospect of extending the traditional variation partitioning (environment and space; COTTENIE, 2005) to include biotic associations by using JSDMs (LEIBOLD, RUDOLPH, *et al.*, 2022). Our results regarding the moderate, but significantly better than random accuracy of inferred covariance structures, even on datasets with hundreds of species, are encouraging for such a research program.

Recently, however, concerns about the usefulness of JSDM for examining species interactions have emerged. For instance, it has been criticized that the species-species associations inferred by JSDM cannot always be linked to ecological interactions because of their symmetric nature (BLANCHET, CAZELLES, and GRAVEL, 2020; POGGIATO *et al.*, 2021; ZURELL, POLLOCK, and THULLER, 2018), that the associations may absorb missing environmental covariates (POGGIATO *et al.*, 2021) or that JSDM associations can be scale dependent (see KÖNIG *et al.*, 2021 although this also applies to ecological interactions, see POISOT, STOUFFER, and GRAVEL, 2015). We acknowledge these observations but do not share all concerns. JSDM estimate associations between species after accounting for the environment. Such associations are not necessarily causal or mechanistic, and they are naturally also influenced by unmeasured predictors, scale and other factors, but they can also be caused by real species interactions, as shown in the study LEIBOLD, RUDOLPH, *et al.* (2022) by and confirmed by us for sjSDM (Figure S3.10). Thus, when interpreted with due care, JSDMs provide useful ecological information beyond pure niche models. If more high-resolution dynamic data were available, we could use more precise (causal) methods to infer the direction of interactions (BARRAQUAND *et al.*, 2021; MOMAL, ROBIN, and AMBROISE, 2020), which likely match much closer to actual species interactions. Yet, for the static community data that make up the bulk of the data available to ecologists today, these methods are not applicable, but JSDMs are and can provide additional information compared to existing alternatives.

## 4.5 Conclusions

We presented sjSDM, a new method to fit JSDBMs, and benchmarked it against state-of-the-art JSDBM software. sjSDM is orders of magnitudes faster than current alternatives, and it can be flexibly regularized, which leads to overall superior performance in inferring the correct species association structure. We emphasize that the superior scaling holds also when using CPU computations, and that the possibility to move calculations on a GPU is only a further advantage of the algorithm. We provide our tool in an R package (<https://github.com/TheoreticalEcology/s-jSDM>, available for Linux, MacOS and Windows), with a simple and intuitive interface and the ability to switch easily between linear and nonlinear modelling, as well as between CPU and GPU computing. The R package also includes extensions for considering abundance data as well as spatial coordinates, and to partition the importance of space, environment and species associations for predicting the observed community composition.

**Acknowledgements** The authors thank Douglas Yu and Yuanheng Li, as well as Gavin Simpson and four anonymous reviewers for their valuable comments and suggestions.

**Data availability statements** The processed datasets for runtime benchmarking (case study 1) are available as Supporting Information for WILKINSON *et al.* (2019). The eDNA dataset is available at [https://github.com/tobiasgf/man\\_vs\\_machine](https://github.com/tobiasgf/man_vs_machine). The analysis and the version of the R package sjSDM used in this analysis are available in an online repository (PICHLER and HARTIG, 2021b). The latest version of the sjSDM R package can be found at [github.com/TheoreticalEcology/s-jSDM](https://github.com/TheoreticalEcology/s-jSDM).



---

## CAN PREDICTIVE MODELS BE USED FOR CAUSAL INFERENCE?

---

**Maximilian Pichler and Florian Hartig**

in prep., preprint available at [2306.10551](https://arxiv.org/abs/2306.10551)

### **Abstract**

Supervised machine learning (ML) and deep learning (DL) algorithms excel at predictive tasks, but it is commonly assumed that they often do so by exploiting non-causal correlations, potentially limiting interpretability and generalizability. Here, we show that this tradeoff between explanation and prediction is not as deep and fundamental as expected. Whereas ML and DL algorithms will indeed tend to use non-causal features for prediction when fed indiscriminately with all data, it is possible to constrain the learning process of any ML and DL algorithm by selecting features according to Pearl's backdoor adjustment criterion. In such a situation, some algorithms, in particular deep neural networks, can provide near unbiased effect estimates under feature collinearity. Remaining biases are explained by specific algorithmic structures as well as hyperparameter choice. Consequently, optimal hyperparameter settings are different when tuned for prediction or inference, confirming the general expectation of a tradeoff between them. However, the effect of this tradeoff is small compared to the effect of a causally constrained feature selection. Thus, once the causal relationship between the features is accounted for, the difference between prediction and explanation may be much smaller than commonly assumed. We also show that such causally constrained models generalize better to new data with altered collinearity structures, suggesting generalization failure may often be due to a lack of causal learning. Our results not only provide a perspective for using ML for inference of (causal) effects but also help to improve the generalizability of fitted ML and DL models to new data.

**Keywords:** Causal Inference, Artificial Intelligence, Deep Learning, Machine Learning

## 5.1 Introduction

In the fields of statistics and machine learning, it is widely recognized that there is a difference between predictive and explanatory or causal modelling (SHMUELI, 2010). One of the reasons is that using correlations between features and the response can improve predictions, even when those variables are not causally connected. Along with the bias-variance tradeoff (e.g. BELKIN *et al.*, 2019; PICHLER and HARTIG, 2023b), the ability to exploit non-causal correlations likely explains the success of supervised machine learning (ML) and deep learning (DL) algorithms in predictive tasks (BURY *et al.*, 2021; JUMPER *et al.*, 2021; QI and MAJDA, 2020); however, such a predictive modelling strategy tacitly accepts that trained ML models will in general not learn the true underlying relationships, which limits their interpretability (suggesting a prediction-explanation tradeoff) and may also partly explain why they often do not generalize well to new data (suggesting an interpolation-extrapolation tradeoff).

Specialized ML approaches for estimating causal effects exist (e.g., causal forest (WAGER and ATHEY, 2018), double/debiased ML (CHERNOZHUKOV *et al.*, 2018), metalearners (KÜNZEL *et al.*, 2019) or causal discovery algorithms (BENGIO *et al.*, 2019; KE *et al.*, 2019)), and we will discuss the relationship between these and the present study later. Here, our goal is to understand if and when classical ML algorithms can correctly adjust for collinear features, which is a prerequisite for using them for causal inference. By means of that, we can also explore if there is indeed a fundamental tradeoff between prediction and explanation when training ML and DL algorithms (SHMUELI, 2010).

The key idea of our study is that if the causal graph is known, research in causal inference has solved the problem of how we should select features such that a statistical regression model (e.g., ordinary least squared (OLS)) would adjust for confounding such that the causal effect of one or several target variables is correctly estimated (PEARL, 2009). As pointed out by ZHAO and HASTIE (2021), these ideas should in principle be transferable to ML and DL models. However, given that ML and DL models rely heavily on (adaptive) regularization and induced regularization biases can affect causal estimates (ZOU and HASTIE, 2005), it remains an open question how well this idea works in practice. Here, we address this problem by first suggesting an explainable AI (xAI) metric to extract effect estimates from fitted ML and DL models, and then performing a number of simulations to examine bias on effect estimates under collinearity in different ML and DL models.

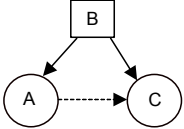
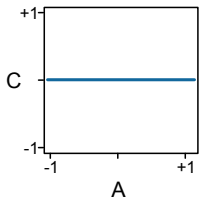
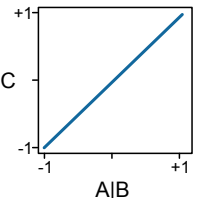
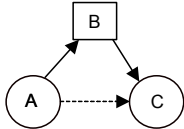
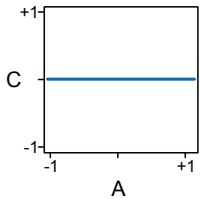
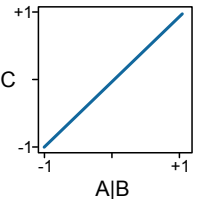
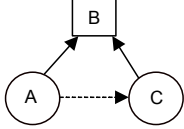
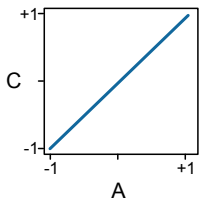
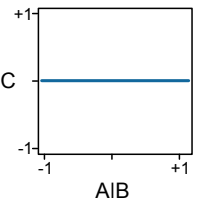
### 5.1.1 Causally constrained ML requires unbiased learning

To understand why bias in effect estimates is crucial for causal inference, we shortly summarize the general approach to separate correlation from causality in static data. The key problem is that a correlation may be caused by a direct causal link, but also by a third variable that causally influences both the feature of interest and the response (i.e., a confounder, see Table 5.1). To adjust for the effect of such additional variables, one must first generate a hypothesis about the underlying graph which describes causal relationships between all features (GREENLAND, 2003; PEARL, 2009). Based on this graph, one can isolate the underlying causal effect of the target feature by conditioning on the other features (adjustment), for example using multiple regression or (piecewise) structural equation models (Table 5.1, BOLLEN and NOBLE, 2011 see also PEARL, 2009). As pointed out by ZHAO and HASTIE (2021), we should be able to transfer the same idea to ML and DL models. We refer to ML models trained with such a set of causally selected features as ‘causally constrained’.

This argument assumes, however that ML algorithms (similar to ordinary least squared (OLS))



**TABLE 5.1:** The column DAG (short for: directed acyclical graph) describes the assumed causal relationship between the variables. The columns “description” describes the correlations created by the respective relationships and the usual statistical adjustment. The estimated effects (raw:  $P(C|A)$  and adjusted:  $P(C|A, B)$  in a multiple regression) are visualized in the last column to the right.

DAG	Description	$P(C A)$	$P(C A, B)$
<p>Confounder</p> 	<ul style="list-style-type: none"> <li>If we look at the unconditional correlation <math>P(C A)</math>, we see a spurious effect.</li> <li>By conditioning on the confounder <math>P(C A, B)</math>, we can isolate the true causal effect</li> </ul>		
<p>Mediator</p> 	<ul style="list-style-type: none"> <li>If we look at the unconditional correlation <math>P(C A)</math>, we see the total effect</li> <li>By conditioning on the mediator <math>P(C A, B)</math>, we can isolate the direct causal effect</li> </ul>		
<p>Collider</p> 	<ul style="list-style-type: none"> <li>If we look at the unconditional correlation <math>P(C A)</math>, we see the true causal effect.</li> <li>By conditioning on a collider <math>p(C A, B)</math>, we create a collider bias and obtain the wrong causal effect</li> </ul>		

regression) provide unbiased effect estimates under collinearity so that the adjustment sketched in Table 5.1 can remove the entire effect of possible confounders, and there are several reasons to cast doubt on that assumption.

Most importantly, it is well-known that certain ML techniques trade off bias against variance, which can disproportionately bias collinear feature effects. For example, shrinkage estimators such as LASSO, RIDGE or elastic-net, although originally motivated by the desire to improve OLS estimates under collinearity (HOERL and KENNARD, 1970), tend to push strong effects of a feature over to other collinear features where the shrinkage loss is weaker (Figure 5.1) (ZOU and HASTIE, 2005). This creates a stronger regularization bias for collinear features than for independent features or the predictions. We call this phenomenon that a causal effect moves over to collinear non-causal feature a “causal spillover”.

Similar issues may arise in ensemble models. In the popular random forest (RF) algorithm, for example, variance in the tree ensemble is increased by randomly hiding features at each split of each tree (BREIMAN, 2001a). This variance decreases the correlation between ensemble members, which can reduce the predictive error of the ensemble (BURNHAM and ANDERSON, 2004; DIETTERICH, 2000; DORMANN, CALABRESE, *et al.*, 2018). However, if a confounder is hidden by this process, its casual effect will spill over to other collinear features, inducing a bias in the effect estimates (GREGORUTTI, MICHEL, and SAINT-PIERRE, 2017, confirmed by Figure 5.1).

A different effect can occur in greedy learning algorithms such as (gradient) boosted regression trees (BRT). In these algorithms, weaker collinear features are only used when the stronger ones are exhausted, which can occur within the internal regression trees or could potentially arise from the boosting (Figure 5.1, Figure S4.1). Based on this, there is a concern that strong features

steal effects from weaker collinear features (we refer to this as ‘causal greediness’).

For neural networks (NN), it is unclear if such biases are expected. Pure (deep) NNs do not explicitly include any of the previously mentioned regularization mechanisms. Nevertheless, it is often reported that trained NNs display a simplicity bias akin to an implicit regularization, which has been associated to the stochastic gradient descent or network architecture (HUH *et al.*, 2021; SHAH *et al.*, 2020). Such a simplicity bias could lead to similar causal spillover as those reported for the elastic net. More importantly, however, NN are in practice usually trained with additional regularization, for example in the form of shrinkage (e.g., elastic net) or dropout. The latter implicitly creates an ensemble model (SRIVASTAVA *et al.*, 2014) and could lead to similar causal spillover as in random forest (Figure S4.3).

### 5.1.2 Measuring causal bias of trained models via xAI

A complication for quantifying and comparing to what extent these theoretical considerations apply for trained ML and DL models is that those models do not directly report effect sizes. However, it is possible to extract feature effects using appropriate model-agnostic explainable AI (xAI) method (MURDOCH *et al.*, 2019). Many number of xAI methods exist, but most quantify feature importance for predictions, which is more analogous to variance partitioning in an ANOVA setting (i.e., a joint measure of effect and variance of a feature) and not of effect sizes (MOLNAR, 2020). Moreover, it is known that many xAI metrics are not robust against feature collinearity, for example because univariate unconditional permutations will generate feature combinations that are outside the range of collinear data (HOOKER and MENTCH, 2019; HOOKER, MENTCH, and ZHOU, 2021; JANZING, MINORICS, and BLÖBAUM, 2020; MOLNAR *et al.*, 2020), which makes them unreliable for our purpose.

In search for a model-agnostic post-hoc xAI metric that corresponds, in the case of linear effects of possible collinear features, exactly to effect sizes estimated by linear regression, we settled on the idea of average conditional effects (ACEs). ACEs, in the statistical literature also known as average marginal effects, are a common choice to extract average effects for nonlinear statistical models (BRAMBOR, CLARK, and GOLDER, 2006). The basic idea behind the ACE is to use the fitted model and calculate the average local derivative of the prediction with respect to a target feature over all observations (see methods). With  $n$  observations, the *ACE* for vector  $\mathbf{x}_k$  ( $k$  indexing the features) is then:

$$ACE_k = \frac{1}{N} \sum_{i=1}^N \frac{\partial \hat{f}(\mathbf{X})}{\partial x_k^{(i)}} \quad (5.1)$$

The idea to use ACE to interpret ML model is not new (SCHOLBECK *et al.*, 2022), but whereas SCHOLBECK *et al.* (2022) suggested to extend ACE for non-linear effects by splitting the feature space in different regions, we argue that a robust average across feature-output relationship corresponds exactly to what one would visually characterize as a learned causal effect and what we need here to compare our ML algorithms with the OLS. The ACE can also easily be extended to infer two-way or higher interactions (see Supporting Information S4).

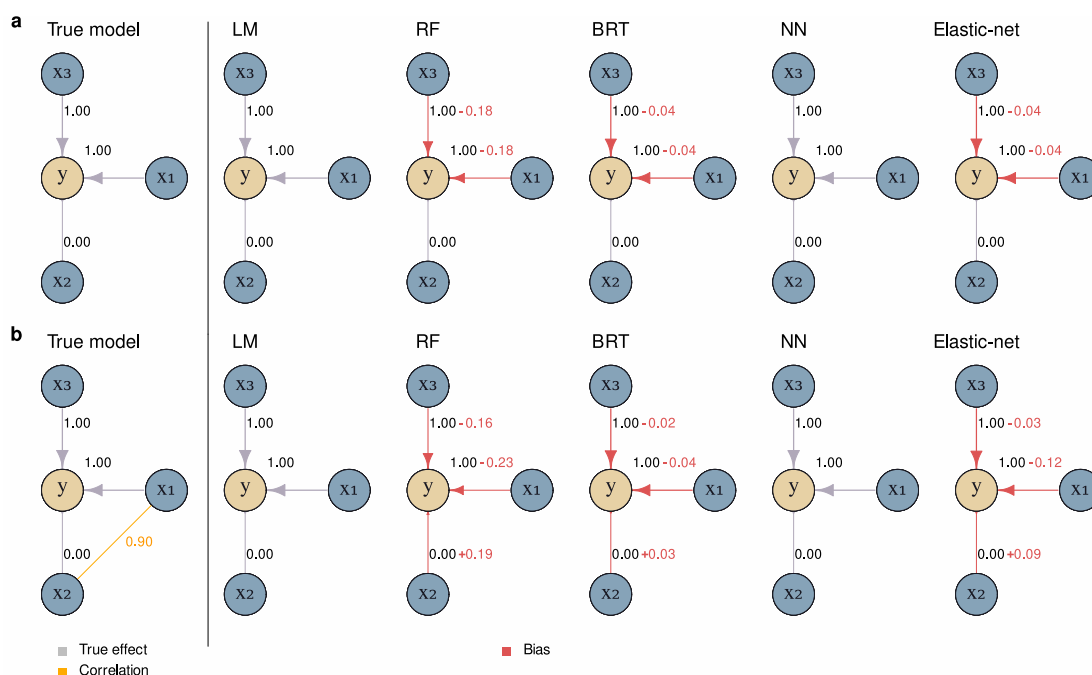
We acknowledge that there are alternatives to ACEs, in particular global xAI metrics based on Shapley values (SUNDARARAJAN and NAJMI, 2019), their algorithmically specific versions such as kernSHAP or treeSHAP values (LUNDBERG, ERION, and LEE, 2018; LUNDBERG and LEE, 2017), or accumulated local effect plots (APLEY and ZHU, 2020). But those do not map directly on regression slopes and are computationally expensive, whereas our results show that ACEs are fast to compute and correspond well to OLS effect sizes in a linear simulation setting.

## 5.2 Results

Equipped with an xAI method for extracting main effects and interactions from fitted ML models, we proceed to examine if ML models learn unbiased effects under feature collinearity. We considered the four major classes of ML algorithms currently in use, which are also representative of different ML paradigms: Random forest (RF, using the bagging paradigm) and boosted regression trees (BRT, using the boosting paradigm) are both ensemble models that rely on the principles of model averaging (DIETTERICH, 2000; DORMANN, CALABRESE, *et al.*, 2018), deep neural networks (NN, 3 hidden layers with 50 hidden nodes), and elastic-net regression models with a LASSO and Ridge regularization (paradigm of shrinkage estimators) (ZOU and HASTIE, 2005).

### 5.2.1 Near-asymptotic performance

In a large-data situation (see methods), our results confirm the theoretical expectations that RF as well as to a lesser degree BRT and elastic net are principally biased towards smaller effect sizes (regularization), even if there is no collinearity (Figure 5.1, row a), whereas pure NN is near unbiased. We also included an OLS regression as a reference, which is mathematically known to be unbiased.



**FIGURE 5.1:** Quantification of causal biases and spillover for different ML algorithms when trained on data simulated from two different causal relationships (a: uncorrelated features with effect sizes ( $\beta_1=1.0$ ,  $\beta_2=0.0$ , and  $\beta_3=1.0$ ), b:  $x_1$  and  $x_2$  being strongly correlated (Pearson correlation factor = 0.9) but only  $x_1$  affects  $y$ ). Sample sizes were sufficiently large that stochastic effects can be excluded (1000 observations and 500 repetitions) and we expected that this would not be a challenge for statistical models such as the linear model (LM, OLS). Effects of the ML models were quantified using average conditional effects.

Under collinearity (Figure 5.1, rows b), additional algorithm-specific biases arise: the strongest causal spillover is observed for the RF algorithm, presumably because feature subsampling leads to open backdoors (already shown by GREGORUTTI, MICHEL, and SAINT-PIERRE, 2017), followed by

elastic-net and BRT, whereas the NN remained unbiased. The fact that BRT showed light spillover and no causal greediness was against our expectation. We explored further and found that pure linear boosting only initially leads to causal greediness; however, in the course of further boosting steps, this is compensated, resulting in an unbiased effect estimate (Figure S4.4). The spillover that we observe is likely caused by other features of the boosting algorithm, in particular the use of regression trees (Figure S4.3).

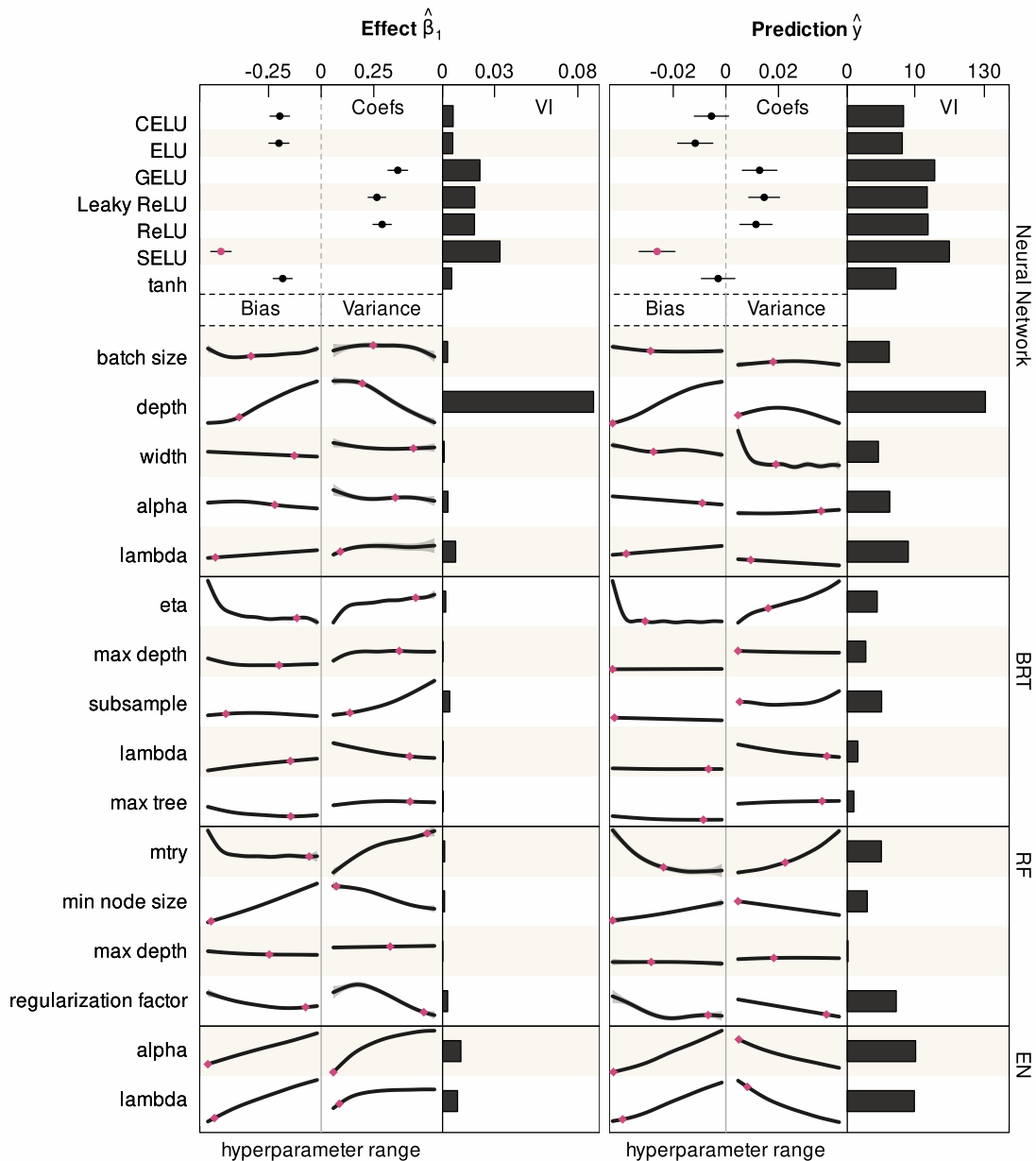
We note that the NN was trained without any regularization, which may be considered unrealistic in a practical scenario. An NN trained with regularization via dropout (WAGER, WANG, and LIANG, 2013) (rate = 0.3) showed similar biases as RF and elastic-net (Figure S4.5). We explain this by the fact that dropout, similar to the feature subsampling in RF, hides some effects during training which can lead to causal spillover.

### 5.2.2 Performance in data-poor situations

While our previous results using a large sample size allow us to understand the mechanisms by which ML algorithms introduce bias into effect estimates, they may also seem somewhat discouraging because, except for the NN, all ML models perform considerably worse than a simple linear regression (Figure 5.1). This, however, was to be expected, because OLS is known to be the best linear unbiased estimator (BLUE) for estimating feature effects.

Advantages for ML models over OLS are expected when either the functional form of the response is nonlinear or unknown, or when there is an advantage to be gained from trading off bias against variance, which is the case in data-poor situations when the variance contributes significantly to the total error ( $MSE = Bias^2 + Var + \sigma^2$  with  $\sigma^2 = irreproducible\ error$ ). In such a situation, ML algorithms might outperform OLS in estimating complex or nonlinear effects, in particular if model hyperparameters that adjust the regularization strength are tuned. To examine such a scenario, we simulated a data-poor regression situation with 100 features and 50, 100, and 600 observations (see methods). The impact of hyperparameters and a separate bias-variance tradeoff for inference and predictions.

In such a data-poor situation, we expect that hyperparameters need to be tuned, and we expect that there is a tradeoff between tuning for either explaining or predicting. To test this, we sampled 1000 different hyperparameters for each model (Table S4.1), calculated the bias and variance for effects and predictions (20 replicates), and modeled the effects of hyperparameters on predictive and inferential MSE using generalized additive models (GAM) and random forest (RF).



**FIGURE 5.2:** Results of hyperparameter tuning for Neural Networks (NN), Boosted Regression Trees (BRT), Random Forests (RF), and Elastic Net (EN) for 100 observations with 100 features. The influence of the hyperparameters on effect  $\hat{\beta}_1$  (bias, variance, and MSE), and the predictions of the model,  $\hat{y}$ , (bias, variance, and MSE) were estimated by a multivariate generalized additive model (GAM). Categorical hyperparameters (activation function in NN) were estimated as fixed effects. The responses (bias, variance, MSE) were centered so that the categorical hyperparameters correspond to the intercepts. The variable importance of the hyperparameters was estimated by a random forest with the MSE of the effect  $\hat{\beta}_1$  (first plot) or the prediction (second plot) as the response. Red dots correspond to the best predicted set of hyperparameters (based on a random forest), in the first plot for the minimum MSE of the effect for  $\hat{\beta}_1$  and in the second plot for the minimum MSE of the predictions ( $\hat{y}$ ).

We find that hyperparameters have significant effects on both bias and variance of effect estimates and predictions. For NNs, the SELU activation function caused the smallest bias on the effect estimate and the prediction (Figure 5.2), but this effect decreased with increasing observations

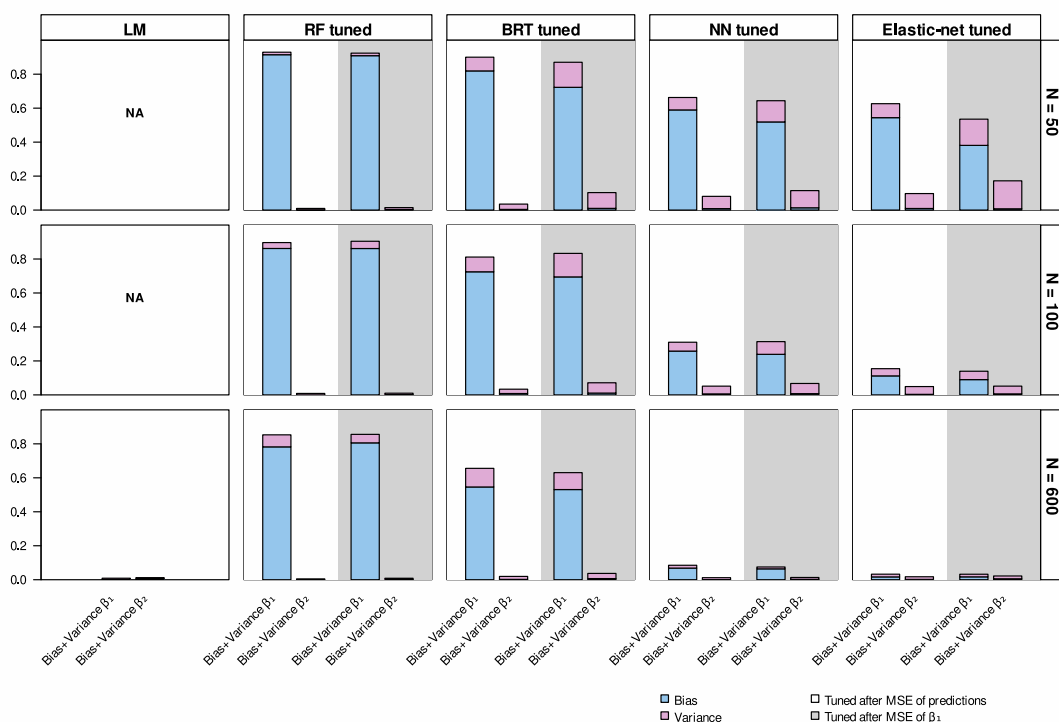
(Figure S4.7-S4.9). More hidden layers (depth) increased the bias on the effect estimates and the prediction (Figure 5.2). For BRT, larger learning rates ( $\eta$ ) and larger number of trees decreased bias on the effect estimates and predictors (Figure 5.2). For RF, more features that are used in each split ( $m_{\text{try}}$ ) and larger minimum node sizes decreased the biases (Figure 5.2). For elastic net, as expected,  $\alpha$  and  $\lambda$  had strong effects on the effect and prediction errors (Figure 5.2).

Often, the effects of the hyperparameters on bias and variance were contrary (depth in NN,  $\eta$  in BRT, and  $m_{\text{try}}$  in RF), which reflects the well-known bias-variance tradeoff and explains why the optimal set of hyperparameters (red, predicted by an RF, Figure 5.2) is not at the marginal optima of the hyperparameter-error associations (Figure 5.2).

Most importantly, although the bias-variance tradeoffs for inference and prediction often showed similar tendencies for hyperparameters, some hyperparameters had notably different effects for the two goals (Figure 5.2), for example the number of features to select from in RF ( $m_{\text{try}}$  in RF). That and the fact that the variance was on different scales for effect estimate and prediction error (not shown in Figure 5.2) led to different optimal hyperparameter sets, meaning that even if the models are causally constrained via the feature selection, there is a tradeoff between tuning their hyperparameters for predictions or for inference.

### 5.2.3 Bias and error on effects induced by algorithm and hyperparameter choice in data-poor simulations

Based on the optimal hyperparameters for prediction and inference, we then quantified bias and variance of present or absent feature effects under feature collinearity (see methods) for all algorithms and again OLS as a reference. Our results show that, as expected, all ML algorithms apply regularization, resulting in increasing bias with smaller data sizes (Figure 5.2). Relatively, however, NN and elastic-net showed the smallest biases, which decreased stronger with more observations while RF showed the largest biases. For the second effect estimate, the zero effect ( $\beta_2$ ), all models showed small biases (Figure 5.3). For 600 observations, the LM was unbiased, as expected (Figure 5.3). Variance was small for all effect estimates (Figure 5.3). Given that elastic net is not a general function approximator but rather ‘just’ a regularized OLS model, we conclude that of the general algorithms, NN seems the preferable choice for the purpose of causal inference.



**FIGURE 5.3:** Bias and variance of estimated effects in data-poor situations.  $N = 50, 100$ , and  $600$  observations of  $100$  weakly correlated features were simulated. True effects in the data generating model were  $\beta_1=1.0, \beta_2$ , and the other  $98$  effects were equally spaced between  $0$  and  $1$ . Models were fitted to the simulated data ( $1000$  replicates) with the optimal hyperparameters (except for LM, which doesn't have hyperparameters). Hyperparameters were selected based on the minimum MSE of  $\beta_1$  (green) or the prediction error (based on  $\hat{y}$ ) (red). Bias and variance were calculated for  $\beta_1$  and  $\beta_2$ . Effects ( $\beta_i$  for  $i = 1, \dots, 100$ ) were approximated using ACE.

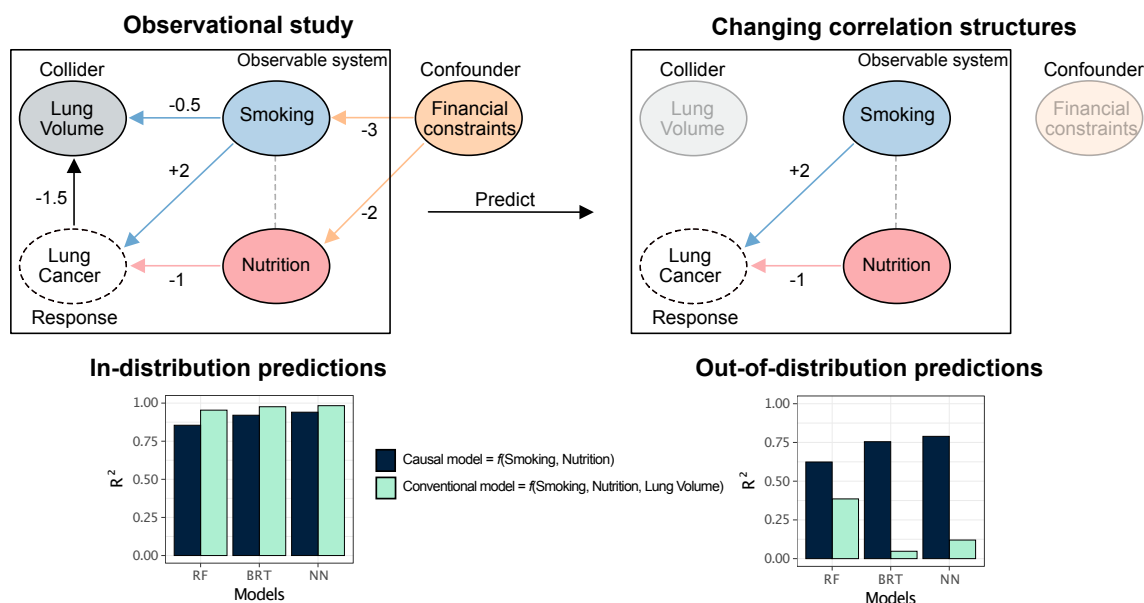
Models with hyperparameters tuned for inference had, on average, lower errors than when using hyperparameters tuned for prediction (Figure 5.3, Table S4.2). This confirms that the bias-variance tradeoff is different for prediction and inference tasks.

#### 5.2.4 Case Study – Predicting out-of-distribution

Having seen that NN can infer near-unbiased estimates under collinearity (given enough observations), it seems contradictory that algorithms such as RF, with strongly biased effect estimates, often outperform NNs in predictive benchmarks. Such results, however, are usually obtained on test data that is out-of-sample but in-distribution, which means that the feature correlation in the hold-out data is identical to the training data. In such a case, we know a priori that the predictor  $\hat{y}$  can be unbiased whilst having biased estimated effects  $\hat{\beta}$  (causal spillover) (ARIF and MACNEIL, 2022a; SHMUELI, 2010). If we predict out-of-distribution, however, for example to predict the effect of interventions or when the correlation structures change (e.g., latent confounders) (Figure 5.4), it will be much less likely that a non-causal model delivers unbiased predictions (KE *et al.*, 2019; PEARL and MACKENZIE, 2018).

To demonstrate this phenomenon, we simulated a case study where we assumed that the goal is to predict lung cancer based on smoking and diet (Figure 5.4) in two different scenarios. In this case

study, we assume models are first trained and validated to predict lung cancer in an observational study and are then used to predict lung cancer in a randomized controlled trial (RCT) (Figure 5.4). By means of the control, the RCT forced predictions to be out-of-distribution. Specifically, we assume that the collider lung volume and the latent confounder financial constraints were controlled in the selection of trial participants, which means that they no longer correlate with the treatment (Figure 5.4). We trained three ML algorithms (RF, BRT, and NN) with two different feature selections, a conventional (full) model with all features (smoking, diet, and lung volume) and a causally constrained model with only smoking and diet used as features.



**FIGURE 5.4:** Difference between causal and conventional ML models for in-distribution and out-of-distribution predictions in a simulated case study. We assume that a first study collected data about the effects of smoking and nutrition on lung cancer. Lung volume is a collider and financial constraints is an unobservable confounder. Smoking and nutrition are correlated because of their latent confounder financial constraints. Then, the data from the observational study (left column) was used to train two different models, causally constrained (causal model) and a conventional model with all features, and to predict lung cancer in another observational study (in-distribution predictions) and in a clinical randomized controlled trial (RCT) (out-of-distribution predictions). In the RCT, patients were treated for lung volume, and received financial support (right side). Lung volume was removed as feature in the causal model because its inclusion would lead to biased effect estimates of smoking and nutrition (collider bias). Smoking and nutrition were both included to block the effect of the unobservable confounder on lung cancer (i.e., lung cancer and financial constraints are d-separated). In the first prediction scenario, the conventional model slightly outperformed the causal model (as measured by  $R^2$ ), whereas in the second, out-of-distribution model, the causal model outperformed the conventional model.

We find that in-distribution, the unconstrained model that uses all features outperformed the causally constrained models. In the second prediction scenario (out-of-distribution), however, the causally constrained models outperformed the conventional (full) model (Figure 5.4). The reason is that including the collider in the training creates a collider bias which biases the effect of smoking. Without the collider, the causal effect of smoking is estimated with lower bias, which reduces the out-of-distribution prediction error (Figure 5.4).

The case study also confirms our previous results that NN and BRT perform better than RF in



estimating the true causal effects and thus in the out-of-distribution tasks. RF is unable to correctly separate collinear features (Figure 5.1), leading to causal spillover in RF between diet and smoking during training (Figure 5.4). However, RF achieves a lower prediction error for the full model (with collider) than BRT and NN, probably by chance because the causal spillover inadvertently leads to advantageous biases due to the collider (Figure 5.4).

## 5.3 Discussion

The aim of our study was to understand if ML and DL algorithms display inherent biases, caused by algorithmic features and regularization methods, that prevents them to separate causal effects in the presence of feature collinearity. Our main finding is that this is indeed partly the case, but not to the same extent for all algorithms and hyperparameter combinations. Particularly NN and BRT, when tuned appropriately, showed surprisingly low bias for the estimated effects under feature collinearity (Figure 5.1, S2.5, S2.6), which allows them to correctly adjust for confounding and other causal structures if the feature selection is causally constrained. This means that if causal connections between features and the response are known, ML algorithms and in particular NNs appear to be a viable alternative to statistical models for adjusting for confounders and estimating feature effects.

### 5.3.1 Understanding the mechanism behind biased of feature effects under collinearity

The different susceptibility of the examined ML algorithms to bias induced by collinearity is presumably the result of different explicit and implicit algorithmic regularization mechanisms in these algorithms. For the elastic net, the regularization and thus the cause of the bias is explicit (and there is work to correct the models for the spillover bias). Also, for RF, it is relatively clear that the random subsampling of features creates an implicit regularization which explains the strong causal spillover observed in our simulations. For other algorithms such as BRT, we can only speculate about the mechanisms behind the observed biases: Our naive BRT implementation showed that pure boosting with linear models can be unbiased (Figure S4.3), while boosting with regression trees (Figure S4.3) can lead to either causal spillover or causal greediness (Figure S4.3). In our simulations, state-of-the-art BRT implementation (used in Figs. 5.2,5.3,5.4) seem to prevent the causal greediness effect and only displayed causal spillover. Note that this is even though we specifically avoided ‘extreme boosting’, which introduces boosting and dropout into BRT (CHEN and GUESTRIN, 2016), which would likely cause additional spillover.

It was often reported that also NNs exhibit a so-called simplicity bias in their predictions with a negative impact on their generalizability, potentially caused by the stochastic gradient descent and not wide enough layers (HUH *et al.*, 2021; SHAH *et al.*, 2020; VALLE-PÉREZ, CAMARGO, and LOUIS, 2019). A predictive simplicity bias should transfer to feature effects, suggesting that also NNs should exhibit causal spillover. We did not find such an effect for unregularized NNs, but we did find that with strong collinearity, both boosting and NN required far more boosting respectively optimization steps that what is needed to obtain reasonable predictive errors until they successfully separated the features (Figure S4.4, S4.13). Reported simplicity biases could thus also be explained by the common approach to stop training once the cross-validation loss does not further improve.

### 5.3.2 Hyperparameters control bias-variance tradeoff for effect estimates

For all algorithms, hyperparameters had substantial effects on the observed biases, especially in data-poor situations (Figure 5.2). While some effects, for example the regularization parameters in the elastic net, were as expected, others like the choice of the activation function in the NN were surprising: SELU strongly reduced the bias of effect estimates and prediction errors (Figure 5.2). The fact that this SELU effect diminished with increasing number of observations, that we used structured (tabular) data, or that we used a regression and not a classification task (RADHAKRISHNAN, BELKIN, and UHLER, 2023) may explain why this was not discovered before (Figure S4.9).

Hyperparameters often had opposite effects on the bias and variance of effect estimates (and prediction errors) (Figure 5.2), reflecting the expected tradeoff between bias-variance when tuning regularization parameters. More importantly, however, the shape of this bias-variance tradeoffs differed for effect estimates versus prediction errors (e.g., `mtry` in RF), resulting in different sets of optimal hyperparameters (Figure 5.2). This confirms the common expectation that there is a tradeoff between tuning models for prediction and explanation. However, the difference between the two was not large, which is reassuring, given that in practical applications, hyperparameter tuning is only possible for the prediction error.

### 5.3.3 Advantages and challenges when using ML models for inference

A question that remains is why and when we should prefer a causally constrained ML algorithms over OLS, which has the advantage of being the best linear unbiased estimator (BLUE). We believe in practice, there are two major drawbacks of OLS or other parametric regression models. First, an OLS requires that the model structure is specified a priori. If this structure is incorrectly specified, the effect of confounder, for example, may not correctly be adjusted, which can induce bias and causal spillover (cf. BREIMAN, 2001b). Related to this, in practice, sample sizes are often prohibitively small for specifying a model with all possible effects, which that analysts either must make ad hoc decisions or accept that the variance of estimates is high and thus the power to see effects low. Machine learning approaches can potentially better tradeoff bias against variance, and it is further possible to tune this tradeoff to metrics that are particularly important for practitioners. For example, our trained ML models had a high reliability at identifying zero effects in the data-poor situations where the OLS failed to fit (Figure 5.3). Among the ML and DL models, elastic-net showed the lowest errors, but we note that this was for a classical elastic net on top of an OLS where we prescribed linear effects. In this case, this worked well because we simulated data with linear effects, but we assume that in real-world scenarios NN will outperform elastic-net unless for the presence of nonlinear effects or feature interactions happens to be guessed exactly right (Figure S2.1).

Comparing our approach to other causal ML algorithms, we see the closest resemblance to double / debiased ML, which uses a two-step process, where in a first step, two models are trained to predict the explanatory variable and the response based on the confounder (adjustment step), and then a final model is trained on the residuals of the first models to estimate the (adjusted) effect of the predictor (estimation step) (CHERNOZHUKOV *et al.*, 2018). The validity of this approach was first proven for OLS (ROBINSON, 1988) and depends only on the unbiasedness of the models involved (CHERNOZHUKOV *et al.*, 2018). For OLS, the approach does not provide any advantage over a direct adjustment in a multiple regression for linear effects. For ML models, the advantage is that the adjustment and estimation models can be independently tuned and chosen. Our approach, on the other hand, which essentially generalizes a multiple regression model, seems to

us easier to implement, understand, and may have advantages in particular predictive scenarios. Another alternative is the popular causal forest algorithm, which also essentially corrects for confounders while predicting the effect of a target feature. We view it as a task for further research to better understand the practical advantages and disadvantages of these alternative approaches. In particular, we believe it would be important to understand which of those approaches leads to a lower error on the estimated causal effects in situations that are representative for practical analysis in psychology, economics, medicine or ecology.

A limitation of all approaches discussed here, including OLS, is that they assume that the causal relationship between the characteristics is known a priori, so that our task is only to adjust for it. In practice, this assumption can often be met because the directions of the effects can be inferred from existing scientific knowledge, but when this is not the case, they must be estimated from the data, which is still extremely challenging.

#### 5.3.4 Advantages of causally constrained models for out-of-distribution predictions

Contradictory to the general assumptions that good predictive and explanatory models differ, we show that causal constraints that aid the model in learning the true underlying causal structure can also aid predictions when the collinearity structure of the feature space changes (out-of-distribution). This is not particularly surprising because it is well-known, even for statistical models, that selecting features causally is not necessarily beneficial for obtaining the lowest in-distribution prediction error, but it may help for out-of-distribution predictions where feature collinearity is changed (DORMANN, SCHYMANSKI, *et al.*, 2012)

Using a hypothetical example of predictions of lung cancer risk in an observational study and a clinical trial, we highlight that these effects could have important real-world applications. In our example, we find that a non-causal model has lower predictive in-distribution error, but higher out-of-distribution error compared to a causally constrained ML model (Figure 5.4). We note that apart from the fact that the causally constrained model generalizes better, it has the additional advantage of being interpretable in terms of causal effects, which is of interest for science and clinical practitioners, but possibly also for questions of fairness in AI (BARREDO ARRIETA *et al.*, 2020).

While it is generally understandable why certain algorithms (in particular RF) show higher biases on inferred feature effect under collinearity (see above), we wonder if these effects have any advantages for in-distribution predictions. It is interesting that state-of-the art BRT algorithms that often show the best performance on tabular data added algorithmic features similar to the random forest on the vanilla algorithm that has lower biases on the effects. We speculate that the spillover caused by the model averaging underlying these additions may actually be helpful in improving stability and reducing variance of the predictions, thus suggesting again that some algorithms may be better suited for in-distribution predictions, while others are better suited for inference or out-of-distribution predictions.

#### 5.3.5 Conclusion

Certain ML and DL algorithms, in particular neural networks, can approximately estimate the effect of one or several target features, adjusted for the effect of other features. Thus, these models can in principle be used like a multiple regression, and if needed, confidence intervals and p-values could be calculated on top based on bootstrapping. The observations that such causally constrained models may have larger in-distribution but lower out-of-distribution predictive errors,

together with the fact that tuning hyperparameters for prediction is often a good proxy for inference as well suggests to us that the tradeoff between predictive modelling and inference may not be as wide and deep as often assumed.

These results have significant implications for both predictive and explanatory modeling. For predictive modelling, they suggest that causally constraining ML and DL models can reduce out-of-distribution prediction error, which may often be a practically relevant objective. For explanatory modeling, it shows that ML algorithms such as BRT and NN can produce reliable inferences. Although more research is needed to better understand their biases and offer appropriate statistical guarantees on effect estimates, their higher flexibility provides at least the theoretical perspective that they could outperform traditional methods in situations with many nonlinearities or higher order interactions, which may actually account for the majority of applied statistical analyses of observational data.

## 5.4 Methods

Statistical analysis and simulations were conducted in R (version 4.0.5, R CORE TEAM, 2021). All code for reproducing our analysis can be found in <https://doi.org/10.5281/zenodo.8052354>. We additionally archive this code in persistent repository upon acceptance of the manuscript.

### 5.4.1 Definition of average conditional effects

To extract the feature effects in a trained ML or DL model, we use average conditional effects (ACE), which are also known under the name average marginal effects. Consider a feature matrix  $X = (x_1, \dots, x_k)^\top$  with  $k$  feature vectors and a response vector  $y$  with their true relationship  $y = f(X)$  and  $\hat{f}(\cdot)$  is estimated by ML algorithms. Because the trained relationship can be highly complex, we find different conditional effects ( $CE_{ik}$ ) (or interactions) for each observation  $i$  of the  $k$ -th feature vector in the feature space. The  $\mathbf{CE}_k$  for feature vector  $\mathbf{x}_k$  is then  $\mathbf{CE}_k = \frac{\partial \hat{f}(\mathbf{X})}{\partial \mathbf{x}_k}$  which is approximated by  $\mathbf{CE}_k \approx \frac{\hat{f}(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k + h, \dots, \mathbf{x}_j) - \hat{f}(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k, \dots, \mathbf{x}_j)}{h}$ ,  $h, h > 0$ . The conditional effects for  $\mathbf{x}_k$  ( $\mathbf{CE}_k$ ) are then averaged to  $ACE_k$ .

For linear effects, any average will produce an ACE that asymptotically corresponds to the coefficients in linear regression models  $y = \beta_1 \mathbf{x}_1 + \dots + \beta_k \mathbf{x}_k$ ,  $\beta_k \approx ACE_k$ . For non-linear feature effects, the problems arise that dense areas in the feature space would be overrepresented in an arithmetic average. There have been several proposals how to average in such a case (SCHOLBECK *et al.*, 2022). As we did not consider nonlinear effects in our simulation, our results are not affected by this problem, but in general, we propose to average the  $ACE_k \sum_{i=1}^N w_i CE_{ik}$ , with weights  $w_i$  proportional to the inverse of the estimated density in the feature space of  $\mathbf{x}_k$ .

### 5.4.2 Near-asymptotic performance

We first simulated two different scenarios (Figure 5.1, first column) with a large sample size of 1000 observations. This sample size is large enough so that effects of stochasticity induced by the data generation process and default hyperparameters for each model can be neglected. The two scenarios were (a) a base scenario with three independent features, one without an effect, and (b) a mediator scenario with two features forming a mediator path and a third feature independently affecting the response (for more details, see Methods). We fitted linear regression models (LM) to each scenario as a reference and compared the estimates to the effects learned by the ML models extracted by the ACE.

We simulated two scenarios with different collinearity structures. In all three scenarios we simulated five features ( $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4, \mathbf{x}_5$ ) and one response vector  $\mathbf{y}$ . In the first scenario, the data generating model was  $y \sim N(1.0 \cdot x_1 + 0.0 \cdot x_2 + 1.0 \cdot x_3 + 0.0 \cdot x_4 + 0.0 \cdot x_5, \sigma)$  with  $\sigma = 0.3$  and all five features independent of each other (no collinearity). The feature matrix  $\mathbf{X}$  was sampled from a multivariate normal distribution with mean vector  $\mu = 0$  and the covariance matrix being the identity. In the second scenario, the data generating model was the same but  $\Sigma$  which was used to sample the feature matrix  $\mathbf{X}$  had an entry of 0.9 ( $\Sigma_{1,2} = \Sigma_{2,1} = 0.9$ ) so that  $x_1$  and  $x_2$  were highly correlated. We sampled from each scenario 1000 observations.

## Model fitting and evaluation

We fitted RF (WRIGHT and ZIEGLER, 2017, 100 trees), BRT (CHEN and GUESTRIN, 2016, 2016; 140 trees; ‘req:squarederror’ objective function), NN (AMESÖDER and PICHLER, 2023; three hidden layers with each 50 units; reLU activation functions; batch size of 100; AdaMax optimizer; learning rate of 0.01; 32 epochs), linear regression model (lm function), and glmnet (FRIEDMAN, HASTIE, and TIBSHIRANI, 2010; OOI, 2021; alpha = 0.2; lambda was tuned via 10-fold) to the data generated by the three scenarios (1,000 observations) ( $\mathbf{X}$  as feature matrix and  $\mathbf{y}$  as response vector). Afterward, we calculated the individual ACE for each of the five features. We repeated the procedure (sampling from the scenarios and fitting the models to the data) 100 times and averaged the results.

As the simulated effects are linear, the theoretical ACE are equivalent to the true linear effects used in the data generating models:  $ACE_k \approx \beta_k$ . To assess bias and variance, we calculated the bias  $Bias = \beta_k - ACE_k$  and the variance of the  $ACE_k$  over 500 replicates for all five features.

### 5.4.3 Performance in data-poor situations

We assume that we are interested in two effects,  $\beta_1 = 1.0$  and  $\beta_2 = 0.0$ . The other effects were equally spaced between zero and 1.0. Features were sampled from a multivariate normal distribution with a covariance matrix ( $\Sigma$ ) sampled from a LKJ distribution ( $\eta = 2$ ) so that the features were weakly correlated on average. We calculated bias and the variance for the two target effects and all models.

The data generating model was  $\mathbf{y} \sim N(\mathbf{X}\beta, \sigma)$  with  $\sigma = 0.3$  with  $\mathbf{y}$  being the feature matrix (100 features) and  $\beta$  the effect vector.  $\mathbf{X}$  was sampled from a multivariate normal distribution with mean vector  $\mu = 0$  and distribution and the covariance matrix ( $\Sigma$ ) was sampled from a LKJ distribution ( $\eta = 2$ ) so that the features were weakly correlated on average. Effects were  $\beta_1 = 1.0$  and  $\beta_2 = 0.0$  and rest of the effects (98) were equally spaced between 0.0 and 1.0.

## Hyperparameter tuning

We performed a hyperparameter search to check if and how hyperparameters influence differently or equally effect estimates and the prediction error, so does a model tune after the prediction error has biased effects? For that, we created data-poor simulation scenarios with the above described data generating model and 50, 100, and 2000 observations and 100 features with effects ( $\beta_i, i = 1, \dots, 100$ ),  $\beta_1 = 1.0$ , and  $\beta_2$  to  $\beta_3$  were equally spaced between 0.0 to 1.0 so that  $\beta_2 = 0.0$  and  $\beta_{100} = 1.0$ .

Features were sampled from a multivariate normal distribution and all features were randomly correlated (Variance-covariance matrix  $\Sigma$ ) was sampled from an LKJ-distribution with  $\eta = 2.0$ .

1,000 combinations of hyperparameters were randomly drawn (Table S4.1). For each draw of

hyperparameters, the data simulation and model fitting were repeated 20 times.  $\widehat{ACE}_1$  and  $\widehat{ACE}_2$  were recorded (for each hyperparameter combination and for each repetition). Bias, variance, and mean square error (MSE) were calculated for estimated effects and the average (over the 20 repetition) MSE for predictions on a holdout of the same size as the training data.

To understand how hyperparameters affect bias, variance, and MSE of estimated effects and predictions, we fitted generalized additive model (GAM) on the hyperparameters with the respective errors as response. The average responses were first subtracted from the responses to set the intercept to 0 (we suppressed the intercept in the GAMs because we were not interested in a reference level). We also fitted a random forest (2000 trees to get stable effects) on the hyperparameters to get variable importances for all hyperparameters.

To get the optimal hyperparameters, we fitted random forest models on the hyperparameters of the MSE for the estimated effect  $\hat{\beta}_1$  and the predictions  $\hat{y}$ . We then predicted for all hyperparameters and selected the hyperparameters with the lowest MSE (Table S4.2, Table S4.3).

### **Model fitting and evaluation**

We fitted RF (WRIGHT and ZIEGLER, 2017), BRT (CHEN and GUESTRIN, 2016), NN (AMESÖDER and PICHLER, 2023), linear regression model (lm function), and elastic-net (FRIEDMAN, HASTIE, and TIBSHIRANI, 2010; OOI, 2021) to the data generated for 50, 100, and 600 observations.

We calculated the individual ACE for the first two effects  $\beta_1 = 1.0$  and  $\beta_2 = 0.0$ . We repeated the procedure (sampling from the scenarios and fitting the models to the data) including the sampling of the covariance matrix  $\Sigma$  1000 times. We calculated bias and variance for both effects.

**Acknowledgements** We thank Tankred Ott and Merle Behr for their valuable comments on the manuscript.

**Data availability statements** Code to reproduce the analysis can be found in the following repository <https://doi.org/10.5281/zenodo.8052354>.

---

## CITO: AN R PACKAGE FOR TRAINING NEURAL NETWORKS USING TORCH

---

**Christian Amesoeder, Florian Hartig, and Maximilian Pichler**

in prep., preprint available at [2303.09599](https://doi.org/10.2303.09599)

### **Abstract**

Deep Neural Networks (DNN) have become a central method in ecology. Most current deep learning (DL) applications rely on one of the major deep learning frameworks, in particular Torch or TensorFlow, to build and train DNN. Using these frameworks, however, requires substantially more experience and time than typical regression functions in the R environment. Here, we present 'cito', a user-friendly R package for DL that allows specifying DNNs in the familiar formula syntax used by many R packages. To fit the models, 'cito' uses 'torch', taking advantage of the numerically optimized torch library, including the ability to switch between training models on the CPU or the graphics processing unit (GPU) (which allows to efficiently train large DNN). Moreover, 'cito' includes many user-friendly functions for model plotting and analysis, including optional confidence intervals (CIs) based on bootstraps for predictions and explainable AI (xAI) metrics for effect sizes and variable importance with CIs and p-values. To showcase a typical analysis pipeline using 'cito', including its built-in xAI features to explore the trained DNN, we build a species distribution model of the African elephant. We hope that by providing a user-friendly R framework to specify, deploy and interpret DNN, 'cito' will make this interesting model class more accessible to ecological data analysis. A stable version of 'cito' can be installed from the comprehensive R archive network (CRAN).

**Keywords:** R language, Machine Learning, Regression, Classification, Species distribution model

## 6.1 Introduction

Deep neural networks (DNN) are increasingly used in ecology and evolution for regression and classification tasks such as species distribution models, image classification or sound analysis (CHRISTIN, HERVET, and LECOMTE, 2019; JOSEPH, 2020a; PICHLER and HARTIG, 2023b; STRYDOM *et al.*, 2021). State-of-the-art DNN are almost exclusively implemented and trained in specialized deep learning (DL) frameworks such as PyTorch or Tensorflow (Abadi *et al.*, 2016; Paszke *et al.*, 2019). These frameworks, most of which are implemented in Python, provide flexible and performant functions and classes that allow users to implement and train complex DL architectures, such as large language models (e.g., GPT-3, BROWN, MANN, *et al.*, 2020; RoBERTA, LIU, OTT, *et al.*, 2019) or complex object detection models (e.g., Mask R-CNN, HE, GKIOXARI, *et al.*, 2017; DeepVit, ZHOU *et al.*, 2021). Their high level of flexibility is appealing to “power users”, but the complexity of these frameworks can be prohibitive or at least repelling for scientists with limited knowledge in the field that merely want to use neural networks in standard applications.

As a response to this problem, several simplified frontends for the major DL frameworks have been developed. Many of those are also available in R, the language used by most ecologists for practical data analysis. Well-known examples are ‘Keras’ for TensorFlow and `luz` for ‘torch’ (CHOLLET, ALLAIRE, *et al.*, 2017; FALBEL, 2023). However, while these frontends indeed simplify the model building process considerably, their general structure and syntax still resembles those of the major Python frameworks rather than those of popular R packages for regression or classification tasks that specify models using the formula syntax such as ‘ranger’, for training random forests, or ‘lme4’, for training mixed-effect models (BATES *et al.*, 2014; WRIGHT and ZIEGLER, 2017). Moreover, DL frontends such as ‘Keras’ or ‘luz’ mainly concentrate on model fitting and include only a very limited set of plots and convenience functions which are common to most R packages. As a result, working with these frontends still requires a considerable amount of training for users that are so far only familiar with standard R packages. Especially because users have to choose or program code for downstream tasks such as bootstrapping, plots or explainable AI (xAI) metrics by hand.

Besides the mentioned frontends to the major DL frameworks, some specialized R packages for training DNN exist that more closely adhere to the syntax used in most popular R packages, in particular the formula syntax to specify the model structure. However, those packages often lack crucial functionalities, and most of them do not make use of state-of-the-art DL frameworks for model fitting. This limits their use for large DNN because of their numerical inefficiency or their inability to train the models on GPUs. Established R packages such as ‘nnet’ or ‘neuralnet’ do not support modern DL techniques, such as different regularization techniques (e.g. dropout) to control the bias-variance tradeoff (FRITSCH, GUENTHER, and WRIGHT, 2019; VENABLES and RIPLEY, 2002) or modern training techniques such as early stopping or learning rate schedulers that help to achieve convergence. The ‘h2o’ package comes with its own Java backend, and while it allows specifying models with the standard formula syntax, its use in R is cumbersome due to its inability to work with default R objects (FRYDA *et al.*, 2023). The ‘brulee’ R package (KUHN and FALBEL, 2022), which uses ‘torch’ to train the DNNs specified in standard R syntax, is very similar to the package presented here, but still lacks some critical features (see section ‘Performance analysis and validation’).

Here, we present ‘cito’, an R package for training fully-connected neural networks using the standard R formula syntax for model specification. Based on the ‘torch’ DL framework, ‘cito’ allows flexible specifying of fully-connected neural networks architectures, supports many modern DL techniques (e.g. dropout and elastic net regularization, learning rate schedulers), can take advantage of CPU and GPU hardware for parallelization, and, despite its simple user interface,



optionally offers a high degree of customization such as user-defined loss functions. Moreover, 'cito' supports many downstream functionalities, such as the possibility to continue the training of existing DNN with modified training parameters for fine-tuning, or the application of explainable AI (xAI) methods to interpret the trained models. As such, 'cito' provides a user-friendly but nevertheless complete analysis pipeline for building neural networks in R.

In the remainder of the paper, we introduce the design principles of 'cito' in more detail, show validation and performance analysis, and showcase the application of cito using the example of a species distribution model of the African elephant.

## 6.2 Design of the cito package

### 6.2.1 Torch backend

'cito' uses 'torch', a variant of PyTorch, as its backend to represent and train the specified neural networks. Until recently, R users who wanted to use PyTorch and Tensorflow had to call their Python bindings through the 'reticulate' package. R packages that relied on this pipeline were thus dependent on appropriate Python installations (e.g. PICHLER and HARTIG, 2021a), which often created dependency issues. This issue got solved with the release of 'torch', a native implementation of the torch libraries with an R frontend (FALBEL and LURASCHI, 2023).

### 6.2.2 Building and training neural networks in cito

With 'torch', R users can essentially use PyTorch natively in R, which solves dependency issues, but not the problem that specifying a DNN with 'torch' is complex.

'cito' addresses this problem by providing one simple command, `dnn()`, which combines everything needed to build and train a fully-connected neural network in one line of code (see Supporting Information S6 for details). The `dnn()` function includes options to modify the network architecture, the training process and the monitoring (e.g. by visualization) of the training and validation loss (Table 6.1), including a baseline loss (based on intercept-only models) that helps to diagnose convergence problems due to inappropriately chosen training hyper-parameters (e.g., learning rate and epochs).

The `dnn()` function returns an S3 object that can be used, for example, with the `continue_training()` function to continue training for additional epochs (iterations) with the same or modified training hyper-parameters or data. Moreover, many standard R functions such as `summary()`, `predict()` or `residuals()` are implemented for the trained models, and additional specialized explainable xAI functions are available for interpreting the fitted networks. More details on these and other functions are available in the R package vignettes that come with the cito package.

The lack of uncertainties (standard errors) is an often-raised concern for DNN. In 'cito', we provide an option to automatically calculate confidence intervals for all outputs (including xAI metrics and predictions) using bootstrapping. As bootstrapping can be computationally expensive, the default for this option is set to false. Bootstrapping can be enabled in the `dnn()` function setting, e.g., `dnn(..., bootstrap = 50)`. Bootstrap standard errors are then automatically propagated through all downstream methods and are also used to generate p-values wherever obvious null hypotheses exist. We recommend starting without bootstrapping to optimize the training procedure (Fig. 2) and to then enable the bootstrap for the final model after the training pipeline has been finalized.

**TABLE 6.1:** Hyperparameters for fully-connected neural networks and their default values in 'cito'. Defaults for all parameters are set to sensible values; however, some parameters typically need to be tuned. Detailed guidance on this is provided in the help file of the `dnn()` function or in the cito R package vignette "Training neural networks".

Architecture		
Name	Explanation	Default
hidden	Quantity and size of hidden Layers	(50, 50)
activation	Activation function for hidden layers	"selu"
bias	Should hidden nodes have bias	TRUE
Training		
Name	Explanation	Default
validation	Split data into test and validation set	0
epochs	Number of training iterations	100
device	Set to "cuda" to train on GPU	"cpu"
plot	Visualize loss during training	TRUE
batchsize	Number of samples used for each training step	32
shuffle	Shuffle batches in between epochs	TRUE
lr	Learning rate	0.01
early_stopping	Stops training early based on validation loss	FALSE
Bootstrap	Number of bootstrap samples	FALSE
Controlling bias-variance trade-off (regularization)		
Name	Explanation	Default
lambda	Strength of elastic net regularization	0
alpha	Split of L1 and L2 regularization	0.5
dropout	Dropout probability of a node	0

### 6.3 Performance comparison and validation of cito

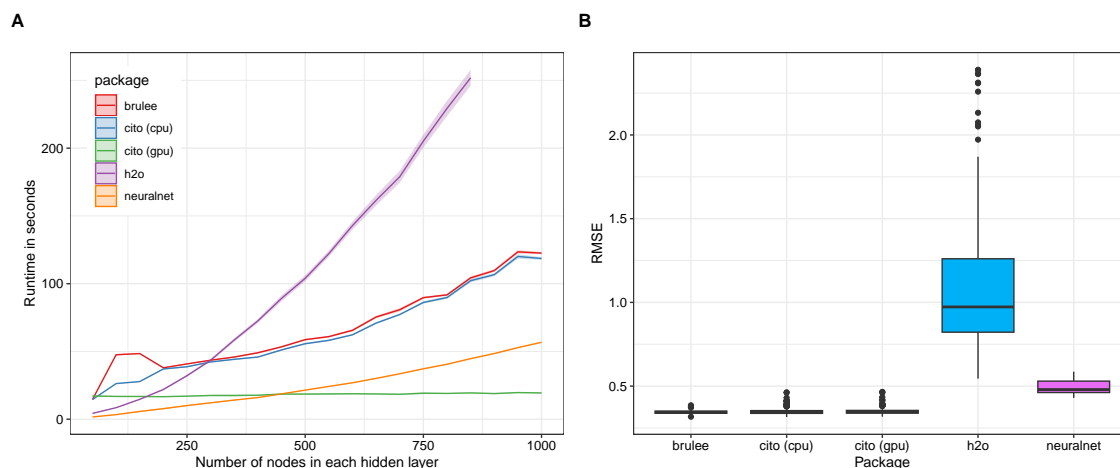
After explaining the design of cito, we shortly compare its performance and functionality with other packages for implementing neural networks in R. We consider in particular 'nnet' and 'neuralnet', which each have their own backend and are not based on modern DL frameworks (FRITSCH, GUENTHER, and WRIGHT, 2019; VENABLES and RIPLEY, 2002), 'h2o', which possesses a much broader toolkit for training neural networks than the previous two packages (FRYDA *et al.*, 2023), and 'brulee' (KUHN and FALBEL, 2022), which, similar to cito, uses the 'torch' DL framework as a backend.

Our comparison shows that ‘cito’ implements more options than other packages, in particular GPU support, the possibility to continue training and custom loss functions and most importantly tools to interpret the trained DNN models (Table 6.2).

**TABLE 6.2:** Feature comparison of R packages used to build fully-connected neural networks

	‘cito’	‘brulee’	‘h2o’	‘neuralnet’	‘nnet’
Customizable network architecture	X	X	X	X	
Fit a probability distribution	X		X		
GPU support	X				
Regularization	X	X	X	X	X
Custom loss function	X			X	
Optimization of additional user-defined parameters	X				
Continue training	X				
Class weights for imbalanced data		X			
Learning rate scheduler	X	X	X		
Feature importance (xAI)	X		X		
Partial dependency plots (xAI)	X				
Accumulated local effect plots (xAI)	X				
Uncertainty (confidence intervals and p-values for xAI metrics and predictions)	X				
Baseline loss (to help with the convergence)	X				

Looking at computational performance, measured by the time it takes to train the networks, we find that some of the older packages, in particular ‘neuralnet’, perform better than the torch-based packages (including ‘cito’) for small networks (Fig. 6.1)). This is probably due to the smaller overhead of these more specialized packages. However, when moving to larger networks (large and especially wide networks are often beneficial for achieving low generalization errors (BELKIN *et al.*, 2019)) ‘cito’ can play out one of the main advantage of modern ML frameworks, which is GPU support. On the GPU, training time in cito is practically independent of the size of the network, confirming the consensus that training large networks requires GPU resources. On a CPU, ‘cito’ performs on par with ‘brulee’, the other torch-based package, but somewhat worse than ‘neuralnet’. We interpret these results as showing that for a simple problem, there is still some overhead of using ‘torch’ as opposed to a native C implementation. Nevertheless, we would argue that the added flexibility and functionality of cito outweighs this advantage of ‘neuralnet’. Moreover, our results suggest that the difference between the torch packages and ‘neuralnet’ lies mainly in the constant overhead needed to set up the models. For large models, their performance is roughly equal.

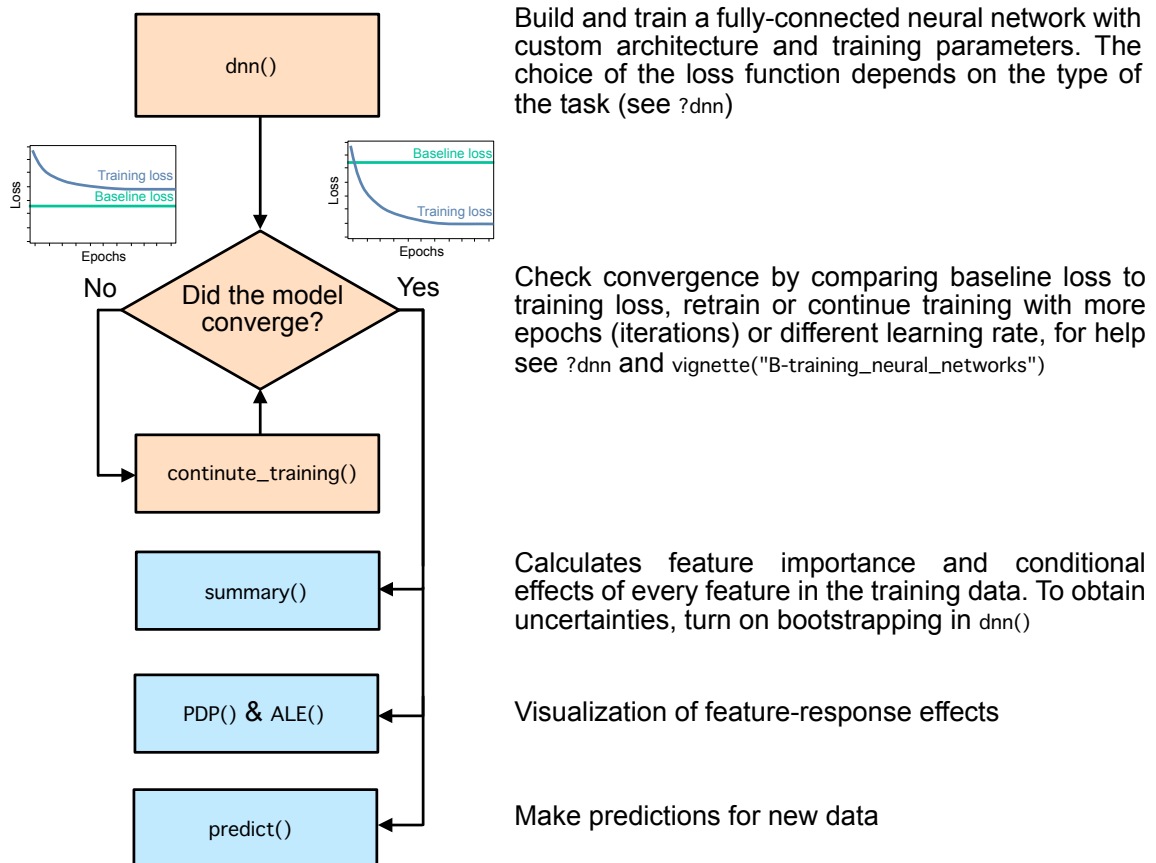


**FIGURE 6.1:** Runtime comparison of different deep learning R software packages (`'brulee'`, `'h2o'`, `'neuralnet'`, and `'cito'` (CPU and GPU)) on different network sizes on an Intel Xeon 6128 and a Nvidia RTX 2080ti. The networks consisted of five equally sized layers (50 to 1000 nodes with a step size of 50) and are trained on a simulated data set with 1000 observations. Panel (A) shows the runtime of the different packages and panel (B) shows the average root mean square error (RMSE) of the models on a holdout of size 1000 observations (RMSE was averaged over different network sizes). Each network was trained 20 times (the dataset was resampled each time).

## 6.4 Workflow and case study

So far, we have mainly discussed the process of model training, which is arguably the core of any machine learning project. Now, we want to comment on the entire workflow when using `'cito'` to build and interpret a predictive model. This workflow usually consists of model specification, training, and interpretation and predictions (Fig. 2). To make the discussion of the workflow more accessible to the reader, we illustrate this workflow with the example (based on RYO *et al.*, 2021) of building a species distribution model (SDM) for the African elephant (*Loxodonta Africana*).

SDMs are niche models that correlate environment with species occurrence data (see ELITH and LEATHWICK, 2009). As occurrence data, we use records of African elephant presence from RYO *et al.* (2021) that was based on ANGELOV (2020), who compiled data from different studies available on GBIF (QUESTAGAME, 2023; MUSILA *et al.*, 2024; NAVARRO, 2024). Those presence-only data were supplemented by ANGELOV (2020) with randomly sampled background points (pseudo-absences) to generate a presence-absence signal for the classifier. As predictors, we used all 19 bioclimatic variables from WorldClim v2 (FOURCADE, BESNARD, and SECONDI, 2018), which were centered and standardized. While it is common in statistical modelling to sample more pseudo-absences than presences, such unbalanced class numbers can be harmful for machine learning algorithms. We therefore randomly undersampled pseudo-absences to match the number of observations (another option would be to oversample presences, but in our example, this resulted in lower accuracy in interim results).



**FIGURE 6.2:** Workflow of building, training and analyzing DNN with 'cito'. Example workflow and analyses for (multi) species distribution models are available as a vignette (vignette("C-Example\_Species\_distribution\_modeling")) or at <https://citoverse.github.io/cito/articles>

Building and training a species-distribution model based on a fully-connected neural network with three hidden layers of 50, 50 and 50 nodes and trains it for 50 epochs can be done in one line of code:

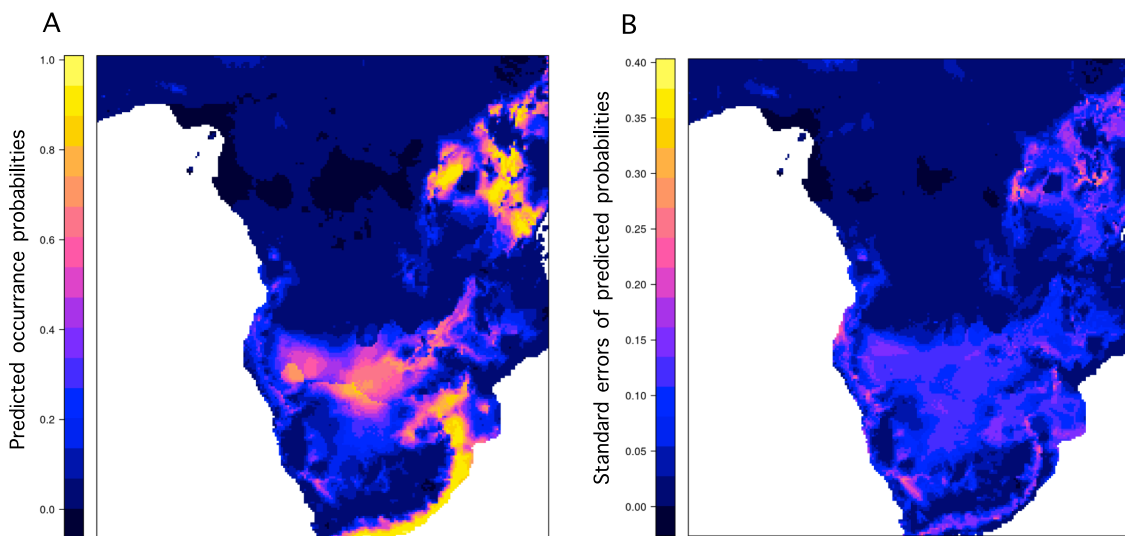
```
nn.fit <- dnn(label~., data = data,
             hidden = c(50, 50, 50), loss = "binomial",
             epochs = 50, lr = 0.1,
             batchsize = 300,
             validation = 0.1, shuffle = TRUE,
             alpha = 0.5, lambda = 0.005,
             early_stopping = 10,
             bootstrap = 30)
```

During training and without bootstrapping, a plot is displayed in R that monitors the training, validation and baseline loss. This plot can be used to diagnose convergence problems, for example if the training loss does not decrease over time or does not fall below the baseline loss. In this case, it would be advisable to abort and restart the training with different hyperparameters (e.g., smaller learning rate), use a learning rate scheduler, or perform a systematic hyperparameter tuning. We provide extensive help on this topic in the documentation and in a vignette (vignette("B-Training\_neural\_networks")). Here we show an example where we restart the train-

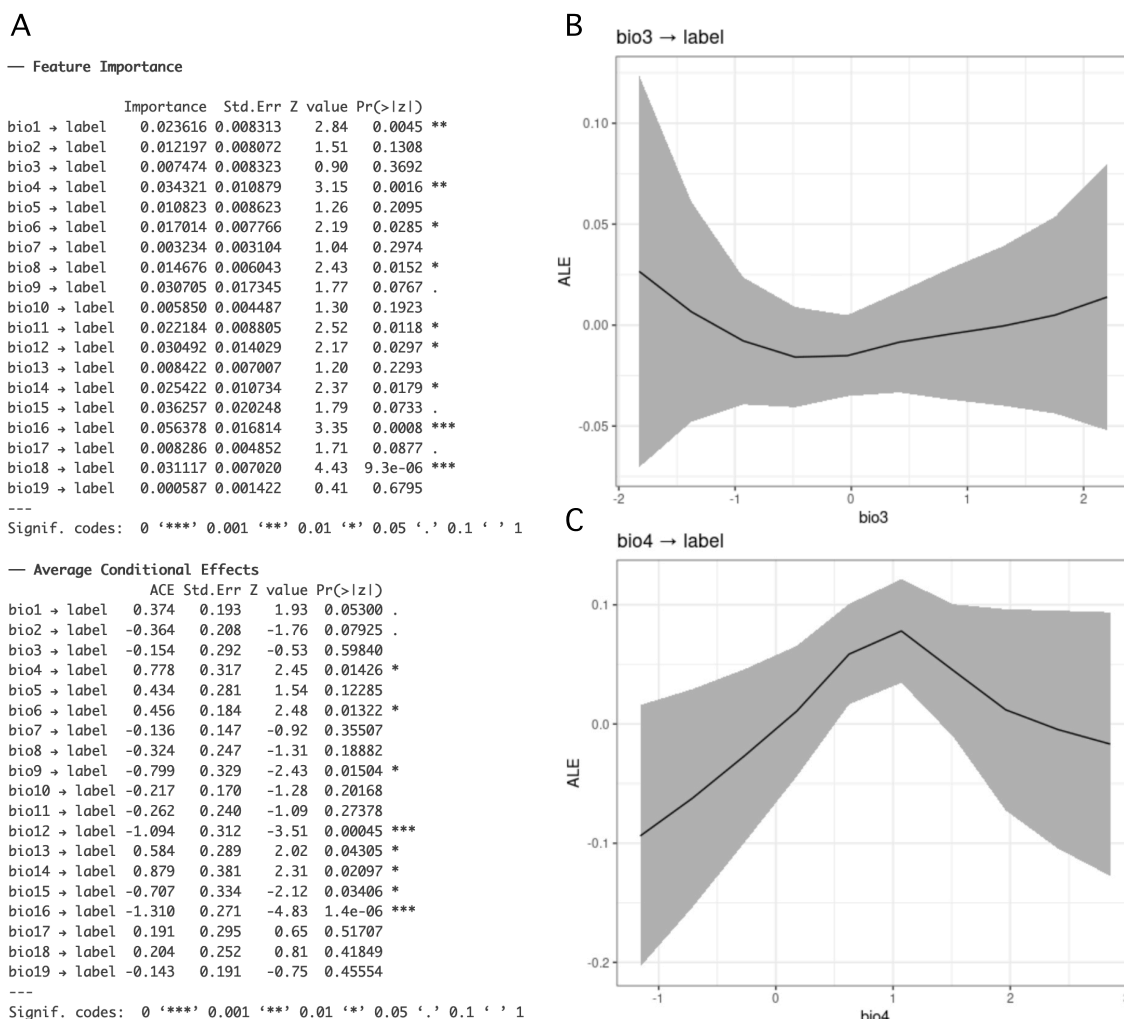
ing with a smaller learning rate and a learning rate scheduler that automatically reduces the learning rate if the loss does not decrease in 8 continuous epochs (patience =8) to achieve a better fit:

```
nn.fit <- continue_training(
  nn.fit,
  epochs = 150,
  changed_params =
    list(lr = 0.05,
         lr_scheduler =
           config_lr_scheduler("reduce_on_plateau",
                               patience = 8,
                               factor = 0.8))
)
```

The trained models can be used with a range of in-built functions. The `predict()` can be used to predict the occurrence probability of the elephant (Fig. 6.3a). The `summary()` function provides an overview about influential variables by calculating their importances (FISHER, RUDIN, and DOMINICI, 2018) as well as average conditional effects (which are an approximation of linear effects, see PICHLER and HARTIG, 2023a) (Fig. 6.4a). Partial dependency plots (PDP) and averaged local effect plots (ALE) functions can be used to display the effect of specific features on the response, in this case the occurrence probability of the elephant (Fig. 6.4b, c). If bootstrapping is enabled, 'cito' automatically uses the bootstrap samples to calculate confidence intervals (CI, as standard errors) for the predictions (Fig. 6.3a), CIs and p-values for the xAI metrics (Fig. 6.4a), and CIs for the PDP and ALE plots (Fig. 6.4b, c).



**FIGURE 6.3:** Predictions and standard errors of prediction for the African elephant from a DNN trained by cito. Panel (A) shows the predicted probability of occurrence of the African elephant. Panel (B) shows the standard error for the predicted probabilities (confidence interval).



**FIGURE 6.4:** xAI metrics with bootstrap confidence intervals (+/- 1 se) from model trained by 'cito'. Panel (A) shows (permutation) feature importances and average conditional effects (approximation of linear effects) from the summary() output for the 19 Bioclim variables. Panel (B) and (C) show the accumulated local effect plots (ALE), i.e., the change of the predicted occurrence probability, for the Bioclim variables 3 (Isothermality) and 4 (Temperature Seasonality).

## 6.5 Conclusion

'cito' is a powerful and versatile R package for building and training fully-connected neural networks with a formula syntax. The package seamlessly integrates into the R regression ecosystem and removes many hurdles in using neural networks for inexperienced users, but also saves programming time for experienced users who just want to build simple neural networks. The unique combination of features provided by 'cito', such as training on a GPU, using custom loss functions, baseline loss, confidence intervals, modern DL training techniques such as continue training, learning rate scheduler or early stopping cannot be found in other packages. Future releases of 'cito' aim to implement additional functionalities such as internal cross validation for hyperparameter optimization, gradient based methods for hyperparameter tuning and the integration of recurrent and convolutional neural networks.

---

**Acknowledgements** We thank Guillaume Blanchet and two anonymous reviewers for their valuable comments and suggestions.

**Data and code availability statements** The processed datasets for the species distribution model (African elephant) are available from ANGELOV, 2020. We used version 1.0.2 of the ‘cito’ package. The ‘cito’ package can be downloaded from CRAN, the code to reproduce the analysis and the benchmark can be found in the following repository <https://github.com/citoverse/A-mesoeder-et-al-2023>. Documentation and rendered vignettes (under articles) can be found on <https://citoverse.github.io/cito/> or on the CRAN website of the package at <https://cran.r-project.org/web/packages/cito/index.html>.



---

## MACHINE-LEARNING ALGORITHMS PREDICT SOIL SEED BANK PERSISTENCE FROM EASILY AVAILABLE TRAITS

---

Sergey Rosbakh, Maximilian Pichler, Peter Poschlod

Published in *Applied Vegetation Science*, 2022, 25, e12660, 10.1111/avsc.12660

### Abstract

**Question:** Soil seed banks (SSB), i.e., pools of viable seeds in the soil and on its surface, play a crucial role in plant biology and ecology. Information on seed persistence in soil is of great importance for fundamental and applied research, yet compiling data sets on this trait still requires enormous efforts. We asked whether the machine-learning (ML) approach could be used to infer and predict SSB properties of a regional flora based on easily available data. Location: Eighteen calcareous grasslands located along an elevational gradient of almost 2000 m in the Bavarian Alps, Germany.

**Methods:** We compared a commonly used ML model (random forest) with a conventional model (linear regression model) as to their ability to predict SSB presence/ absence and density using empirical data on SSB characteristics (environmental, seed traits and phylogenetic predictors). Further, we identified the most important determinants of seed persistence in soil for predicting qualitative and quantitative SSB characteristics using the ML approach.

**Results:** We demonstrated that the ML model predicts SSB characteristics significantly better than the linear regression model. A single set of predictors (either environment, or seed traits, or phylogenetic eigenvectors) was sufficient for the ML model to achieve high performance in predicting SSB characteristics. Importantly, we established that a few widely available SSB predictors can achieve high predictive power in the ML approach, suggesting a high flexibility of the developed approach for use in various study systems.

**Conclusions:** Our study provides a novel methodological approach that combines empirical knowledge on the determinants of SSB characteristics with a modern, flexible statistical approach based on ML. It clearly demonstrates that ML can be developed into a key tool to facilitate labor-intensive, costly and time-consuming functional trait research.

**Keywords:** artificial intelligence, persistence, predictive modeling, random forest, seed, soil, trait

## 7.1 Introduction

Soil seed banks (SSB), pools of viable seeds in the soil and on its surface, play a key role in plant biology and ecology at different levels of organization. They bridge short-and long-term environmental conditions temporarily unsuitable for growth and reproduction, especially in habitats subject to high climatic variability and high levels of disturbance, competition and predation (FENNER, FENNER, THOMPSON, *et al.*, 2005; SAATKAMP *et al.*, 2014). Regeneration resulting from persistent SSBs helps plants to recover the original state of populations and communities, including genetic diversity (HONNAY *et al.*, 2008), after they have been altered by environmental fluctuations (VANDVIK *et al.*, 2016). Thus, the ability of seeds to persist in the soil for long periods is a crucial bet-hedging strategy (HARPER *et al.*, 1977; ROSBAKH and POSCHLOD, 2021; VENABLE and BROWN, 1988) contributing to a plant's adaptive potential and ecosystem resilience, which is especially important in times of global change (OOI, 2012; WALCK *et al.*, 2011).

Species that are not able to persist in the soil at a local or regional scale are particularly vulnerable to extinction risk, whereas species with persistent SSBs can easily recover even after direct destruction of the above-ground vegetation (PLUE and COUSINS, 2018; PLUE, VAN CALSTER, *et al.*, 2021; STÖCKLIN and FISCHER, 1999). Thus, knowledge about species' ability to form persistent SSBs is of great importance for fundamental and applied research, such as nature conservation or restoration (BAKKER *et al.*, 1996; FAIST, FERRENBURG, and COLLINGE, 2013; WILLEMS and BIK, 1998) and management of invasive species (GIORIA, LE ROUX, *et al.*, 2019). However, compiling databases on seed persistence in soil requires enormous effort when collecting primary data: such studies are labor-intensive, costly, and time-consuming. As a result, the existing studies are limited to a few regions (e.g., temperate Europe PLUE, VAN CALSTER, *et al.*, 2021) and specific habitats (e.g., grasslands KLEYER *et al.*, 2008). Consequently, life-history trait databases suffer from a chronic problem of missing data on seed persistence in soil. We are simply uncertain about the survival potential of species in the soil in entire local and regional flora, which impedes, for example, extinction risk assessment studies (STÖCKLIN and FISCHER, 1999), research on community assembly (JIMÉNEZ-ALFARO *et al.*, 2016), and habitat restoration programs (HÖLZEL and OTTE, 2004).

Several characteristics determine seed persistence in the soil, including seed and whole-plant traits, vegetation and environmental properties, and various combinations thereof (POSCHLOD *et al.*, 2013; SAATKAMP *et al.*, 2014). To begin with, morphological (seed shape and seed size) and physiological traits (dormancy) have been widely used as predictors of seed persistence in soil: species with comparatively small, round, dormant seeds tend to build persistent and dense(r) banks in the soil (BEKKER *et al.*, 1998; GIORIA, PYŠEK, *et al.*, 2020; HONDA, 2008). Further, seed number (i.e., total seed production per individual plant), species population density and its dominance in the vegetation are considered important, especially for species that form SSBs, as species that produce a high number of seeds (often annuals; PHARTYAL *et al.*, 2020) and/or dominate in the vegetation canopy tend to have denser SSBs (KALIN ARROYO *et al.*, 1999; GIORIA, LE ROUX, *et al.*, 2019; HÖLZEL and OTTE, 2004).

Importantly, the predictive power of these characteristics varies strongly with local environmental conditions, suggesting that abiotic and biotic factors can mediate species' ability to form SSBs (ABEDI, BARTELHEIMER, and POSCHLOD, 2014; LONG *et al.*, 2015; ROSBAKH and POSCHLOD, 2021; SAATKAMP *et al.*, 2014). In general, species richness, composition, and density of the SSBs are positively correlated with conditions of unpredictable growth, frequent disturbance and high-risk recruitment (ANDERSON, SCHÜTZ, and RISCH, 2012; GIORIA, PYŠEK, *et al.*, 2020). Previous research on seed bank variation across successional gradients in different habitats indicates that

the persistence and size of SSBs decrease with successional maturity (GIORIA, PYŠEK, *et al.*, 2020; PLUE, VAN CALSTER, *et al.*, 2021; WARR, KENT, and THOMPSON, 1994). Additionally, a few existing studies on SSB variability along environmental gradients have revealed that all characteristics of SSBs, but particularly seed density, are negatively correlated with levels of abiotic stress, e.g., climate and edaphic conditions (FUNES *et al.*, 2003), due to their direct and indirect effects on seed persistence in the soil (FENNER, FENNER, THOMPSON, *et al.*, 2005; LONG *et al.*, 2015; POSCHLOD *et al.*, 2013; SAATKAMP *et al.*, 2014). Finally, the recent study by GIORIA, PYŠEK, *et al.* (2020) demonstrated that SSB type and density depend on species relatedness, suggesting that the ability to form persistent and/or dense SSBs might be inferred from phylogeny.

Such interconnected and variable relationships between predictors of SSB persistence (plant and seed traits, habitat preferences and phylogeny) make predicting SSB characteristics challenging. In particular, conventional statistical methods, such as regression models, are not suitable for this task as their learning, i.e., finding a relationship between the predictors (e.g., seed traits) and the response (SSB characteristics), is guided and constrained by a priori assumption (s) about the underlying relationships, thereby limiting their predictions with a pre-defined set of rules, which for SSB characteristics are currently only poorly understood. In this context, machine-learning (ML) is a promising tool for solving this problem.

Modern ML algorithms can flexibly identify the best predictors, non-linearities of predictors, and interactions between predictors, and usually achieve higher predictive performance than regression models (BREIMAN, 2001b; LECUN, BENGIO, and HINTON, 2015). Recent studies have demonstrated that ML models can successfully predict plant–environment relationships and outperform conventional methods, for example generalized linear models (GLMs), by a substantial margin (PICHLER, BOREUX, *et al.*, 2020). Moreover, ML models cope well with high-dimensional data. And yet, the ML approach has never been applied to infer and predict species persistence in SSBs. Finally, many potential predictors of SSB persistence have been identified in recent years (e.g., seed traits (LONG *et al.*, 2015) or phylogeny (GIORIA, PYŠEK, *et al.*, 2020)), but it remains unclear which characteristics contribute the most to predicting SSB persistence. This is an important question since collecting data on seed and plant traits, phylogeny, and environmental data results in different costs. ML could offer an attractive solution, as most likely multiple predictors that exist for SSB are difficult to identify with regression models, while ML is more flexible and efficient in detecting the most predictive patterns.

The main aim of this study is to test the applicability of the ML approach to infer and predict SSB properties in a regional flora. Specifically, we ask two questions: (1) can we predict species' abilities to build an SSB (and its density) better with ML than with commonly used (generalized) linear models? (2) What determinants of seed persistence in soil (environmental characteristics, seed traits and phylogenetic relatedness) are the most important for predicting qualitative and quantitative SSB characteristics using the ML approach? The practical utility of this approach in seed ecological research is demonstrated using an extensive SSB survey of a set of easily available seed and plant traits, and species phylogeny and environmental characteristics conducted in 18 species-rich grasslands located along a climatic gradient. The study is also intended to provide detailed explanations of the methods used to stimulate further usage of the ML approach in seed science research.

## 7.2 Material and Methods

### 7.2.1 Study system

The field data were collected from species-rich calcareous grasslands on nutrient-poor soil located along an elevational gradient in the Bavarian Alps (northern part of the Calcareous Alps, Germany; Figure S5.1) from 656 to 2363 m above sea level. We selected this study system for two main reasons. First, these ecosystems are ideal for studying the relative impacts of environmental (un)favorability on SSBs because the elevation gradient encompasses strong variation in climatic factors (temperature), soil conditions (soil moisture and nutrients), disturbance regimes (substrate stability, past and present land-use type) and many other environmental properties (KÖRNER, 2007) potentially affecting seed persistence in soil. Second, the relatively high number of taxonomically and functionally diverse species occurring in the studied grasslands allowed us to test the influence of plant phylogeny and seed traits on seed persistence in soil.

The study region is typical for the Northern Alps in Southern Germany, with steep Triassic lime and dolomite mountain peaks. The climate has mean annual precipitation rates up to 1500–2000 mm/year and a strong altitudinal decrease in mean annual temperature of ca.  $-0.6^{\circ}\text{C}/1000$  m of elevation (MARKE *et al.*, 2013). The lower montane vegetation is dominated by tall forbs and grasses, which are replaced by sedges, short-stature herbs and dwarf shrubs as altitude increases. During the first half of the 20th century, the traditional practice of grazing and mowing ceased, although several study sites were occasionally grazed by cattle or wild ungulates. The nomenclature follows OBERDORFER (1949).

### 7.2.2 Soil seed bank survey

In 2009, we selected 18 sites (Figure S5.1) located at different elevations representing different grassland vegetation types typical for the study region and easily accessible by foot for soil sample transportation. The SSBs were studied by cultivating the soil samples in an open greenhouse in Regensburg. More specifically, the soil samples were collected right after snowmelt: from the beginning of April to the second half of May in the years 2010–2017 (the sampling period is elevation-specific). The sampling period was spread over eight years due to limited space for soil sample cultivation. We assumed that the studied SSBs are rarely subject to considerable year-to-year fluctuations, as the disturbance levels in the study system are very low and succession rates are slow. Thus, it is most likely that sampling over different years did not affect the SSB characteristics. At each site, we randomly selected ten  $2\text{m}\cdot 2\text{m}$  plots (replicates) with homogeneous vegetation. The plots were located at more or less similar distances from each other within an area of ca.  $1000\text{ m}^2$  at each site. At each plot, soil was cored with a soil auger (4 cm diameter) to a maximum depth at 10 random locations and the samples were bulked together. The top layer of each soil core including the litter layer and the top centimeter was removed to exclude transient seeds present at the surface. We focused on the top 10 cm of the soil profile to account for elevation-specific differences in the sampled volume of soil, as lowland grasslands tend to have deeper soils as compared to their upland counterparts. A preliminary study conducted in a few lowland sites indicated that this approach would not affect the correctness of the SSB characteristics, as very few viable seeds were found below the first 10 cm of the soil profile (S. Rosbakh, unpublished data). Altogether, there were 100 soil samples from each site, resulting in 1800 samples in total.

The collected soil samples were transported to the lab, where they were stored at  $+4^{\circ}\text{C}$  for a few days before being processed. The soil samples were bulked by sieving through a 0.2-mm sieve,

spread thinly and evenly on plastic trays (40 cm wide) filled with potting soil, and cultivated outdoors at the University of Regensburg (Germany). To allow all viable seeds to germinate, the samples were cultivated for two successive growing seasons. Emerged seedlings were identified and removed from the trays. Five containers with potting soil only were used to control for contamination by airborne seeds or seeds present in the potting soil. After the initial flush of germination during the first cultivation year had ended, the soil samples were carefully turned over with a fork to facilitate the germination of ungerminated seeds. After cold stratification during the winter between two growing seasons, the soil samples were turned over one more time. Cultivation was discontinued when no more seedlings emerged for eight consecutive weeks.

### 7.2.3 SSB predictors

#### Environmental characteristics of the study sites

We considered three main types of SSB predictors: environmental factors, seed traits and phylogeny. Environmental predictors included thermal conditions, water and nutrient supply, and disturbance (grazing). Abundances of individual species in the vegetation at each site were included in the group of environmental predictors as they can be considered to be a result of abiotic filtering. The vegetation was surveyed in the same plots from which the soil samples were taken. The surveys were conducted in the same year that the soil was sampled in ten 2m · 2m plots per site at the peak of the growing season, which was elevation-specific. In each plot, the abundance of all vascular plant species was estimated based on the following scale: 0.1%–1%, 1%–5%, 5%–25%, 25%–50%, 50%–75%, and 75%–100%. The relative abundance of a species at a site was then calculated as the mean value of its abundance in all plots.

Site thermal conditions during the vegetation period were estimated with the help of the Landolt indicator value for temperature (Landolt's T), a proxy for mean soil and surface temperatures after snowmelt (LANDOLT *et al.*, 2010; SCHERRER and KÖRNER, 2011). Similarly, we used Landolt indicator values for water availability (Landolt's F) and soil nutrients (Landolt's N) as proxies for site water and nutrient supply during the vegetation period respectively. We opted for these indicator values because they are strongly correlated with directly measured temperature and soil parameters (e.g., air temperature, soil phosphorus content, soil depth; e.g., ROSBAKH and POSCHLOD, 2021) and due to their wider availability. Finally, grazing intensity, the main disturbance factor at the study sites, was recorded at all study sites and included three levels: (1) no current agricultural usage but occasional grazing by sheep and wild ungulates; (2) occasional extensive grazing by cows; and (3) mountain dairy farm with permanently grazing cows (except for site HO5, which was extensively grazed by sheep).

#### Phylogeny

To infer the influence of species' phylogenetic relatedness on seed persistence in soil, e.g., GIORIA, PŮŠEK, *et al.* (2020), we included information on the phylogenetic distances between study species as variables in the models. We made no inferences about the potential evolutionary processes underlying possible correlation between SSB properties and species phylogeny. The phylogenetic relationships among all the studied species were summarized by calculating eigenvectors extracted from a principal coordinates analysis (PCoA) representing the variation in the phylogenetic distances among species (PENONE *et al.*, 2014). We used the first 13 eigenvectors that represented more than 60% of the variation in the phylogenetic distances among species. The calculation of the eigenvectors was based on a dated phylogeny of a large European flora (DURKA and MICHALSKI, 2012).

## 7.2.4 Data analysis

All statistical calculations were done with the help of R software (version 4.1, R CORE TEAM, 2022).

### Data preparation

Based on the vegetation survey and soil cultivation data, we predicted the ability of a species to form a persistent SSB at a study site as a binary variable (1, able to form a seed bank; 0, otherwise). Furthermore, we predicted SSB density (seeds/m<sup>2</sup>), a quantitative measure of persistence in soil, for each species at each study site by adding up the numbers of seedlings germinated from the corresponding soil samples.

In the first step, we compiled a data set including SSB data (both the binary variable for ability to build a seed bank and seed bank density), environmental characteristics, seed traits, and phylogenetic relatedness for each species occurring at each study site both in the vegetation and in the SSB. In other words, the analyzed data set contained seed bank data for multiple species at the same plot, i.e., every row in the data set represented a species–plot combination.

We transformed the “seed dormancy” ordinal variable into a continuous variable. Missing values in the data set (seed shape for five species, productivity for 45 species and dormancy for 37 species) were imputed using the missRanger R package (an alternative implementation of the original proposed method (STEKHOVEN and BÜHLMANN, 2012)). As information about vegetation succession is rarely available in SSB research (many species from previous succession stages can survive in the soil for longer periods of time), the observations with vegetation equal to zero were removed from the data set. All predictors were standardized (centered and divided by their standard deviation) prior to analysis. Because SSB density was heavily skewed, we applied logarithmic transformation to it ( $\log(SSBdensity+0.001)$ ) and used the log-transformed variable as a response variable in our models. Model assumptions were met in all cases, when applicable.

### 7.2.5 Model evaluation

Evaluating models on the data on which they have been trained leads to underestimation of the actual predictive error for new data (ROBERTS *et al.*, 2017). To estimate the generalization ability of a model (i.e., how accurate the predictions of a model are for new observations), it has to be evaluated on a part of the data set that was not used for training the model, the so-called holdout. We used k-folded cross-validation (i.e., split the data set into several holdouts so that each data point appears once in the holdout data set, trained the model n times on the n training data sets, and averaged the predictive errors on the n holdout data sets; see ROBERTS *et al.*, 2017)

While cross-validation can produce accurate estimates of predictive performance, performance can still be overestimated if the observations are non-independent, for example in the presence of spatial auto-correlation (ROBERTS *et al.*, 2017). To counteract this, we used nine-folded, spatially blocked cross-validation to account for spatial dependencies introduced by the 18 sites from which the observations were collected. In each split, observations for 16 sites were used to train the model and the holdouts of two sites were used to estimate the predictive error. We used nine-folded blocked cross-validation for all the different sets of predictors.

For the calculation of the predictive error/performance (on the holdouts of the cross-validation), we used the area under the receiver operating characteristic (ROC) curve (AUC) for models when predicting the presence/absence of SSB, and the R-squared for models when predicting SSB density. AUC measures how well the model can differentiate between two response classes

(presence and absence of SSB). The AUC and R-squared were averaged over the nine holdouts of the cross-validation.

### 7.2.6 Performance of the ML and the conventional approach in predicting SSB characteristics

To test whether the ML approach is more advantageous than conventional approaches to predicting SSB characteristics, we used two common representatives of these groups. For ML, we used the random forest model (RF BREIMAN, 2001a) which has advantages over other ML models such as the low number of hyperparameters and the associated easier usability. Hyperparameters are parameters of the model itself (not to be confused with parameters that are optimized by the model), which are usually optimized in a trial-and-error search to find the optimal set for a specific data set (CLAESEN and DE MOOR, 2015). In addition, RF copes well with small data sets, can handle different types of responses (e.g., presence/absence of SSB and SSB density in our study), and is implemented in numerous programming languages. Foregoing the established procedures, we skipped hyperparameter optimization, opting instead to test the achievable predictive performance with the default hyperparameters because hyperparameter tuning usually requires expert knowledge. We used the RF implementation from the ranger R package (version 0.12.1; WRIGHT and ZIEGLER, 2017).

For the conventional statistical approach, we used linear regression (with log-transformed SSB density as the response variable) and logistic regression models (presence/absence of SB as the response variable) as these are commonly used tools in analyses of ecological data. The training or “learning” in regression models is specified by the hypothesis. Because linear regression models cannot learn outside of their hypothesis, i.e., if interactions are not specified the model cannot account for them, we added all the predictors additively, as well as all the combinations of predictor–predictor interactions. To compensate for the lack of power (interim results showed that the regression model would not converge with so many predictors), we applied elastic-net regularization (ZOU and HASTIE, 2005) via the glmnet R package (FRIEDMAN, HASTIE, and TIBSHIRANI, 2010). The strength of the regularization and the weighting between the l1 and the l2 regularization were tuned via three-fold cross-validation.

We used the mlr R package (version 0.9.0; LANG *et al.*, 2019) to train and evaluate the models.

### 7.2.7 Relative importance of environmental characteristics, seed traits and phylogeny in predicting SSB characteristics

We identified the relative importance of single predictors and corresponding functional groups (seed traits, environment and phylogeny) using the RF models as they demonstrated better performance than the regression framework (see below).

### 7.2.8 Identifying individual important predictors

RF provides quantitative information about the importance of the predictors. This ranking, called variable importance (BREIMAN, 2001a), should be not confused with regression coefficients in regression models, since the absolute values of those variables’ importance are uninformative and depend on the data set (and the number of predictors). However, the relative importance of the variables vs each other can be used to rank the predictors to identify the most predictive ones. Thus, to identify the most important predictors, we fitted RF on all the predictors and ranked the importance of the predictors based on their variable importance.

To assess the ability of the different functional groups to predict SSB, we divided the predictors into “Environment” (Landolt’s T, Landolt’s F, Landolt’s N, grazing intensity, cover), “Seed” (mass, shape, production, dormancy, endosperm presence/absence), “Phylogeny” (the first 13 phylogenetic eigenvectors), and “All” (all predictors). We then fitted the models (RF and regression) on the different groups and estimated the predictive performance via nine-folded blocked cross-validation (see above).

### 7.2.9 Minimal requirements for predicting SSB features with the ML approach

The choice of type and number of model predictors in ecological research strongly depends on the available data. Thus, to estimate the minimal set of SSB determinants required to achieve high predictive performance for SSB characteristics, we selected the four previously identified predictors (see corresponding sections) with the highest variable importance, which were (temperature, c5, c7, mass) for the presence/absence of SSB and (temperature, c7, mass, c5) for SSB density, to test their predictive performance.

In the second step, we first tested an RF model with only the first predictor (temperature) and in subsequent steps we sequentially added the remainder of the four predictors to the set of predictors. In each step, we estimated the predictive performance via nine-folded blocked cross-validation as described above.

### 7.2.10 Functional relationship of important predictors and SSB

Machine-learning models are often referred to as black-box models because it remains unknown what relationships the ML model learns in order to generate predictions. In linear regression models, the a priori hypothesis restricts the model’s learning, and the model is not capable of learning outside of this hypothesis (e.g., given two predictors A and B, if the interaction of A and B is not specified, the model cannot learn it). In ML, however, the idea is that the model should be capable of automatically identifying the best predictive patterns in the data (BREIMAN, 2001b), which makes ML a great tool for predictive modeling but comes with the cost of low interpretability (BREIMAN, 2001b). However, findings of discriminative ML models have driven the development of explainable artificial intelligence (xAI) methods and tools (BARREDO ARRIETA *et al.*, 2020). The idea of xAI is to reveal post-hoc the predictive patterns used by the ML model (BARREDO ARRIETA *et al.*, 2020; PICHLER, BOREUX, *et al.*, 2020; RYO *et al.*, 2021).

To check whether the predictive patterns the ML model used to predict SSB density are ecologically plausible, we used an approach based on accumulated local effect plots (ALE; APLEY, 2016) to explore the functional relationships between predictors (temperature, shape, and mass) and the response variable (presence/ absence of SSB, and density of SSB; MOLNAR, 2020). Briefly, ALEs are based on the idea of sampling predictors individually while keeping the other predictors fixed. If the sampled predictor is “important,” the predictions will be affected more strongly. Phylogenetic predictors were not considered because they cannot be linked to actual ecological mechanisms, making their interpretation pointless.

## 7.3 Results

### 7.3.1 Vegetation and soil seed bank surveys

At the 18 study sites, we recorded 290 species belonging to 45 families. The most dominant families were Asteraceae (45 species), Poaceae (30 species), Cyperaceae (21 species) and Caryophyllaceae



(16 species). Graminoids dominated in the vegetation of all the sites surveyed.

In total, 247,995 seedlings belonging to 162 species and 35 families germinated in the collected soil samples. Thus, germinable seeds of 128 species (e.g., *Campanula alpina*, *Ligusticum mutellina* and *Valeriana montana*) were not found in the collected soil samples. Of the species present in the SSB, seeds of 65 species, for example, *Carex flacca*, *Hypericum perforatum* and *Veronica officinalis*, were found at each site where the corresponding species occurred in the vegetation. Seeds of 97 species, such as *Alchemilla vulgaris*, *Nardus stricta*, and *Ranunculus montanus*, displayed a variable behavior in the surveyed SSBs, being present at some sites and absent from others. The seed density of species present in the SSB ranged from eight (*Carex sylvatica*, *Potentilla aurea*) to 63,603 (*Sagina saginoides*) with an average of 1617 seeds/m<sup>2</sup>.

### 7.3.2 Predictive performance of the ML and conventional approaches in predicting SSB characteristics

When comparing the performance of ML (RF) and conventional approaches (linear and generalized linear model) in predicting SSB characteristics, we found that the ML approach achieved an AUC of 86.1% and the GLM, an AUC of 76.9% when predicting the presence/ absence of SSB (Figure 7.1; intersection of the circles). In predicting the density of SSBs, the ML approach achieved an  $R^2$  of 41.7%, whereas the conventional approach (linear regression model) achieved an  $R^2$  of 18.7% (Figure 7.1; intersection of the circles).

For SSB density, the combination of environmental characteristics and phylogeny included in the RF model resulted in an  $R^2$  of 38.1%, and was followed by the combination of environmental characteristics and seed traits (of 33.2%), and seed traits and phylogeny ( $R^2$  of 32.5%). Among single groups of predictors, seed traits and phylogeny had the highest predictive performance with an  $R^2$  of 33.8% and 31.9% respectively. Environmental characteristics alone were predictive of only 6.2% of SSB density in the data set.

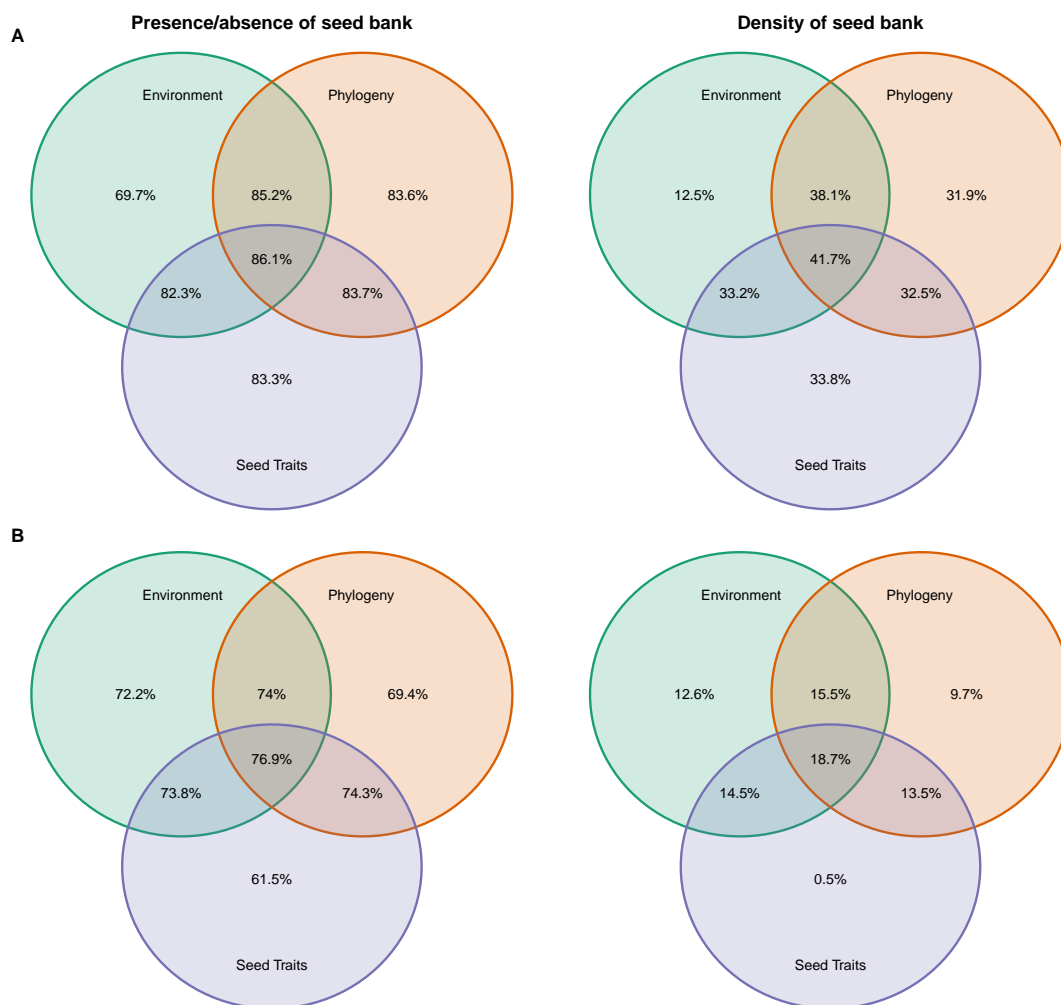
The conventional approach had a substantially lower predictive performance than the ML model, with an  $R^2$  of 18.7% when all predictors were used (Figure 7.1b), followed by the combination of phylogenetic and seed, and phylogenetic and environmental characteristics ( $R^2$  of 13.5% and 15.5%). Among single groups of predictors, seed characteristics showed the lowest predictive performance with an  $R^2$  of 0.5%, while phylogenetic and environmental characteristics achieved higher predictive performances with an  $R^2$  of 9.7% and 12.6%.

All the differences between the ML model and the (generalized) linear model were statistically significant excepting where only the group of environmental predictors was used (Tables S5.1 and S5.2).

### 7.3.3 Relative importance of environmental characteristics, seed traits and phylogeny in predicting SSB characteristics

#### Identifying individual important predictors

When looking at the variable importance of SSB predictors, we found that for both types of response (presence/absence (Figure 7.2a) and density (Figure 7.2b)) the temperature conditions at the surveyed sites were the most important predictor (9% and 10% for SSB presence/ absence and density respectively, Figure 7.2). The remaining environmental characteristics (soil nutrients and moisture, grazing intensity), several seed traits (seed mass, shape and production) and all the phylogenetic eigenvectors had comparable variable importance for SSB characteristics (3%–6%).

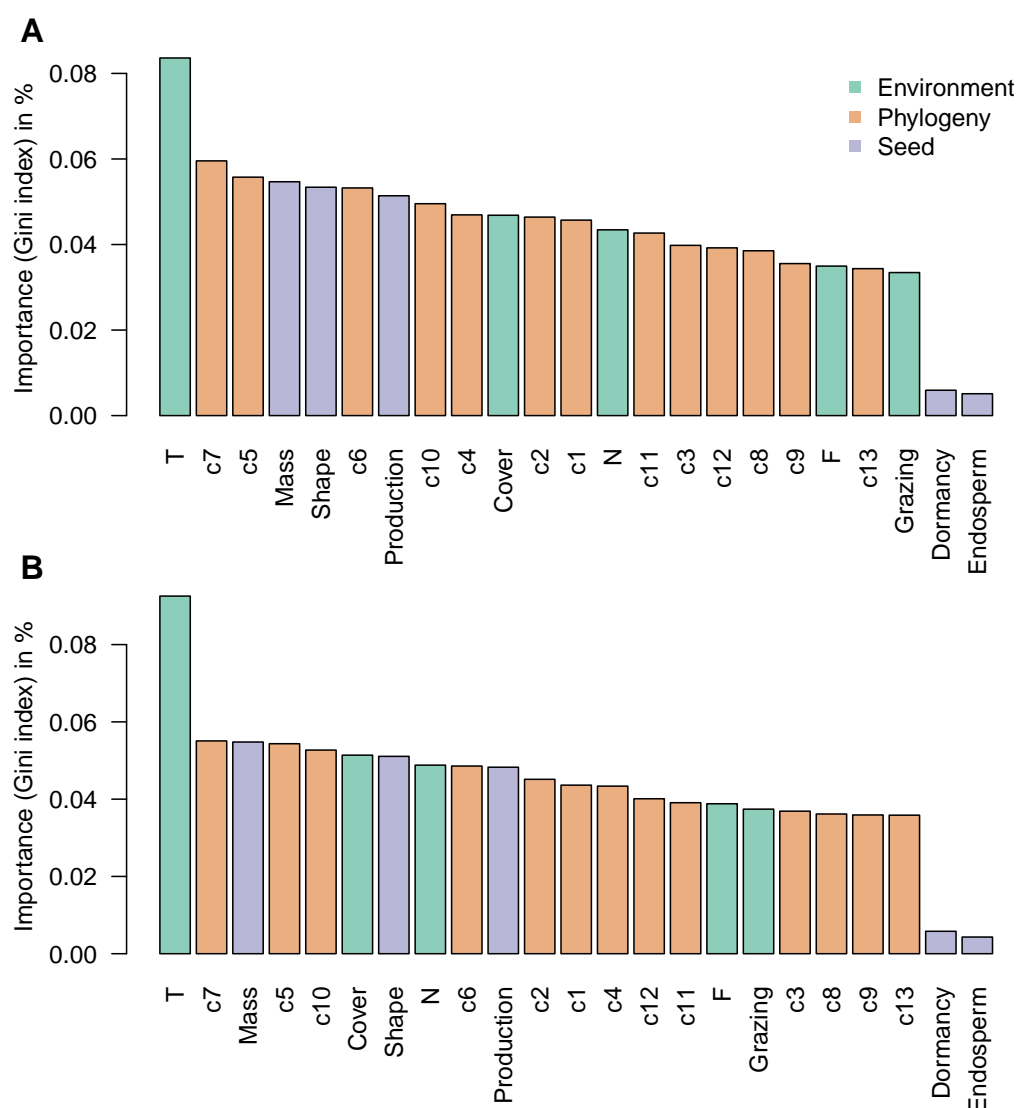


**FIGURE 7.1:** Performance of the random forest machine-learning model (a) and the conventional regression model (b) in predicting presence/ absence and density of seed banks. Both models were fitted on three sets of predictors (environment: temperature, nitrogen, moisture, grazing; seed traits: production, mass, endosperm, shape, and dormancy; phylogeny: phylogenetic axes that explain 60% of the variation). The intersections show the performance of the different combinations of predictors. Predictions for presence/absence of SSB (left column) were evaluated by AUC and predictions for SSB density (right column) were evaluated by  $R^2$ . Models were evaluated by blocked nine-folded cross-validation (observations were from 18 different plots; in each validation step 16 plots were used for training and two plots for validation)

Seed dormancy and endosperm presence/absence had comparatively low variable importance (<1%, Figure 7.2).

### Minimal requirements for predicting SSB features with the ML approach

We identified site temperature conditions as the predictor with the highest predictive performance (AUC of 65.8% and  $R^2$  of 6.2%, Figure 7.3) for both response types (Figure 7.3). For predicting SSB presence/ absence, the addition of the phylogenetic eigenvector c5 as a predictor is already sufficient to reach an AUC of 0.79, which corresponds to 91% of the maximal achievable predictive performance of 0.861 (Figure 7.1a). For predicting SSB density, two additional predictors, seed mass and the phylogenetic eigenvector c5, were necessary to reach 80% of the maximal achievable

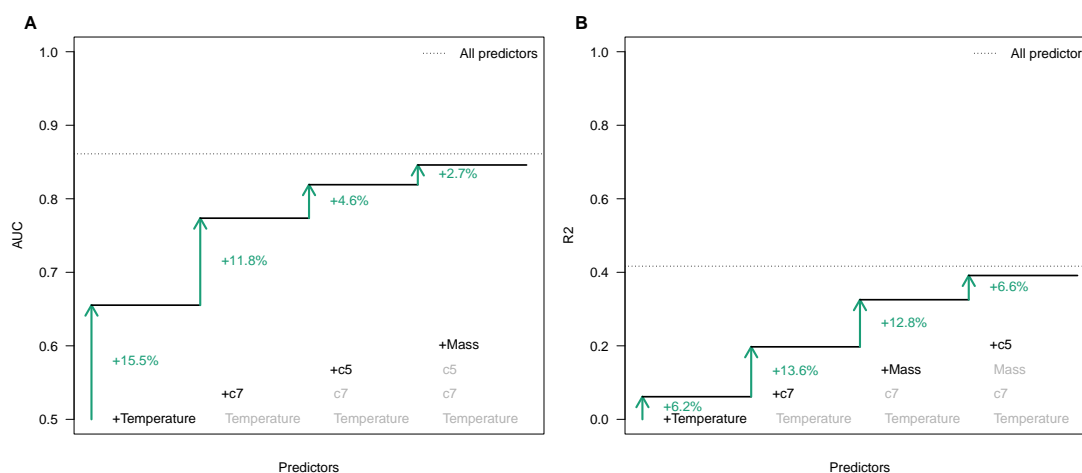


**FIGURE 7.2:** Variable importance in random forest model fitted on presence/ absence of seed banks (a) and on density of seed banks (b), plotted in descending order as per their relative importance measured by the Gini index in percent. All available predictors were used. Abbreviations: T, N, F: Landolt's indicator values for temperature, soil fertility and moisture respectively; c1–13: phylogenetic eigenvectors

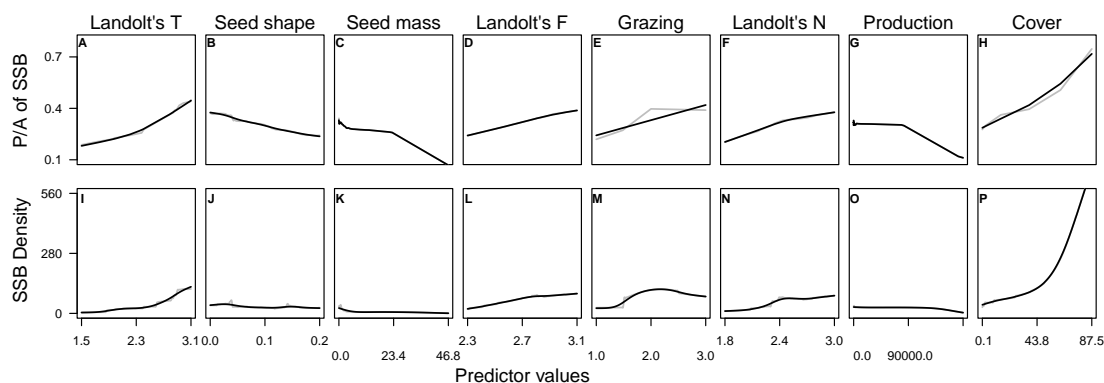
predictive performance (Figures 7.1b, 7.3b).

### 7.3.4 Functional relationship of important predictors and SSBs

According to the RF model, the probability of a species forming an SSB increased with increasing site temperature, soil moisture and fertility, and species abundance, and decreased with increasing seed mass, seed shape value, and seed production (Figure 7.4a–h). Species occurring at sites with relatively high grazing intensity tended to form persistent seed banks in the soil. SSB density was positively affected by site temperature conditions, soil moisture availability, and species abundance (Figure 7.4i–p).



**FIGURE 7.3:** Predictive performance of random forest models for soil seed bank presence/absence (a) and density (b) with the four most important predictors. Temperature, Landolt's indicator values for temperature; Mass, seed mass; c5 and c7, phylogenetic eigenvectors



**FIGURE 7.4:** Conditional dependency profiles (based on accumulated local effects) from random forest model for the environmental and the seed trait predictors. Predictors are sorted according to their variable importance found by the random forest model. (a–h) The profiles for predicting the presence/absence of seed bank formation; (i–p) the profiles for predicting seed bank density. The grey lines are the profiles, and the black lines are smoothing splines

## 7.4 Discussion

There is a growing demand for knowledge on soil seed persistence in both basic and applied plant ecological research. Information on species' ability to form persistent SSBs and their quantitative characteristics is not only crucial for understanding past, present and future plant population dynamics (SAATKAMP *et al.*, 2014; WALCK *et al.*, 2011), but also for restoration projects (HÖLZEL and OTTE, 2004), risk assessment (STÖCKLIN and FISCHER, 1999) and invasive-species management (GIORIA, LE ROUX, *et al.*, 2019). SSB surveys cannot by themselves satisfy such a need for knowledge as the field, and particularly the cultivation part of this approach, is still extremely resource-intensive.

Our study closes this gap by providing a novel methodological approach combining empirical knowledge on the determinants of SSB characteristics and a modern, flexible statistical approach based on ML. We first demonstrated that the ML approach substantially outperforms conventional

statistical methods in predicting SSB characteristics. Second, we found that SSB characteristics can be predicted with high accuracy regardless of the available predictor type (environmental characteristics, seed trait and phylogeny). Finally, we revealed that a few widely available SSB predictors can already achieve a high predictive power in the ML approach, suggesting high flexibility of the developed approach for use in various study systems.

#### 7.4.1 Predictive performance of the ML and conventional approaches in predicting SSB characteristics

In our study, the ML approach (RF) outperformed the (generalized) linear model considerably in predicting SSB characteristics (Figure 7.1). This finding confirms previous studies (e.g., PICHLER, BOREUX, *et al.*, 2020) and our expectations that given the complex nature of predictors and their interactions, accurate analysis of patterns in SSB, and more generally ecological data, requires a more flexible approach. The comparatively better performance of the ML approach in predicting SSB characteristics can be explained by two points.

First, assuming the correct functional form of the relationship between predictors and response variable is essential for accurate predictions. However, conventional statistical models such as linear regression models are constrained in their learning by a priori assumptions (the hypotheses) about the underlying system. Thus, the modeler needs to correctly specify the functional relationships between predictors and predicted characteristics (e.g., linear or non-linear) as well as the relationships between predictors. Moreover, it remains doubtful whether we can do the same for phylogenetic predictors, which can be seen as proxies for unmeasured traits (MORALES-CASTILLA *et al.*, 2015) and can improve the accuracy of predictions for ecological data (e.g. BROUSSEAU, GRAVEL, and HANDA, 2018a; POMERANZ *et al.*, 2019). However, making assumptions about their functional form or linking them to environmental or seed trait predictors is difficult as we cannot connect them to actual ecological mechanisms.

Second, conventional approaches often lack flexibility. As mentioned earlier, ecological patterns are usually scale-dependent (KÖNIG *et al.*, 2021; POISOT, STOUFFER, and GRAVEL, 2015) and nuisance predictors are required to account for locally varying functional forms, but entail loss of statistical power and give no guarantee that the possible fluctuating predictive patterns of predictors will be successfully captured. However, our results demonstrated that in this case ML could offer a powerful solution, as it is able to automatically identify and learn flexible predictive patterns (BREIMAN, 2001a; BREIMAN, 2001b).

Based on previous research, we assumed the existence of several predictive patterns for SSB characteristics. Our results indicate that RF individually achieved a high performance for phylogenetic and seed trait predictors (Figure 7.1), but their combination did not greatly further increase predictive performance. Assuming that phylogeny is a proxy for unmeasured traits correlated with SSB persistence, our findings thus confirm that phylogeny and information about seed traits encode similar predictive patterns for SSB characteristics and using both does not increase predictive performance greatly (Figure 7.1). On the other hand, for the conventional statistical models predictive performance increased greatly when all sets of functional predictors were used compared to use of individual groups (Figure 7.1). This implies that such statistical models cannot make the best use of the predictive patterns in the individual groups, indicating that some of the predictive patterns are non-linear and require higher flexibility. In contrast, the ML model was able to utilize the available individual predictive patterns, highlighting the advantages of the ML approach in predicting SSB characteristics when the availability of predictors is limited by temporal and/or financial resources.

#### 7.4.2 Relative importance of environmental characteristics, seed traits and phylogeny in predicting SSB characteristics

When predicting SSB characteristics with the ML approach, we found that the temperature conditions of the surveyed sites were the most important predictor, both for SSB presence/absence and density. The conditional dependency profiles of RF for this predictor revealed that species from warmer sites (i.e., higher Landolt's T values) were more likely to build up a persistent SSB with higher seed density (Figure 7.3). This finding is in line with our recent study in the same study system (ROSBAKH and POSCHLOD, 2021) and observations made elsewhere (ORTEGA, LEVASSOR, and PECO, 1997; WELLING, TOLVANEN, and LAINE, 2004) that the importance of SSBs for plant persistence gradually decreases with increasing elevation. Low-temperature stress in colder sites, including the short growth period with generally low temperatures coupled with frequent and severe frost events, negatively affects regeneration by seed. Therefore, because of the unpredictable seed input into the soil, plants shift their main persistence strategy from replacement of individuals by seeding germinating from the SSB to in situ maintenance of established individual plants by emphasizing stasis of adult stages (ROSBAKH and POSCHLOD, 2021).

The remaining environmental characteristics (soil nutrients and moisture, grazing intensity, and species abundance in the vegetation ("cover")) were found to be important predictors of SSB characteristics of equal importance, though with smaller predictive power than temperature. Although the ML approach does not allow for direct hypothesis testing and P-value calculations, which are usually used to confirm/reject postulated hypotheses, these findings are ecologically plausible as they agree well with previous SSB research. First, the detected low probability that species with persistent SSB and low SSB density would be present in sites with nutrient-poor and dry soils (i.e., lower Landolt's F and N values) is in line with the general observation that all components of SSBs, and particularly seed density (the curves for SSB density are much steeper than for SSB presence/absence; Figure 7.4), are negatively correlated with levels of abiotic stress (e.g., edaphic conditions; FUNES *et al.*, 2003), due to its direct and indirect effects on seed persistence in the soil FENNER, FENNER, THOMPSON, *et al.*, 2005; LONG *et al.*, 2015; POSCHLOD *et al.*, 2013; SAATKAMP *et al.*, 2014. Second, the revealed positive effects of grazing animals on SSB persistence and density agree well with the previous finding that frequent (moderate) disturbance favors formation of persistent SSBs with a high density due to the establishment of gaps by grazing and trampling, favoring species with a ruderal strategy (GRIME, 2006; RENNE and TRACY, 2007). Finally, SSB persistence and density were positively affected by plant abundance in the vegetation, a pattern known from other systems and explained by a comparatively large seed input into soils from the dominant species (SAATKAMP *et al.*, 2014). These results, however, come with the reservation that we did not check which predictor–predictor interactions were learned by RF. It is likely that RF found some, but the high predictive performances of a few single predictors (Figure 7.3) suggest that these are negligible.

In our study, three out of five seed traits: mass, shape and production, performed well in predicting SSB characteristics. Like in other SSB studies (BEKKER *et al.*, 1998; GIORIA, PÝŠEK, *et al.*, 2020; HONDA, 2008), the species in our system with comparatively small, round seeds, a seed morphology that favors easier seed burial and reduces risk of predation (FENNER, FENNER, THOMPSON, *et al.*, 2005), tended to build persistent and dense (r) banks in the soil. Seed production, a trait with predictive performance comparable to that of seed mass and shape, had a negative effect on SSB persistence and density, especially in species that produce more than 9000 seeds per ramet. This finding contradicts previous observations that high seed production is an important determinant of SSB characteristics due to the positive trade-off between number of produced seeds and their mass, i.e. productive species tend to produce smaller seeds that persist in the soil (SAATKAMP

*et al.*, 2014).

Seed dormancy and endosperm presence played a minor role in predicting SSB characteristics, as they showed the lowest predictive performance in the calculated models. The former finding agrees well with the studies by THOMPSON *et al.* (2003) and GIORIA, PÝŠEK, *et al.* (2020), which demonstrated that seed dormancy is an important mechanism promoting seed persistence in the soil but, overall, is a poor predictor of SSB characteristics. The weak predictive power of endosperm presence in inferring SSB characteristics supports the conclusion by LONG *et al.* (2015) that this trait, which can nevertheless serve as a good proxy for seed longevity in storage (PROBERT, DAWS, and HAY, 2009; TAUSCH *et al.*, 2019), does not reflect species' ability to persist in soil.

Including phylogenetic eigenvectors considerably improved RF model performance in predicting both SSB characteristics of interest. These results agree well with recent trait-based research showing that phylogenetic predictors contain information on unobserved traits, thereby increasing the predictive power of models (DESJARDINS-PROULX *et al.*, 2017; MORALES-CASTILLA *et al.*, 2015; POMERANZ *et al.*, 2019). In the SSB context, these unobserved (and usually hard-to-measure) traits might include a number of ecophysiological adaptations, such as desiccation tolerance and/or genetic degradation resistance, which positively influence inherent seed longevity and thus seed persistence in soil (LONG *et al.*, 2015). Alternatively, the good predictive performance of the phylogenetic predictors could be explained by their correlation with the seed traits correlated with SSB persistence (mass, shape productivity; Figures 7.2 and 7.4, Supporting Information S5 Figure S5.3), which are not randomly distributed across phylogeny (e.g. GIORIA, PÝŠEK, *et al.*, 2020). Although in our study it was not feasible to separate these two explanations from each other, we believe that in our case the latter explanation is more likely, as both the "Seed" and "Phylogeny" groups of predictors showed the highest predictive performance of the three groups but including both did not substantially improve predictive performance.

Besides testing different sets of predictors (environment, seed, and phylogeny), we also wanted to identify the minimal combination of the best predictors independently of their group. In our study, we considered 22 predictors, a comparatively large number that would entail high labor and temporal costs of data collection, especially in poorly studied regional flora. Our results indicate that both SSB components can be predicted with high accuracy based only on a few characteristics that can be obtained from already existing sources. For example, for studies conducted in Europe, information on site temperature conditions could be obtained from regional indicator values (e.g. LANDOLT *et al.*, 2010; TYLER *et al.*, 2021), data on seed shape and mass, from trait data bases (KLEYER *et al.*, 2008; LIU, COSSU, and DICKIE, 2019), and phylogenetic vectors, from the work by DURKA and MICHALSKI (2012). In other regions with poorer data coverage, global ready-to-use phylogenies (e.g. JIN and QIAN, 2019) in combination with in situ measurements of relatively simple seed morphological traits, such as mass and shape, could be used as reliable predictors of SSB characteristics.

**Data availability statements** The code and data used for the analysis are publicly available at <https://github.com/MaximilianPi/Rosbakh-Pichler-Poschlod-2021>.





---

## FIXED OR RANDOM? ON THE RELIABILITY OF MIXED-EFFECTS MODELS FOR A SMALL NUMBER OF LEVELS IN GROUPING VARIABLES

---

Johannes Oberpriller, Melina de Souza Leite, Maximilian Pichler

Published in *Ecology and Evolution*, 2020, 12, e9602, 10.1002/ece3.9062

**Abstract** Biological data are often intrinsically hierarchical (e.g., species from different genera, plants within different mountain regions) which made mixed-effects models a common analysis tool in ecology and evolution because they can account for the non-independence. Many questions around their practical applications are solved but one is still debated: Should we treat a grouping variable with a low number of levels as a random or fixed-effect? In such situations, the variance estimate of the random effect can be imprecise, but it is unknown if this affects statistical power and type I error rates of the fixed-effects of interest. Here, we analyzed the consequences of treating a grouping variable with 2–8 levels as fixed- or random-effect in correctly specified and alternative models (under- or overparametrized models). We calculated type I error rates and statistical power for all model specifications and quantified the influences of study design on these quantities. We found no influence of model choice on type I error rate and power on the population-level effect (slope) for random intercept only models. However, with varying intercepts and slopes in the data-generating process, using a random slope and intercept model, and switching to a fixed-effects model, in case of a singular fit, avoids overconfidence in the results. Additionally, the number and difference between levels strongly influences power and type I error. We conclude that inferring the correct random-effect structure is of great importance to obtain correct type I error rates. We encourage to start with a mixed-effects model independent of the number of levels in the grouping variable and switch to a fixed-effects model only in case of a singular fit. With these recommendations, we allow for more informative choices about study design and data analysis and make ecological inference with mixed-effects models more robust for small number of levels.

**Keywords:** Mixed-effects models, generalized linear models, multilevel models, hierarchical models, fixed effects, random effects

## 8.1 Introduction

Many biological data from experimental or observational studies have hierarchical grouping (or blocking, or clustering) structures that introduces dependencies among observations (McMAHON and DIEZ, 2007; BOLKER *et al.*, 2009; HARRISON *et al.*, 2018). A statistical analysis must account for these dependencies to ensure consistency of statistical properties (e.g., type I error rate) (ARNQVIST, 2020a), a task for which linear and generalized mixed-effects models (LMMs or GLMMs) were designed (LAIRD and WARE, 1982; CHEN and DUNSON, 2003). Mixed-effects models have replaced ANOVAs as the common tool for variance analysis (WAINWRIGHT, LEATHERDALE, and DUBIN, 2007; BOLKER *et al.*, 2009; BOISGONTIER and CHEVAL, 2016) because they allow simultaneous analysis of variance at different hierarchical levels (KRUEGER and TIAN, 2004; BOISGONTIER and CHEVAL, 2016), handle unbalanced study designs better (SWALLOW and MONAHAN, 1984; LINDSTROM and BATES, 1988; PINHEIRO and BATES, 1995; LITTELL, 2002), and have better statistical properties for missing data (BAAYEN, DAVIDSON, and BATES, 2008).

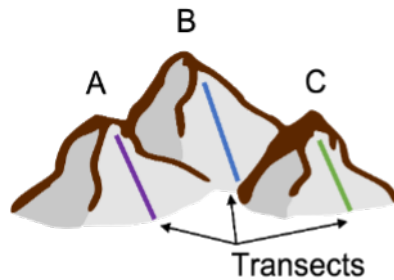
Mixed-effects models have the ability to adapt to different data structures, but the flexibility (8.1 WAINWRIGHT, LEATHERDALE, and DUBIN, 2007) that comes with them also leads to discussions about their challenging application (NAKAGAWA and SCHIELZETH, 2013; DIXON, 2016). This includes data-related properties such as the best way to handle overdispersion (HARRISON, 2015; HARRISON, 2014), small sample sizes in the individual blocks (GELMAN and HILL, 2007), technical aspects such as robustness to wrong distributional assumptions of the random effects (SCHIELZETH, DINGEMANSE, *et al.*, 2020), and to questions about how to compare different mixed-effects models (e.g. using  $R^2$  NAKAGAWA and SCHIELZETH, 2013). Additionally, there are application-oriented issues (HARRISON *et al.*, 2018; METEYARD and DAVIES, 2020) such as the question about the complexity of the random-effect structure (BARR *et al.*, 2013; but see MATUSCHEK *et al.*, 2017), the interpretation of random-effects (e.g. DIXON, 2016), or when a grouping variable should be treated as random or fixed-effect (HARRISON *et al.*, 2018).

A priori, modeling a grouping variable as fixed- or random-effect are for balanced study designs equally well suited for multilevel analysis (TOWNSEND *et al.*, 2013; KADANE, 2020). There are no strict rules because the best strategy generally depends on the goal of the analysis (GELMAN and HILL, 2007, Box 8.2), however, for unbalanced designs there are some subtleties. For instance, random-effect estimates incorporate between and within group information whereas the corresponding fixed-effects model (grouping variable is specified as a fixed-effect) only within group information which leads to different weighting of the individual level estimates (not in balanced study designs) (MCLEAN, SANDERS, and STROUP, 1991; DIXON, 2016; SHAVER, 2019; but see GIESSELMANN and SCHMIDT-CATRAN, 2020). This is important when one is interested in the actual level effects themselves (narrow-sense inference analysis), but also when only interested in the population-level effect (broad-sense inference analysis), i.e. where the individual levels of the grouping variable are not of interest and one uses a nonlinear model. For this type of analysis, for a fixed-effect model we cannot simply build the weighted average over the individual levels to obtain the population-level effect, because the nonlinearity does not commute with the expectation value.

The different inferential conclusions that result from fixed and random effect modeling are due to the different assumptions underlying these two options (MILLAR and ANDERSON, 2004). Modeling a grouping variable as random-effect implicitly assumes that the individual levels of the grouping variable are realizations of a common distribution, usually a normal distribution, for which the variance and the mean (the population-level effect) need to be estimated (e.g. DERSIMONIAN and LAIRD, 1986). As random effects are commonly parametrized so that the random-effect has a

**Box 8.1:** Scenario of an ecological study design with grouping/blocking variables

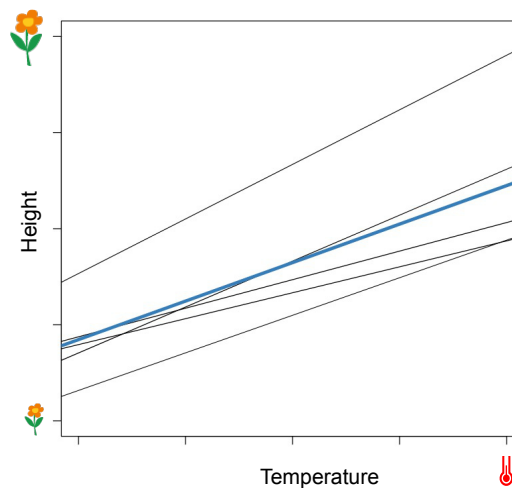
**Sampling design.** Suppose we want to understand the population-level effect of temperature on the height of a plant species that grows in different mountains. We hypothesize that higher temperature (lower altitude) increases the height of flowering plants. To do so, we establish altitudinal transects in many mountains and collect information from a certain number of plants. In this idealized scenario, we assume that the temperature predictor variable is collinear with altitude and not confounded with any other predictors like soil type, moisture, or pH.



**Problem.** The transects are not in the same geographical alignment, the type of soil varies in each mountain, and the plants are genetically very distinct among populations. All these factors introduce differences among populations that are not exactly of our interest (given our hypotheses), but statistically, plants of the same mountain are non-independent observations. The mountains can be considered as grouping, blocking or control variable.

**Modeling options.** We may use a mixed-effects model with a random intercept and slope (Box 8.2) for mountain to account for the differences among populations (grey lines in Figure 8.1) while still modeling the relationship of interest as fixed-effects (blueline). An alternative may be to use a fixed-effects model, i.e., to include mountain as a categorical predictor (Box 8.2).

**Hypothesis.** The height of flowering plants increases with temperature:



**FIGURE 8.1:** Individual realizations of the height dependence on temperature (grey lines) and the overall realization (blue line).

zero mean, this assumption shrinks the estimates of each random-effect level to zero. In contrast, treating a grouping variable as a fixed-effect makes no distributional assumptions about the individual level estimates (i.e., treating the levels separately of each other and thus no between level information is used to estimate the level effects). The random-effect model has fewer effective parameters than the fixed-effects model because of the shrinkage (e.g. GELMAN and HILL, 2007) which can lead in balanced designs to higher statistical power to detect significant population-level effects at the cost of higher computational and numeric demand (BOLKER *et al.*, 2009), discussions on how to correctly calculate p-values in unbalanced designs (BOLKER *et al.*, 2009, see NUGENT and KLEINMAN, 2021) and a bias towards zero of the random-effect estimates (JOHNSON, BARRY, *et al.*, 2015).

So, if we are not interested in each individual level effect (broad-sense inference), random effect modeling seems preferable over fixed-effects modeling. It is, however, unclear if these advantages remain when the number of levels in the grouping variable is small (cf. HARRISON *et al.*, 2018), because this might cause an imprecise and biased random-effects' variance estimate (HARRISON *et al.*, 2018), which then could influence the population-level effect estimate of the mixed-effects model (HOX, MOERBEEK, and SCHOOT, 2017).

The ecological literature suggests as a rule of thumb that an approximately precise estimate of the random-effect' variance requires at least five, sometimes eight, levels (BOLKER, 2015; HARRIS, 2015; HARRISON *et al.*, 2018). With four or fewer levels in the grouping variable, the preferred alternative is to include it as a fixed-effect (GELMAN and HILL, 2007; BOLKER *et al.*, 2009; BOLKER, 2015). But this threshold seems to be arbitrary chosen as it varies by discipline e.g. 10–20 in psychology (MCNEISH and STAPLETON, 2016) or 30–50 in sociology (MAAS and HOX, 2005). To our knowledge, however, none of these values were based on a systematic analysis of how the modeling choice of the grouping variable affects statistical properties such as the type I error rate and power of the estimated population-level effects (i.e., the weighted average slope or intercept over a grouping variable).

Here, we analyze a situation where an analyst wants to infer the population-level effect and decided to use a mixed-effects model but is confronted with a low number of levels in the grouping variable. For this scenario, we simulated an unbalanced study design on the height of a plant on a temperature gradient to compare empirical power and type I error with a varying number of levels (two to eight mountains). To represent the challenge of correctly specifying the model structure and the consequences if the structure is not correctly specified, we additionally tested mis-specified models (overparametrized or underparametrized versions of the fixed and mixed-effects models). To quantify the effect of these modeling choices on the population-level effect, we compared: type I error rates and statistical power. Based on our results and in the context of broad-sense inference, we give practical recommendations on when to include grouping variables as random-effect or as fixed-effect.

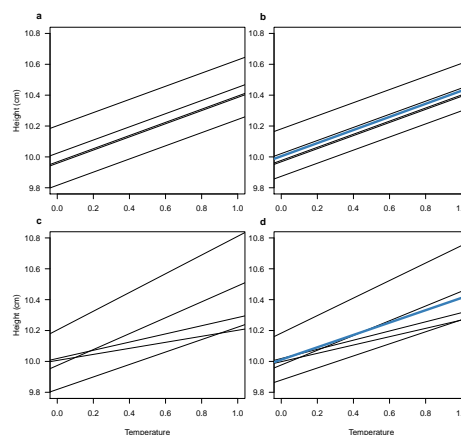
## 8.2 Methods

To compare random- and fixed-effects modeling of a grouping variable with small number of levels, we simulated data based on our hypothetical example from Box 8.1. We hypothesized, that higher temperatures increase the average height of plants. We simulated an unbalanced study design—a common scenario in ecology and evolution (SCHIELZETH, DINGEMANSE, *et al.*, 2020)—with two to eight mountains and a varying number of plants for each mountain (expected range between 40–360 plants per mountain) while keeping the overall number of plants constant (on average 200 plants per mountain) along altitudinal transects. For each case, we simulated

## Box 8.2: Modeling a grouping variable as random or fixed-effect

**Fixed or random effect?** The question of whether to include a grouping (blocking) variable as random or fixed-effect in the analysis depends on several factors. Fixed-effects are usually used when the analysts are interested in the individual level estimates of a grouping variable (BOLKER *et al.*, 2009) and these are independent, mutually exclusive, and completely observed (e.g., control and treatment in experiments, male and female when analyzing differences between sex) (e.g. HEDGES and VEVEA, 1998; GUNASEKARA *et al.*, 2014). Random-effects are modeling choices when the variance between the different levels (BOLKER *et al.*, 2009) and not the exact estimates of the different levels are of interest (e.g. DERSIMONIAN and LAIRD, 1986). Additionally, random-effects can be used when not every realization of the underlying mechanism can be observed (e.g., species across a number of observational sites in different geographic areas) but the analysts want to control for its influence (i.e., pseudo-replication, see ARNQVIST, 2020a). The two options differ in their interpretation, mixed-effects models use between- and within-group information whereas fixed-effects models use only within-group information. This subtle difference is important when for instance treatment or group differences are the goal of the analysis. Another important difference is that when modeling the categorical variable as fixed-effect conclusions apply to the levels used in the study, while when modeling as random-effect conclusions apply to the population of levels from where the studied levels were randomly sampled. However, in our example (Box 1), we are mainly interested in the population-level effect and not in the group differences which makes the inferential distinction negligible. See GELMAN (2005a) or GELMAN and HILL (2007) for more decision criteria for whether an effect is random or fixed.

**Technical differences between random and fixed-effects.** When specifying a grouping variable as fixed-effect, the model with a default contrast in R estimates the effect of one reference level (see SCHIELZETH, 2010) differences between the reference level and possible linear combinations of other levels (Figure 8.2.1a,c). Thus, it is not possible for fixed-effects models to estimate mean effect over groups (i.e., the population-level effect), but it can be calculated using e.g., bootstrapping (see Supporting Information S6), with sum-to-zero contrasts, or follow-on packages such as emmeans (LENTH, 2021). Mixed-effects models estimate the population-level effect and its variance and from a Bayesian perspective each individual level effect or from a frequentist perspective predict future realizations of the individual random-effect levels — Best Unbiased Linear Predictor (Figure 8.2.1b, d). Blocking variables may not only imply different intercepts (Figure 8.2.1 a, b), but also different slopes (Figure 8.2.1 c, d — the temperature “ecological” effect). In fixed-effects models, this is done by introducing an interaction between the population level effect and the grouping variable. With mixed-effects models the choice of modeling different slopes and their correlation to intercepts for each group is related to the study design and may have impact on modeling structure and inference. Such correlations between random slopes and random intercepts are fitted by default but can be disabled.



**FIGURE 8.2:** Fixed- and mixed-effects models fit to simulated data with random intercept (a,b) and random intercept and slope (c,d) for each mountain in the example from Box 8.1. Lines represent the individual estimates for each mountain. The blue line is the estimated population-level effect of mixed-effects models.

5000 datasets.

### 8.2.1 Scenario A – random intercepts per mountain

In scenario A, we assumed mountains only differ in their intercepts (mean height) and the effect of temperature (slope) is the same for each mountain (constant slope over the levels of the grouping variable, Table 8.1, Eq. M1). We tested two different mixed-effects model structures: a correctly specified model which corresponds to the data generating process (Table 8.1, Eq. M4) and an overparametrized model (Table 8.1, Eq. M5) with an additional random slope for each mountain. Since in real studies the true underlying data generating process is unknown, it is useful to understand if an overparametrized model correctly estimates the variances of the random effects to zero and predicts all random slope levels to zero (or nearly zero) and, thus, approximate the data generating process (Table 8.1, Eq. M1).

As fixed-effect alternatives, we tested the correctly specified model with mountain as fixed intercept together with temperature as slope (Table 8.1, Eq. M3), and an underparametrized model omitting mountain at all (Table 8.1, Eq. M2). This last model corresponds to a mixed-effects model that estimates the variances of the random effect to be zero and thus predicts the random effects to be zero.

**TABLE 8.1:** Data-generating and tested models for each scenario: Scenario A random intercept for each mountain and B random intercept and slope for each mountain. For the fixed-effects models, we used R syntax for model formula in `lm` function and for the mixed-effects models we used syntax from the `lmer` function from `lme4`. The response variable is height of flowering plants (H1, Box 8.1) and T is the temperature effect.

	Scenario A	Scenario B	Description
	Random intercept only	Random intercept and slope	
<b>Data-generating model</b>	(M1) $\text{Height} \sim T + (1 \text{mountain})$	(M6) $\text{Height} \sim T + (1 \text{mountain}) + (0 + T \text{mountain})$	Effect of intercept (and slope in B vary across mountains)
<b>Tested models</b>			
<b>Fixed-effect models</b>	(M2) $\text{Height} \sim T$	(M7) $\text{Height} \sim T$	Temperature only main effect – underparametrized model
	(M3) $\text{Height} \sim 0 + T + \text{mountain}$	(M8) $\text{Height} \sim 0 + \text{mountain} + T:\text{mountain}$	Main effects of temperature and mountain (and interaction in B) – slightly more complex model
<b>Mixed effect models</b>		(M9) $\text{Height} \sim T + (1 \text{mountain})$	Temperature and mountain both vary – underparametrized models
	(M4) $\text{Height} \sim T + (1 \text{mountain})$	(M10) $\text{Height} \sim T + (1 \text{mountain}) + (0 + T \text{mountain})$	Effect of intercept (and uncorrelated slope temperature in B) vary across mountains correctly specified models
	(M5) $\text{Height} \sim T + (1 \text{mountain}) + (0 + T \text{mountain})$	(M11) $\text{Height} \sim T + (T \text{mountain})$	Effect of intercept and slope temperature (correlated effects in B across mountains – overparametrized ) models

### 8.2.2 Scenario B – random intercepts and random slopes per mountain

In scenario B, we assumed the data generating process contained a random intercept and a random slope (without correlation among the random slopes and intercepts) for each mountain (Table 8.1, Eq. M6). Here, the population-level effect (temperature) differs among levels of the grouping variable (mountain). We tested three different mixed-effects model structures: a correctly specified model corresponding to the data generating process (Table 8.1, Eq. M10), an overparametrized model containing an extra term for the correlation of the random intercept and random slope (Table 8.1, Eq. M11), and an underparametrized model with only a random intercept for each mountain (Table 8.1, Eq. M9). We used the underparametrized model to test the effect of not accounting for important contributions to the data-generating process. Note, however,

only in case of balanced designs and linear models the population-level effect estimate from the underparametrized model is consistent with the full model, because of different weighting schemes (for unbalanced designs) and the fact that the expected value of a non-linear transformation of estimates is not the same as the non-linear transformation of the expected value of these estimates.

As fixed-effect alternatives, we tested the correctly specified model with the main effects of temperature, mountain and their interaction (Table 8.1, Eq. M 8), and the underparametrized model without mountain as predictor (Table 8.1, Eq. M 7). We tested the last model because mixed-effects models that estimate zero variance for both random-effects are virtually the same as fixed-effects models that omit the grouping variable.

### 8.2.3 Model fitting

We fitted linear mixed-effects models to our simulated data with the lme4 R package (BATES *et al.*, 2014) together with the lmerTest (KUZNETSOVA, BROCKHOFF, and CHRISTENSEN, 2017) package, which uses the Kenward-Rogers approximation to get the p-values of the fixed-effects. For fixed-effects models, we used the lmf function of the R stats package (R CORE TEAM, 2021). For fixed-effects models in scenario A, we extracted p-values from the summary function and, for scenario B, we used the fitted variance-covariance matrix and the individual level effects to bootstrap the population-level effect and its standard error (see Supporting Information S6).

Obtaining p-values for mixed-effects models is intensively discussed in the statistical community and they are only exact for simple designs and balanced data (KUZNETSOVA, BROCKHOFF, and CHRISTENSEN, 2017). One reason is that in order to calculate p-values in mixed-effects models denominator degrees of freedom must be calculated, which generally can only be approximated (KUZNETSOVA, BROCKHOFF, and CHRISTENSEN, 2017). For best practice in which situations one should use which approximation see (BOLKER *et al.*, 2009; see also (NUGENT and KLEINMAN, 2021)). The lmerTest package uses the Satterthwaite method to approximate the degree of freedoms of the fixed-effects in the linear mixed-effect model.

We used the restricted maximum likelihood estimator (REML) (for a comparison of REML and maximum likelihood estimator (MLE) see Supporting Information S6). All results of mixed-effects models presented in scenario A and B are for the datasets without singular fits (see section Variances of random-effects and singular fits). Technically, singular fits occur when at least one of the variances (diagonal elements) in the Cholesky decomposition of the variance-covariance matrix are exactly zero or correlations between different random-effects are estimated close to -1 or 1.

We repeated the analysis for the glmmTMB R-package because it uses a different implementation to estimate mixed-effect models, see Supporting Information S6 for methods and results.

### 8.2.4 Statistical properties and simulation setup

We used type I error rate and statistical power of the population-level effects (average height and temperature) to compare the modeling options. For example, type I error rate for the temperature (slope) is the probability to identify a temperature effect as statistically significant although the effect is zero. Statistical power in this case is the probability to detect the temperature effect as significant if the effect is truly greater than zero. For a correctly calibrated statistical test, the type I error is expected to be equal to the alpha-level (in our case 5%).

To investigate type I error rates of the models on the intercept (average height) and average

slope (temperature effect), we simulated data with no effects, i.e., the effects of temperature and mountain on height is zero. To additionally investigate statistical power, we simulated an example with a weak effect which corresponds to an average increase in size per unit step of the standardized temperature (linear scale) of 0.4 cm.

For scenarios A and B, the individual effects for each mountain were drawn from a normal distribution with variance of 0.01 and 0.25 around the average effects: 0.4 cm average height (intercept), and 0.4 cm average increase in size with temperature (slope). We chose to run and compare simulations with these two values for the variance of the random effects to understand better how a larger or smaller variance may interfere in type I and power.

### 8.2.5 Variances of random-effects and singular fits

To understand how the number of levels affected random-effects variance estimates, we compared the variance estimates for random intercepts and slopes from the correctly specified mixed-effects model in scenario B (Table 8.1, Eq. M10). We also compared optimization routines (REML and MLE) in terms of estimating zero variances (singular fits, see below) (see Supporting Information S6). For bounded optimizations, which most R packages apply for the variance, it has been shown that the null distribution of a random effect's variance is a combination of a point mass at zero and a chi-squared distribution (STRAM and LEE, 1994). For the sampling distribution with a true variance unequal to zero there are no proofs, but one would expect a similar distribution.

While singular fits do not signal a convergence issue, the consensus is that the results of such models are not reliable. However, we decided to use non-singular fits and additionally non-singular and singular fits combined for calculating power and type I error for the mixed-effects models and to infer the effect of singular fits on the averaged statistical properties. We classified a dataset as singular or non-singular if the mixed-effects model ran in lme4 reported a singular fit warning message. For fixed-effects models, we used estimates from non-singular and singular datasets combined.

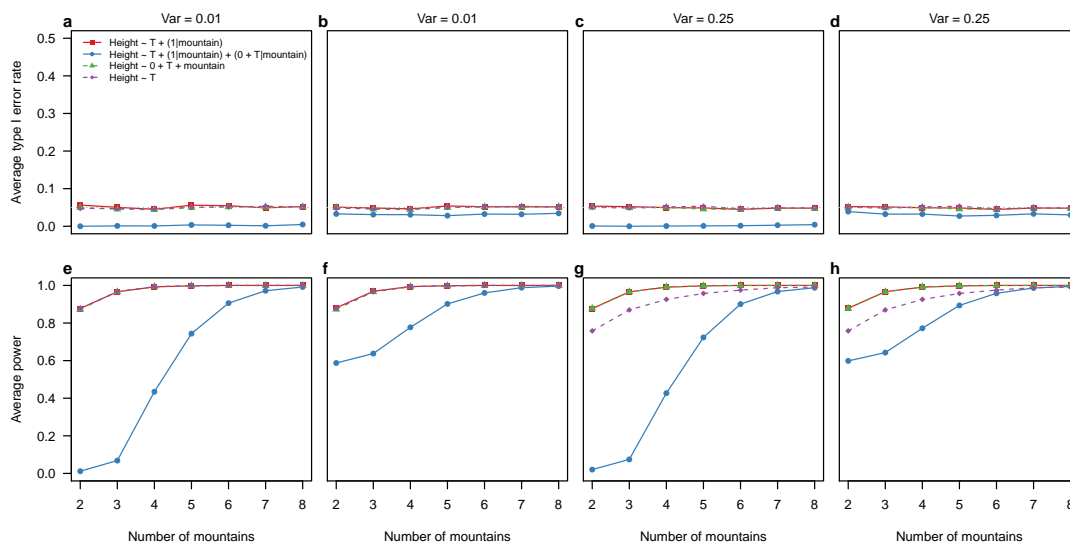
Using only non-singular fits for calculating power and type I error impacts these statistical properties (e.g., type I error) because they are conditional on this selection and thus likely not to be at the nominal level (e.g., 5% for type I error rate). However, as our main intention is to report the type I error rates from the point of the analyst who may adjust the model structure to dispose of the singular fit, our reported rates represent empirical type I error rates.

### 8.2.6 Quantifying the influences of study design on power and type I error

Power and type I error of the population-level effect may depend not only on the number of levels (mountains) but also on the random-effect variance, the overall number of observations and the balance of observations among levels. To further quantify the impact of these study design factors on statistical power and type I error rate of the population-level effect, we additionally ran 1,000 iterations (each with 1,000 non-singular model fits) with the data generating process from scenario B for our ecological example. Thereby, we sampled the number of mountains from 2 to 20 with equal probability for each number, the random-effects variances from  $10^{-4}$  to 4, the overall number of observations from 10 to 500 times the number of mountains. Additionally, to create different degrees of unbalance in data, we sampled for each mountain the average share of total observations from 0.1 to 0.9, which corresponds to at least 3 observations per mountain. We used the difference between the largest and the lowest proportion as proxy for the degree of unbalance.



For the so generated data, we fitted the correctly specified linear mixed-effects and fixed-effects models from scenario B (Table 8.1, Eq 8) and calculated type I error rate and statistical power of the population-level effect. We then fitted a quantile regression using the `qgam` R-package (FASIOLO *et al.*, 2020), with the statistical property (power and type I error rate) as response and variance, number of levels, total number of observations and the unbalance proxy as splines. We used a quantile regression with splines as we expect a non-linear relationship.



**FIGURE 8.3:** Average type I error rates and average power for linear fixed- and mixed-effects models fitted to simulated data with 2–8 mountains (random intercept for each mountain - Scenario A). For each scenario, 5,000 simulations and models were tested; (a, b, e, f) show results for simulated data with a variance of 0.01 in the random effects; (c, d, g, h) show results for simulated data with a variance of 0.25 in the random effects; (a, c, e, g) show results for mixed-effects models only from datasets in which mixed-effects models converged without presenting singular fit problems and (b, d, f, h) results for mixed-effects models for all datasets. Results for fixed-effects (a-h) model are from all datasets. (a-d) the dotted line represents the 5% alpha level

## 8.3 Results

### 8.3.1 Scenario A – random intercepts per mountain

When the effect of the temperature predictor was the same among mountains, irrespectively of the number of levels (mountains), all models except for the overparametrized model (random intercept and slope) showed an average type I error rate of 5% (Figure 8.3a–d). Average power increased (Figure 8.3e–h) with the number of mountains from 90% (2 mountains) to 100% (5–8 mountains). Note that the model omitting the grouping variable presented similar properties as the other models for small variances in the random effect. However, when increasing the variance of the random intercept in the simulation, the model omitting the grouping variable showed lower power (Figure 8.3g,h).

For the overparametrized model, we found, on average, a lower type I error rate of less than 5% (Figure 8.3a-d), and lower average statistical power to detect the temperature effect for a small number of mountains (Figure 8.3e-h). When combining singular and non-singular fits, the overparametrized model had more average power compared to only non-singular fits and an average type I error closer to the nominal level (Figure 8.3).

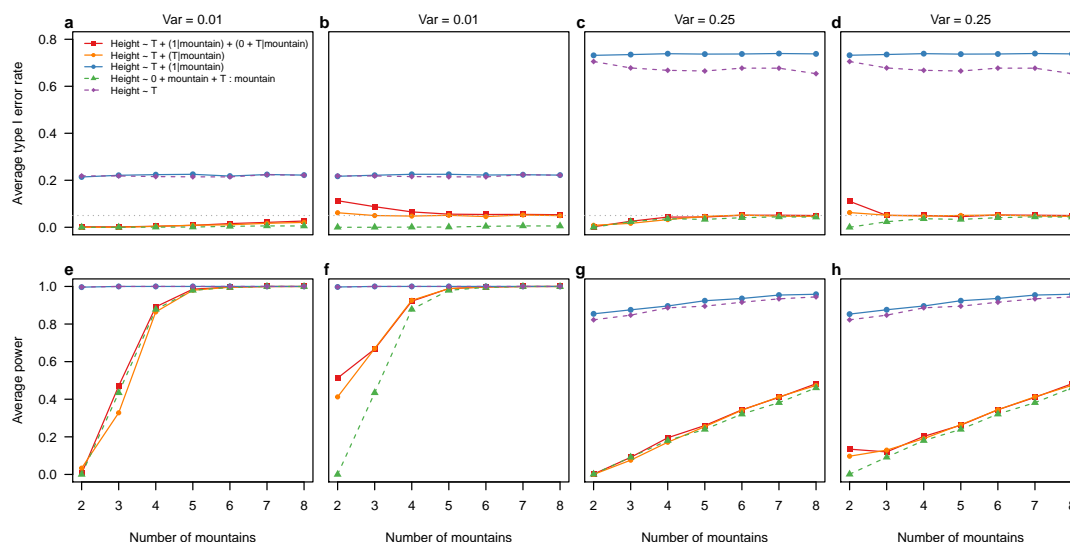
The results for the intercept for the different models (see Figure S6.9) are similar to the results for the slope in scenario B (see below).

### 8.3.2 Scenario B – random intercepts and slopes per mountain

In scenario B, where the effect of the temperature differed among levels, the modeling decision influenced the average power and average type I error (Figure 8.4). We found that average type I error rate of the correctly specified mixed-effects model (Table 8.1, Equation M10) slightly increased (Figure 8.4a) with the number of levels towards the nominal value (0.05) (Figure 8.4a). The increase was stronger for larger variances (0.25) in the random effects (Figure 8.4c). With singular fits, the mixed-effects models showed a higher average type I error rate than the nominal level for lower number of mountains (Figure 8.4b, d). With a higher variance in the random effects, the average type I error rate was only increased for two levels (Figure 8.4d). The overparametrized model with correlated random intercept and random slope (Table 8.1, Equation M11) presented similar properties, but with decreased average power (Figure 8.4e-h).

For the correctly specified fixed-effects model, average type I error ( $\approx 2\%$ ) stayed constant with the number of levels (Figure 8.4c) and a low variance in the random effects but increased stronger to the nominal level with a higher variance (Figure 8.4d). Average power increased with the number of mountains (Figure 8.4e-h). The mixed-effects model showed higher average power than the fixed-effects model irrespective of the number of mountains (Figure 8.4e-h).

The underparametrized model without the grouping variable had a higher average type I error rate (0.2) and higher average power than the other models (Figure 8.4e-h). With a higher variance, the average type I error rate was even higher (0.8; Figure 8.4c, d).



**FIGURE 8.4:** Average type I error rates and average power for linear (mixed-effect) models fitted to simulated data with 2–8 mountains for scenario B (random intercept and random slope for each mountain range). For each scenario, 5,000 simulations and models were tested; (a, b, e, f) show results for simulated data with a variance of 0.01 in the random effects; (c, d, g, h) show results for simulated data with a variance of 0.25 in the random effects; (a, c, e, g) show results for mixed-effects models only from datasets in which mixed-effects models converged without presenting singular fit problems and (b, d, f, h) results for mixed-effects models for all datasets. Results for fixed-effects (a-h) model are from all datasets. In (a-d) the dotted line represents the 5% alpha level

### 8.3.3 Variance estimates of random effects and singular fits

We found, for the models (singular and non-singular fit results combined) in Scenario B (random intercept and slope) that random-effects' variance estimates of the correctly specified model (Table 8.3, Equation M10) approximately distributed as a chi-squared distribution around the correct value (0.01) and a point mass at zero (Figure 8.5a,b median is near to zero). The point mass at zero decreased in height with increasing number of levels, i.e., less models estimated a variance of zero with an increasing number of mountains (Figure 8.5a,b, see also Table S6.1). There was smaller bias for the random intercept variance estimates than for the random slope variance estimates, which were still biased for eight levels. When looking at models without singular fits, the variance estimates were chi-squared distributed (Figure 8.5c,d). The bias towards larger values was stronger compared to estimates with singular fits, especially for the random slope estimates (Figure 8.5d).

By comparing the fitting algorithms, we found that using MLE led to more zero-variance estimates, i.e., singular fits, (Figures S6.3, S6.4) than REML. Additionally, using MLE, non-singular variance estimates were strongly biased (Figure S6.3, S6.4), but the bias decreases with increasing number of levels. As expected, for both optimization routines, increasing the number of levels reduced the number of singular fits (Table S6.1).

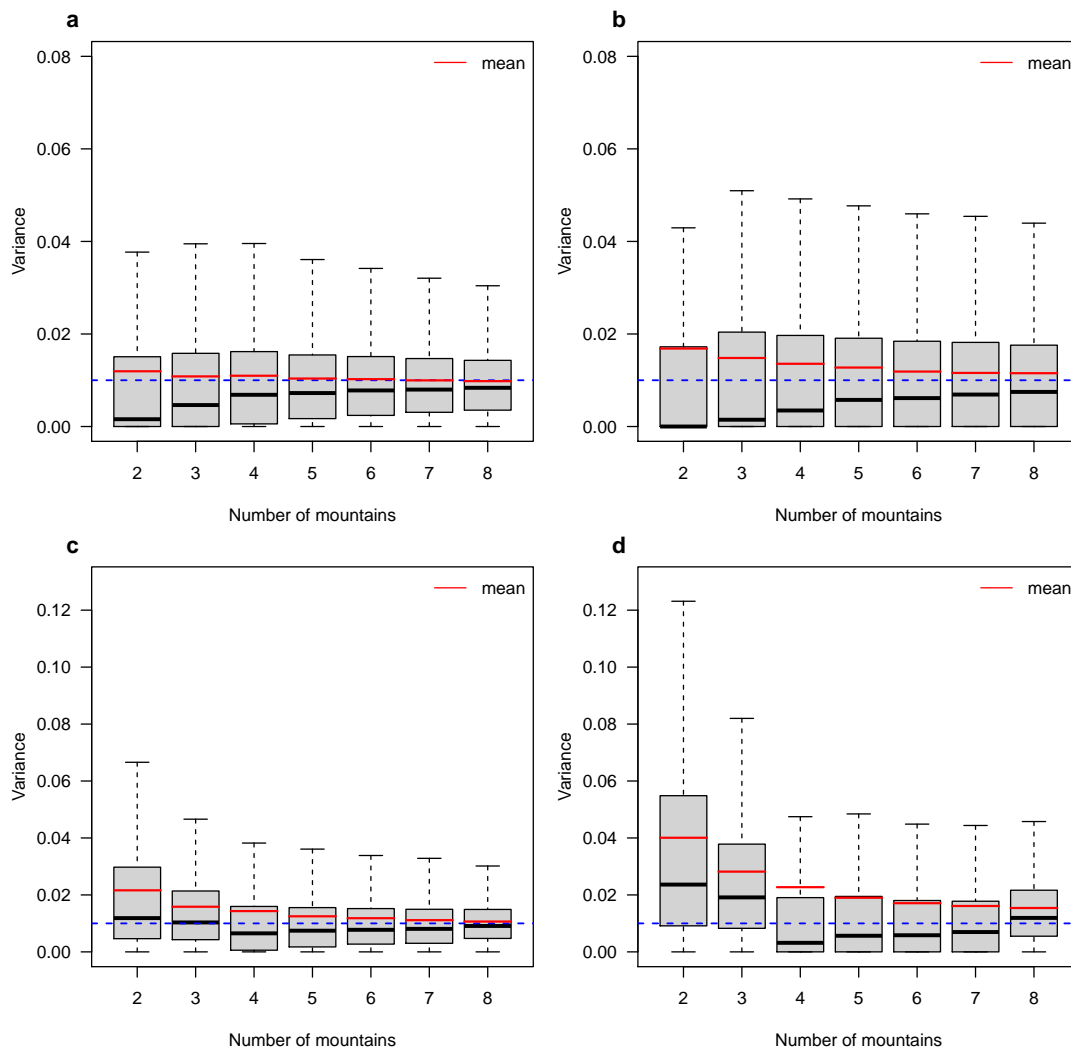
We found that singular fits led to different type I error rates and statistical power (Figure 8.6) in mixed-and fixed-effects models. For singular fits, the type I error rate of the correctly specified mixed-effects model was constant around 10% (like the model omitting the grouping variable), while with non-singular fits it was 1% for two levels and increased towards 3% with eight levels (Figure 4a). In comparison, the fixed-effects model had similar type I error rates (no distinction between singular and non-singular fits because fixed-effects models do not estimate the variance of the individual level estimates), both increasing from 0% (two levels) towards 1% (eight levels) (Figure 8.6c).

We also found differences in power for the mixed-effects models between singular and non-singular fits (Figure 8.6b, d). The power of the mixed-effects model with correct structure was higher for singular than non-singular fits especially for a low number of mountains (Figure 8.6b).

### 8.3.4 Quantifying the influences of study design on power and type I error

We found that the average type I error of mixed-effects models is slightly closer to the nominal value than its fixed-effect counterpart (Figure 8.7a). Additionally, we found that the number of levels most strongly influences the type I error rate for mixed-as well as fixed-effects model (Figure 8.7c). With five or more levels, however, the influence of the number of levels becomes negligible. Differences between the mixed-and fixed-effects models arose for the variance and the total number of observations. Here, the mixed-effects model was less influenced by a small random-effects variance and a low number of total observations than the fixed-effects model (Figure 8.7b,d). Balance, following our definition, (see Methods) did not influence the population-level effect in both models (Figure 8.7e).

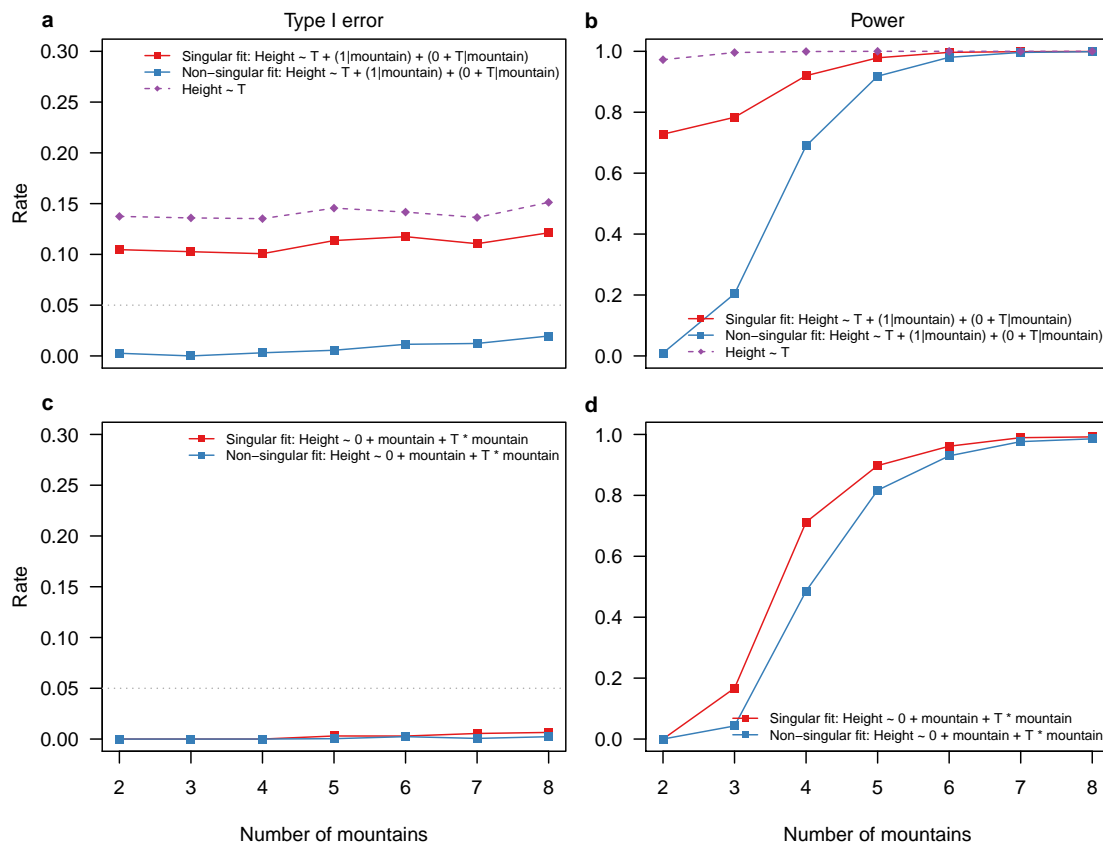
For power, we found no difference between a fixed-and mixed-effects model (Figure 8.7f-j). For both models, an increase in variance decreased the power, while increasing the number of levels increased the power (Figure 8.7g,i). The total number of observations and the balance between groups had less influence (Figure 8.7h,j).



**FIGURE 8.5:** Variance estimates of random intercepts (a, c) and random slopes (b, d) for linear mixed-effects models (LMM, Table 8.1, Equation M10) in Scenario B, fitted with lme4 using REML to simulated data with 2–8 mountains. Figures (a) and (b) show the results for all models (singular and non-singular fits) and figures (c) and (d) show the results for only non-singular fits. For each scenario, 5,000 simulations and models were tested. The blue dotted lines represent the true variance used in the simulation (0.01), and the red lines the average variance estimates

## 8.4 Discussion

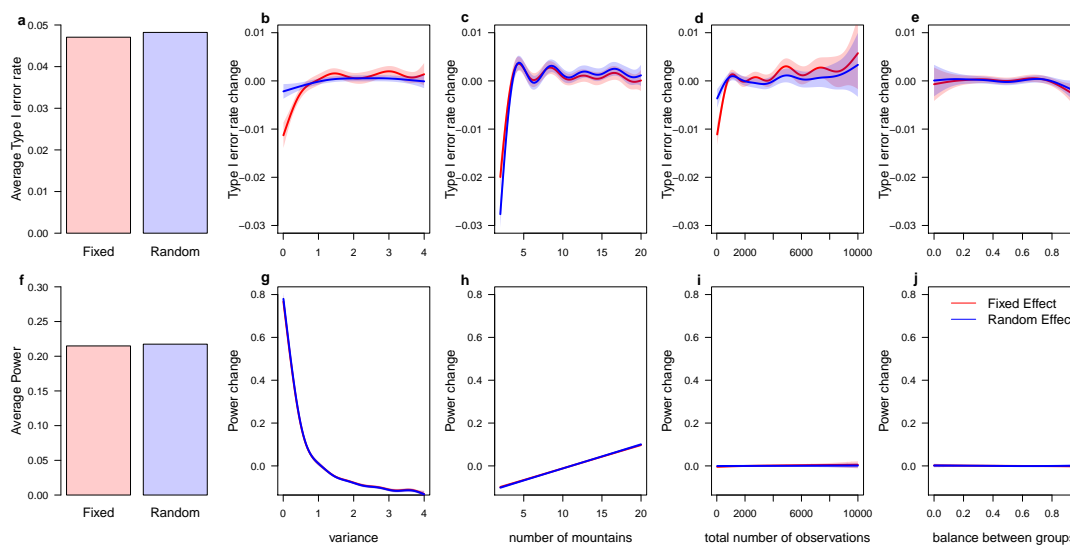
Ecological data collections or experiments produce data with grouping structures, and mixed-effects models can account for these dependencies. The main questions we explored in this article were: “should analysts stick to the mixed-effects model or fall back to a fixed-effects model, when the grouping variable has few levels?”, and “how does this decision influence statistical power and type I error rate of the population-level effect?” Here, we showed with simulations that mixed-effects models with a small number of levels in the grouping variable are technically robust (Figure 8.4), and that the decision between random and fixed effect matters most when the effect size of the ecological predictor variable differs among levels (Figure 8.4).



**FIGURE 8.6:** Type I error rate and power of the correctly specified linear fixed and mixed-effects models in scenario B. We separated the datasets based on if when fitted they presented a singular fit (red lines) or non-singular fit (blue lines) warning. Figure (a) and (b) are results for the linear mixed-effects models, and (c) and (d) for the linear fixed-effects models. For comparisons, we show also results for the fixed-effects model that omits the grouping variable (mountain)

When the effect of the ecological predictor is the same for each level of the grouping variable (scenario A, random intercept model), almost all models presented the same average power and average type I error (see also GOMES, 2021) (Figure 8.3a-d). The only exception was the overparametrized model that presented too low average type I errors and lower average power (Figure 1). We speculate that the model was unable to correctly predict the additional random effects to zero. Notably, for scenario A, the underparametrized model omitting the grouping variable presented correct average type I error rate (Figure 8.3a-d). However, this is illusive because average power decreased with increasing effect sizes of the random effects (Figure 8.3g, h). This confirms that the grouping variable needs to be included to correctly partition the variance among the different predictors (BELL, FAIRBROTHER, and JONES, 2019; GELMAN, 2005b; GELMAN and HILL, 2007). Also, including the grouping variable is mandatory if one is interested in the average intercept, otherwise it would cause inflated average type I error rates (see Figure S6.1; see the following section).

When the effect size of the ecological predictor differs for each level of the grouping variable (scenario B; random intercept, and random slope model), the average type I error and power were influenced by both model choice and the presence of singular fit warnings. The mixed-effects models had a better average type I error than the fixed-effects models, especially for a larger



**FIGURE 8.7:** Comparing the influence of study design factors on the type I error rate (b-e) and power (g-j) of linear mixed- (blue lines) and fixed-effects models (red lines) with their respective average values (a, f). We found that the variance of the random-effects and the number of levels (number of mountains) are the most important values to get correct type I error. For this analysis, we used the plant height example for Scenario B (random intercept and random slope). Results for mixed-effects models are only from datasets in which mixed-effects models converged without presenting singular fit problems, while results for fixed-effects model are from all datasets

number of mountains (Figure 8.4). Power was comparable between mixed- and fixed-effects models. But with non-singular and singular fits combined, the mixed-effects model had higher type I error rates and power than the fixed-effects models. In both cases, the mixed-effects models showed good type I error rates (about more or less than 5%) for a small number of levels.

Overparametrized mixed-effects models presented in both scenarios slightly lower average type I error and average power compared to the correctly parameterized mixed-effects model (Figures 8.3 and 8.4). This trade-off between type I error and power is in line with MATUSCHEK *et al.* (2017) for different model complexities. Overall, the overparametrized models are more conservative but have less power than the simplified models. We think these more conservative estimates are preferable over anti-conservative estimates, because some analysts tend to try a variety of analyses and only report significant ones (SIMMONS, NELSON, and SIMONSOHN, 2011), and more conservative average type I error counteract this procedure.

However, dropping the correlation structure between random effects should be carefully considered. It is possible that the type I error rate increases when no correlation in the model is assumed although there is one in the data-generating process. Group-mean centering of the population-level effect may mitigate the requirement of assuming a correlation, but it also changes the interpretation of the model because the individual levels are not referenced to the population-level effect anymore (they are now independent).

In scenario B, underparametrized models exhibited inflated type I errors (in line with SCHIELZETH and FORSTMEIER, 2009; BARR *et al.*, 2013; BELL, FAIRBROTHER, and JONES, 2019) but very high average power (Figure 8.4). We speculate that additional variance coming from the difference between levels in the grouping variable, which is not accounted, is attributed to the population-

level effect and causes overconfident estimates.

#### 8.4.1 Variances of random effects and singular fits

The rate of singular fits was very high for the small number of levels (Figure 8.5; Table S6.1). In our simulations, singular fits corresponded to zero variance estimates of the random effects. The resulting distribution of variance estimates consisted of a right skewed chi-squared distribution and a point mass at zero (many zeros corresponding to the singular fits) as expected (see STRAM and LEE, 1994). The variance estimates were biased and imprecise with a small number of levels, but the bias decreased with the number of levels towards zero (MCNEISH, 2017). Removing the singular fits led to even more bias in the variance estimates (Figure 8.5c,d).

The biased variance estimates are caused by ensuring positive variances in the optimization routines (BATES *et al.*, 2014; BROOKS, KRISTENSEN, BENTHEM, MAGNUSSON, BERG, NIELSEN, SKAUG, MÄCHLER, *et al.*, 2017). In case of a singular fit, the correctly specified mixed-effects model had similar power and type I error as a fixed-effects model dropping the grouping variable (Figure 8.6): no difference between the levels, which corresponds to a fixed-effects model without the grouping variable. However, the models still differed in their number of parameters (and degrees of freedom) which might explain the slight differences in power and type I error (Figure 8.6). When switching to fixed-effects models for singular fits in the random effect, the type I error rate and power were similar to the random-effect model with non-singular fits (Figure 8.6).

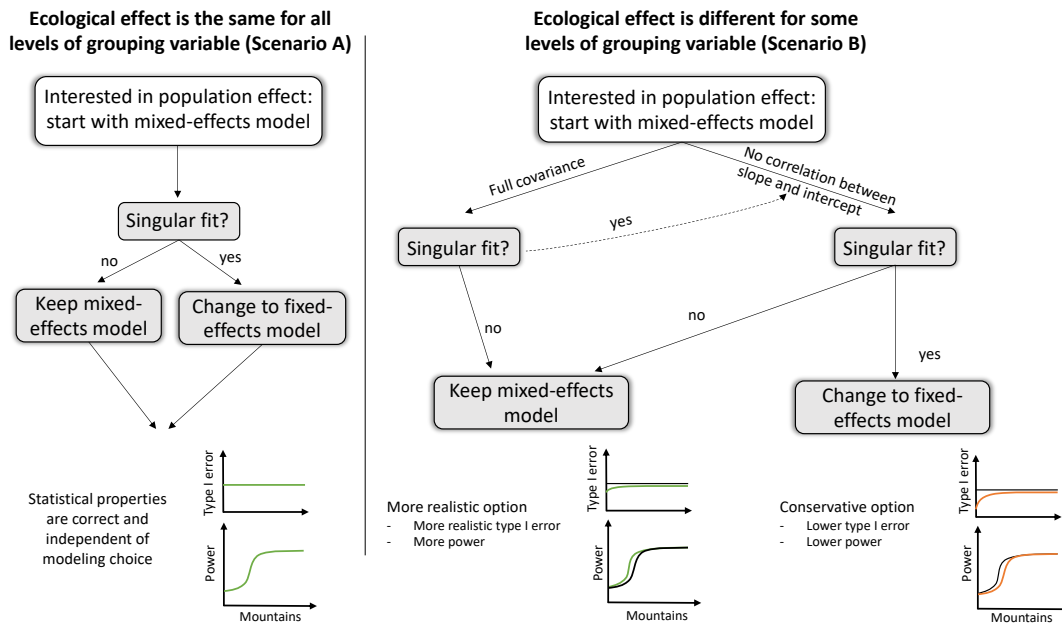
#### 8.4.2 Connection to study design

Earlier studies reported mixed recommendations about important study design factors. While some studies only stressed the importance of the total number of observations (MARTIN, NUSSEY, *et al.*, 2011; POL, 2012), we found, in accordance with AARTS *et al.* (2014), that the number of levels and the variance between levels have a strong influence on type I error rates and power. Due to our simulation design, which automatically increases the number of observations when increasing the number of levels, we however, cannot perfectly separate the effects of number of observations and levels from each other.

The influence of the variance on power and type I error is mixed. On the one hand, increasing the variance had a positive effect on the type I error for both models but the fixed-effects model was more strongly affected (Figure 8.7). The different distributional assumptions might explain this different behavior: the mixed-effects model assumes the levels to be normally distributed and estimates the variance of the levels' flexibly, whereas the fixed-effects model makes no distributional assumptions. We speculate that the mixed-effects model benefits from this informative distribution assumption in this edge case with less than five levels. On the other hand, increasing the variance over a certain value (Figure 8.7g) decreased the power of both models because more variance is explained by the difference between levels, and this increases the uncertainty of the slope effect estimate.

Given the strong influence of the number of mountains on type I error rates, we encourage to design a study with at least eight levels because with more than eight levels, the type I error rate was approximately not affected by the number of levels (Figure 8.7c). In our scenarios, the influence of the unbalanced number of observations between levels was small (Figure 8.7) confirming the robustness of mixed-effects to unbalanced data (PINHEIRO and BATES, 1995; SCHIELZETH, DINGEMANSE, *et al.*, 2020; SWALLOW and MONAHAN, 1984). However, if possible one should try to balance the groups because despite the robustness of mixed-effect models to an unbalanced design, it impacts the interpretation of the random effects and balanced studies

create the least problems regarding the model option (DIXON, 2016). Moreover, the impact of study design on type I error and power stresses the importance of pre-experiments and power analyses (e.g. BRYBAERT and STEVENS, 2018; GREEN and MACLEOD, 2016; JOHNSON, BARRY, *et al.*, 2015) to maximize the meaningfulness and efficiency of a study.



**FIGURE 8.8:** Consequences and recommendations for mixed-effects models with a small number of levels in the random effect. When the ecological effect (population-level effect) does not differ between different levels of the grouping variable (left side) all modeling options, which include the grouping variable, lead to the same results, and thus, only a singular fit requires a change to a fixed-effects model. If the ecological effect (population-level effect) differs among levels (middle to right side), starting with the mixed-effects model and only changing to the fixed-effects model in case of a singular fit is recommended

### 8.4.3 Practical suggestion

Before giving practical advice, we must recall the exact situation in which this manuscript acts. We assume that an analyst is interested in a population-level effect, and that they have already decided to use a mixed-effects model (broad-sense analysis, not interested in the individual levels effects), but faces a small number of levels, so that our recommendations only apply to such situations.

In this situation, the variance estimates of the random effects stabilizes in a reasonable manner with at least five levels in a grouping variable (Figure 8.4). With less than five levels, variance estimates are biased to zero (Figure 3) though without an effect on the observed average type I error rates of the population-level effect (Figures 8.3, 8.4). We rather found that the question of how to deal with a singular fit in the mixed-effects model is more crucial than the actual number of levels. If there is a singular fit warning, switching to the fixed-effects model leads to more conservative average type I error rates (Figure 8.4). Acknowledging that most singular fits occur with a small number of levels (Table S6.1), this might also explain the common rule of thumb to not fit a grouping variable as random effect if it has fewer than five levels (BOLKER, 2015; BOLKER *et al.*, 2009; GELMAN and HILL, 2007).

Our recommendations are summarized in Figure 8.8. We recommend starting with the mixed-



effects model, regardless of the number of levels, and switching to a fixed-effects model only in case of a singular fit warning. How to deal with singular fits is a topic of ongoing discussion. While BARR *et al.* (2013) states to start with the maximum model and simplify the model in case of convergence issues and singular fits, MATUSCHEK *et al.* (2017) suggests to think a priori about using simpler models because of higher power in return of increased type I error rate. However, we disagree with the view of (MATUSCHEK *et al.*, 2017) that trading a small increase in type I error rate for higher power is favorable, even though it could still be an interesting solution with the often-small number of observations in ecological studies, when the increase in power prevails upon the increase in type I error rate. We follow the position of BARR *et al.* (2013), and thus recommend starting with correlated random slope and intercept, when the population-level effect differs among levels. If obtaining a singular fit, switch to uncorrelated random-effects (following MATUSCHEK *et al.*, 2017), and in case of another singular fit, switch to a fixed-effects model.

Our recommendations assume that the random effect structure (e.g., random slope or not) is known a priori, which is often difficult in practice. Although model selection is theoretically possible for random effects (e.g., simulated (restricted) LRTs (WIENCIERZ, GREVEN, and KÜCHENHOFF, 2011)) or by residual checks (as facilitated by HARTIG, 2019), the frequentist point of view recommends sticking closely to the a priori-derived hypothesis, otherwise the risks such as they arise from multiple testing increase. Moreover, if the grouping variable was included as a confounder, this erroneous omission can cause a high type I error and wrong estimates. If there is uncertainty about the random-effect structure or concern about the statistical power, more time should be invested up front in hypothesis design and appropriate power analyses for mixed-effects models (e.g. BRYBAERT and STEVENS, 2018; GREEN and MACLEOD, 2016).

## 8.5 Conclusion

In conclusion, we showed that mixed-effects models are more robust than previously thought, despite the biased variance estimates for low number of levels in the grouping variable. We found that power and type I error of the population-level effect are robust against the model choice when the ecological effect is the same among the levels of the grouping variable, however, the model matters when the ecological effect differs among levels. When in doubt about the data-generating process, we encourage starting with a simplified model (random intercept only) and consult model diagnostics and simulated LRTs to check for evidence of random slope effects. When finding evidence for random slopes in these tests, we recommend starting with the mixed-effects model and switching only to a fixed-effects model in case of a singular fit problem. With this work, we provide a practical guideline, which helps analysts in the study design, the data analysis, and thus, making ecological inference more informative and robust.

**Acknowledgements** The idea of the manuscript originated from a discussion in the Theoretical Ecology seminar and was further developed in the Coding Club at University of Regensburg. We thank Rainer Spang, Carsten Dormann, Magdalena Mair, Björn Reineking, Sean McMahon, Andreas Ettner, and Florian Hartig for comments and discussions on earlier versions of the manuscript. We also thank three anonymous reviewers for their valuable comments and suggestions. JO and MP were funded by the Bavarian Ministry of Science and the Arts in the context of Bavarian Climate Research Network (bayklif). MSL was funded by the Smithsonian Predoctoral Fellowship. Open Access funding enabled and organized by Projekt DEAL.

**Data availability statements** No empirical data was used in this study. Code to run and analyze the experiments can be found at <https://zenodo.org/record/5817298#.YdRJ9VMxnRY>.



The goal of this thesis was to assess the potential of ML and DL algorithms for data analysis in E&E. To this end, we first introduced the concepts of ML and DL algorithms, reviewed their current state in E&E, and explored potential applications beyond prediction such as inference and for computational statistics. In the following, I will list the main results which are then discussed in the following.

**1. ML and DL algorithms adjust their complexity data-dependent which explains their superior predictive performance and attractiveness for inference:**

In chapter 2 we explain that ML and DL algorithms only started recently to gain popularity probably due to their unfamiliar algorithmic nature, their diversity and computational costs. The ability to automatically adjust complexity explains the superior predictive performance of ML and DL algorithms and make them also attractive for inference of complex patterns. Moreover, this further facilitated by new methods such as explainable AI.

**2. ML and DL algorithms improve the inference of complex ecological patterns:**

In chapter 3 we used ML and DL algorithms to improve predictions of species interactions in plant-pollinator networks. Moreover, we found that ML and DL algorithms, coupled with xAI tools, can be used to infer ecological patterns (or effects) such as trait-matching.

In chapter 5 we found that the reliability of inference with ML and DL algorithms depends on the choice of ML and DL algorithms, the hyperparameters, and the data. NN and BRT achieved the lowest bias under feature collinearity. Furthermore, we found that the optimal set of hyperparameters differs for prediction and inference.

**3. Statistical models can be made scalable using deep learning frameworks:**

In chapter 4 we showed that the DL framework PyTorch can be used to accelerate the approximation of the multivariate probit. Thus, we found that JSMD can be fit to community data with hundreds of species in reasonable runtime while still achieving a high accuracy in the estimated parameters.

## 9.1 Discussion of the results

### 9.1.1 Unraveling the opacity of ML and DL algorithms

In chapter 2 we clarified some of the opacity surrounding ML and DL algorithms. Unlike statistical models, ML and DL algorithms rely on algorithmic approaches to maximize their flexibility which in turn they control with their data-dependent complexity adjustments. It should be noted,

however, that we were not able to cover all aspects of ML and DL algorithms. For example, simple concepts such as the optimal DL architecture are still poorly understood. The history of DL vision models is characterized by different architectures, ranging from inception modules (SZEGEDY *et al.*, 2015), to residual skip networks (HE, ZHANG, *et al.*, 2015), to vision transformers (e.g., LIU, LIN, *et al.*, 2021), back to standard convolutional modules (LIU, MAO, *et al.*, 2022), suggesting that we are still far from understanding the fundamental principles that govern the effectiveness of DL architectures.

Given the superior predictive performance of ML and DL algorithms, we know that complexity adjustment works well in practice. By relying on internal and external complexity adjustment, ML and DL algorithms can transform models with high initial complexity into models with low effective complexity. One consequence of this is that, contrary to a common misconception, the data requirement or need for "big data" for ML and DL algorithms is not as stringent. Less data and more complex ML and DL algorithms only increase the risk of overfitting and probably require more thorough tuning.

We noted in chapter 2 that it is unclear why DL algorithms perform worse than classical ML algorithms on structured data. However, among others, I suspect a practical reason for this. While classical ML algorithms are available in user-friendly software solutions (R or Python packages, e.g. WRIGHT and ZIEGLER, 2017; CHEN and GUESTRIN, 2016), while fitting DL algorithms requires more expert knowledge. We have made an important contribution to fill this gap for fitting DNN with the 'cito' package (chapter 6) which provides a user-friendly interface and supports downstream techniques such as xAI tools. We have also started to implement approaches to support training and convergence of NNs, which is another bottleneck in training DNNs, to close the gap to classical ML algorithms.

### **9.1.2 ML and DL algorithms improves inference of complex ecological patterns**

With chapter 3 we made two important contributions to the field of E&E. From an ecological point of view, our results confirm that the trait-matching signal can indeed be a strong predictor of plant-pollinator interactions, supporting the idea of pollination syndromes (OLLERTON *et al.*, 2009; ROSAS-GUERRERO *et al.*, 2014a). However, much remains unclear. Studies have shown that plant-pollinator interactions vary not only on different spatial scales but also on different temporal scales (SCHWARZ *et al.*, 2021; POISOT, STOUFFER, and GRAVEL, 2015) suggesting that the randomness behind plant-pollinator interactions should not be underestimated. We partly explored this by also testing the predictive performance of species abundances without trait-matching (chapter 3). If abundances are available, species interactions can be corrected for them, as we did for the plant-hummingbird data but when species abundances are unknown, it is unclear whether the models can pick up the right, or rather a reliable (or causal, see chapter 5), signal for robust predictions.

From a technical point of view, we set a milestone by demonstrating that ML and DL algorithms can be used for inference. In particular we demonstrated that ML and DL algorithms, coupled with xAI, can achieve higher inference accuracy than statistical models for complex ecological patterns (chapter 3, chapter 7). In doing so, we confirmed that the mixed results on trait-matching from previous studies are due to inappropriate models, such as statistical models that lack flexibility (BROUSSEAU, GRAVEL, and HANDA, 2018b; POMERANZ *et al.*, 2019). One limitation, however, is that it is difficult to assess the validity of our inference at this point. The intraspecific traits in the plant-pollinator networks were most likely correlated, and as noted in chapter 5, collinearity between traits can affect algorithms such as RF and the Friedmans H-statistic and thus can decrease the

reliability of the estimated effects. However, we found mostly strong interspecific trait-matching signals where traits were uncorrelated, supporting that trait-matching occurs but not exactly where it is intraspecific.

One shortcoming of ML-based inference for variable-variable interactions is the Friedmans H-statistic we used in chapter 3. The Friedmans H-statistic metric is prone to collinearity and is computationally expensive. Therefore, in chapter 5 we explored a new xAI metric to infer variable interactions based on conditional mean effects with the advantages of being robust against collinearity and much faster to compute. While first results for the ACE were promising (Supporting Information S4), we also found that ML and DL algorithms seem to differ in their ability to model variable interactions (Figure S4.1). Future research should look more into this and also compare it with other ML and DL algorithm-based approaches to infer variable interactions (e.g. BEHR *et al.*, 2022).

It is often said that predictive and explanatory models are fundamentally different (e.g. SHMUELI, 2010). We could partly confirm this in chapter 5 since we found a difference between linear models (as a representative of explanatory models) and ML and DL algorithms. However, we also found that the optimal set of hyperparameters for ML and DL algorithms differ for prediction and inference. This means that the inferential properties of ML and DL algorithms can be improved. Moreover, the general difference between predictive and explanatory models regarding the model structure (choice of variables) is not fixed. For out-of-distribution prediction, explanatory and predictive models align in their model structure as the causality of the model structure becomes more important (chapter 5; PEARL and MACKENZIE, 2018). We can use this to indirectly improve the reliability of inference with ML and DL algorithms. As we are unable directly tune ML and DL algorithms for inference (because the true model is unknown), we can use out-of-distribution predictions in the tuning to force importance of a causal model structure. For example, we can do this by blocking the variable that creates the collinearity, known as blocked CV (ROBERTS *et al.*, 2017). We applied this in the chapter 3 which probably increased the reliability of inferred trait-matching effects. More fundamentally, this also argues for avoiding variable selection outside the tuning (e.g., using variance inflation factors) and letting the data decide the best model structure.

Finally, it should be noted that all models, including explanatory models, are subject to error tradeoffs and there is no true winner. Reliable inference means adjusting for all confounders and getting all functional forms right (PEARL, 2009). But when data is limited all models must accept tradeoffs. For example, if the number of observations exceeds the number of variables, the solution will be underdetermined leading to extremely high model uncertainty. ML and DL algorithms deal with this by using regularization and complexity adjustments (chapter 2) while statistical models often use model selection which also reduces variance at the cost of bias. So in the end, the more important question should be which error has higher costs.

### 9.1.3 Leveraging ML for computational statistics

In chapter 4 we found that DL frameworks can be used to make statistical models scalable. We used PyTorch for a faster approximation of the multivariate probit model used in joint species distribution models. Thus, the original MVP-based JSDM and its advantages such as high accuracy in estimates can be adapted to large community data such as novel community data (e.g. HARTIG, ABREGO, *et al.*, 2024) in a reasonable time, paving the way for new community analyses.

Ecologically, this is exciting because it facilitates analyses of large communities. New sampling methods such as sensors, remote sensing, and eDNA are capable of producing massive community

data (HARTIG, ABREGO, *et al.*, 2024). Whereas in the past the data was the limiting factor, this has now shifted to the side of community models. But with frameworks like sJSDM, we make it possible to analyze this high-dimensional data without having to make the trade-offs that other JSDM models based on latent variable models (LVMs) have. LVM models make JSDM scalable by using a low-rank approximation of the association matrix, which reduces runtime, but also makes the parameterization of the association matrix rigid (chapter 4), potentially leading to false-positive estimates. This is concerning when associations are a central part of the analyses, such as disentangling the internal structure of communities (spatial, environmental, and biotic associations) (CAI *et al.*, 2023; LEIBOLD, RUDOLPH, *et al.*, 2022).

On the other hand, the LVM model may have advantages over the MVP model. First, the MVP model does not correct niche estimates for biotic associations (POGGIATO *et al.*, 2021). The biotic associations only capture the variance and covariances not explained by the fixed effects. Thus, the choice of environmental variables affects the variance-covariance matrix, but not vice versa. In contrast, in the LVM model, the latent variables are on the scale of the linear predictors (similar to random effects) and can affect the fixed-effects and vice-versa (WARTON *et al.*, 2015), for example, if the latent variables are collinear with environmental variables (VAN EE, IVAN, and HOOTEN, 2022). Theoretically, latent variables could account for unobserved confounders which could improve predictions (because it improves causality), but studies found mixed results regarding a potential superior predictive performance of LVM (chapter 4, WILKINSON *et al.*, 2019).

Our approach in chapter 4 may be applicable to other statistical models, such as mixed effects models. In chapter 8, we found that we need at least 5 levels to reliably estimate random effects, but models with thousands of levels or with structured random effects also tend to scale poorly. Mixed-effects models are typically fit by integrating out their random effects (e.g. BATES *et al.*, 2014; BOOTH and HOBERT, 1999), which is similar to our MC approximation for the MVP (which can also be interpreted as integrating over potential random effects on species, see chapter 4). Thus, the MVP approximation of CHEN, XUE, and GOMES (2018) may be transferable to mixed-effects models, providing an alternative for estimating large mixed-effects models.

## 9.2 Conclusion and future research

Nature is complex. In chapter 3 and chapter 7, ML and DL algorithms predicted significantly better than classical statistical models. Thus, we conclude that the statistical models were too rigid for the patterns in the data, suggesting that nature is more complex we can assume with statistical models, and probably even more complex than we expect. With chapter 3 we stress the importance of having the "right" tools, not limited by flexibility or the analyst to derive the right model structure in terms of the functional form of the effects. The theoretically limitless flexibility of ML and DL algorithms, coupled with their complexity tuning, may meet these expectations, provided we can make this pipeline robust.

An important gap not addressed in this work is that xAI has been criticized for its arbitrariness (RUDIN, 2019), potentially leading to adversarial results (BORDT *et al.*, 2022). One reason is that xAI tools oversimplify complex, non-interpretable functions (ML and DL algorithms). While I agree that this is indeed a problem, alternatives such as statistical models would face the same challenge (e.g., trait-matching). I think that some of these problems can be attributed to our poor, technical, understanding of xAI methods with respect to inference (e.g., sensitivity of xAI methods to collinearity) and their lack of statistical robustness (e.g., confidence intervals). In fact, we already know how to make certain parts of this pipeline robust. For example, statistical properties such as confidence or prediction intervals can be approximated (e.g., via bootstrapping,

chapter 6 and conformal prediction, FONTANA, ZENI, and VANTINI, 2023). It is just that many parts this ML-based inference pipeline are not readily accessible. We need software solutions that combine all steps of the pipeline and are validated (e.g., correct coverage) such as modern mixed-effects model packages (e.g., glmmTMB, BROOKS, KRISTENSEN, BENTHEM, MAGNUSSON, BERG, NIELSEN, SKAUG, MACHLER, *et al.*, 2017). With the 'cito' package and a focus on inference, we have laid the groundwork for such a pipeline with DL algorithms. However, this is only a starting point, as we have not yet begun to address the big problems of statistical modeling such as non-independent observations (risk of pseudoreplication, ARNQVIST, 2020b) which can have serious consequences such as inflated Type I error rates.

Another way in which data analysis in E&E can benefit from ML and DL algorithms is by combining statistical models with ML and DL algorithms. The idea is to combine the best of both worlds, the data-driven approach of ML and DL algorithms and the knowledge-based approach of statistical models. An interesting example is presented by JOSEPH (2020b) who argues for combining hierarchical models with neural networks. We have also hinted at this idea by providing an option in sjSDM to account for space using DNNs (chapter 4), combining a linear model with a DNN. This could be generalized to ecological modeling, similar to generalized additive models (where linear effects can be combined with splines (WOOD, 2017)), with the advantage of flexibly accounting for confounding structures including variable interactions.

In conclusion, ML and DL algorithms certainly have the potential to become indispensable tools for analyzing structured data in the field of E&E. In this work, we have shown that ML and DL algorithms can infer complex patterns and make statistical models scalable, demonstrating their potential beyond simple prediction tasks. I strongly believe that ML and DL algorithms will play a similarly important role as statistical models for data analysis in E&E in the future. Particularly because ML and DL algorithms may not be an alternative but a necessity to cope with the complexity in nature and the increasing dimensionality of ecological data.





## Supporting Information S1 for Chapter 2

### 1 Trend analysis

For the global trend analysis in Figure 1, we used the R package ‘europepmc’ (v0.4.1, JAHN, 2021) to search from 1920 to 2021 the PubMed and Medline NLM databases. We used the following queries ‘deep learning’, (‘machine learning’ OR ‘machine-learning’), and (‘p value’ OR ‘p-value’ OR ‘statistically significant’) as representatives for Deep Learning, Machine Learning, and classical statistical approaches. The number of hits were normalized by total hits in each year. For the stream charts in Figure 2.4, we used the search queries Table S1.1 and added them to the query ‘AND (“ecology” OR "ecolog\*" OR "evolution")’ to restrict the queries to hits from the ecology and evolution field.

**TABLE S1.1:** Search queries and their corresponding ML and DL algorithm.

Queries	ML and DL algorithm
(“artificial neural network” OR “deep neural network” OR “multi-layer perceptron” OR “fully connected neural network”)	Deep neural network (ANN)
(“convolutional neural network” OR “object detection”)	Convolutional neural network (CNN)
(“recurrent neural network”)	Recurrent neural network (RNN)
(“graph neural network” OR “graph convolutional”)	Graph neural network (GNN)
(“random forest”)	Random Forest (RF)
(“boosted regression tree” OR “boosted reg” OR “gradient boosting” OR “adaboost”)	Boosted Regression Trees (BRT)
(“k-nearest-neighbor”)	k-nearest neighbor (kNN)
(“ridge regression” OR “lasso regression” OR “elastic-net” OR “elastic net”)	Ridge, lasso, or elastic-net regression
(“support vector machine” OR “support vector”)	Support vector machine (SVM)

For the word clouds in Table 1, we used again the ‘europepmc’ R package to search abstracts and titles within the ML and DL algorithm specific queries (Table S1.1) for the following ecological keywords: species distribution, species interaction, mortality, remote sensing, invasive, decision

making, ecosystem, species identification, species detection, extinction, functional trait, ecological network, biodiversity, and camera trap.

We used the R packages ‘tm’ (FEINERER, HORNIK, and MEYER, 2008), ‘wordcloud’ (Fellows (2018)), and ‘wordcloud2’ (Lang and Chien (2018)) to analyze and create the final word cloud plots.

## 2 Algorithms

Ideas and code for all algorithms discussed here are available at <https://maximilianpi.github.io/Pichler-and-Hartig-2022/>.

## Supporting Information S2 for Chapter 3

### 1 Plant-pollinator simulation

In our simulated plant-pollinator the trait-trait interactions (trait-matching) result from a Gaussian niche. To ensure that fraction of two traits (e.g.  $A1 = 0.5$  for species  $i$  and  $B2 = 0.7$  for species  $j$ ) has the same distance to zero from both sides of the mean, we used the logarithmic fraction:  $\log(0.5/0.7) \approx 0.34$  and  $\log(0.7/0.5) \approx -0.34$ . The maximum for this fraction is  $A1$  equal  $B1$ :  $\log(0.5/0.5) = 0$  and obtains the highest interaction probability of the Gaussian niche ( $\mu = 0$ ).

The difference between logarithmic and no logarithmic is shown in Fig. S2.1. For no logarithmic fraction, equal absolute distances for the optimal fraction  $A1$  equal  $B1$  cannot be guaranteed.

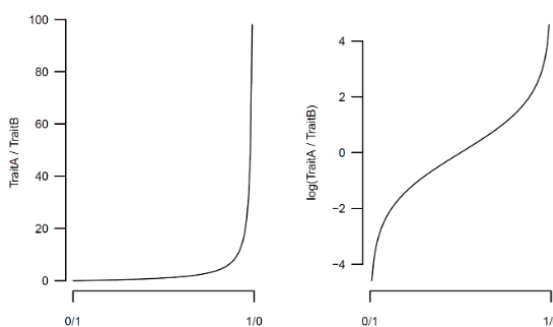


FIGURE S2.1: Unlogged and logged fraction of two trait-trait matches.

### 2 Plant-pollinator databases

The global plant-pollinator database is available on: <https://doi.org/10.6084/m9.figshare.9980471.v1>

TABLE S2.1: Detailed information on plant traits for the plant-pollinator database

Trait	Type	Levels	Additional information
Type	Discrete	arboreous, herbaceous	
Flower season	Discrete	sprism, summer, spriaut, spring, autspri, sumspri, autumn, year, sumaut, wispring, winter	Describes the seasonal range. For instance, sprism correspond to spring - summer range.
Flower diameter	Continuous [mm]		
Flower corolla	Discrete	campanulate open, tubular	
Nectar	Discrete	Yes, No	Whether flower contains nectar or not.
Flower color	Discrete	white, yellow, purple, pink, green, blue, red	
Bloom system	Discrete	insects, insects/bats, insects/bats, insects/birds	Type of pollinator
Self-pollination	Discrete	Yes, No	
Inflorescence	Discrete	solitary, solitary/clusters, solitary/pairs, yes	
Composite	Discrete	Yes, No	

TABLE S2.2: Detailed information on pollinator traits for the plant-pollinator database

Trait	Type	levels
Guild	Discrete	andrenidae, bumblebees, butterflies, coleoptera, cuckoo bees, flies, honey bees, moths, other, other bees, stingless bees, sweat bees, syrphids, wasps
Tongue	Continuous [mm]	
Body	Continuous [mm]	
Sociality	Discrete	Yes, No
Feeding	Discrete	oligolectic, parasitic, polylectic

### 3 Model Training

#### Comparison of predictive performance under different network characteristics

We standardized the features before fitting. We fitted models' hyper-parameter in 30 random tuning steps. We used nested cross-validation, five times outer and three times inner, to maximize generalization and to estimate overfitting. We applied cross-validation by putting an insect with all its possible plant interaction partners out. We regularized deep neural network (DNN) and convolutional neural network (CNN) with dropout ( $rate = 0.2$ ) for presence-absence plant-pollinator interactions and for plant-pollinator interaction counts with batch normalization (interim results showed that batch normalization worked better for plant-pollinator interaction counts).

**TABLE S2.3:** Overview of hyper-parameters we tuned in the predictive performance comparison.

Model	Hyper-parameter	Range	R package
RF	mtry	2 - (number of features-1)	randomForest (Liaw and Wiener, 2002), ranger Wright and Ziegler, 2017)
	nodesize	2 - 50	
	replace	Yes/No	
	learning rate	0.1 - 0.0001	keras (Chollet et al. 2017), tensorflow (Abadi et al. 2015)
	hidden nodes	5 - 50	
DNN	number of layers	1 - 5	
	bias	Yes/No	
	optimizer	Sgd, adam, rmsprop	
	decay (optimizer decay)	0.9 - 0.99	
	number of layers	1 - 6	
	learning rate	0.1 - 0.0001	keras (Chollet et al. 2017), tensorflow (Abadi et al. 2015)
	hidden nodes in fc layer	10 - 80	
CNN	number of kernels (filter)	8 - 30	
	pooling	max/average	
	decay (optimizer decay)	0.9 - 0.99	
	booster	gbtree, dart	Xgboost (Chen and Guestrin 2016)
	sample type	uniform/weighted	
	normalize type	tree/forest	
	eta	0.01 - 0.5	
	max depth	1 - 10	
BRT	lambda	0.1 - 10	
	alpha	$2^{-10}$ - $2^5$	
	min child weight	0 - 10	
	number of rounds	1 - 500	
	dropout rate	0 - 0.2	
	skip dropout	0 - 0.3	
kNN	k	1 - 10	kknn (Schliep and Hechenbichler 2014)
	kernel	Rectangular, triangular, epanechnikov, optimal	
naive Bayes	laplace	0 - 6	e1071 (Meyer et al. 2019)
SVM	lambda	0.01 - 20	liquidSVM (Steinwart and Thomann 2017)
	gamma	0.01 - 20	
	kernel	rbf, poisson	

We used early stopping (patience = 10) and a callback to reduce loss on plateaus to optimize training in DNNs.

The study was done with the statistical computing software R (R CORE TEAM, 2019). Tuning and cross-validation was implemented in our Trait-Matching package with the help of the R package mlr (BISCHL *et al.*, 2016) (Bischl *et al.* 2016, version 2.12).

---

## 4 Description of the ML algorithms

### Random Forest (RF) and boosted regression trees (BRT)

RF and BRT are based on classification and regression trees. During training, the dataset is split by specific values of the predictors' distributions. In each split, the predictors are searched for the value that split the predictors and response so that the response's variance (regression) is minimized or the accuracy is maximized (classification).

BRT fit hundreds of trees sequentially on the data. During training, the first tree has the observed labels as response while the subsequent trees have the residual errors as responses. This is known as gradient boosting.

The RF algorithm compromise two random steps: (1) RF fits hundreds of trees on bootstrap samples and (2) in contrast to BRT or trees, the RF algorithm randomly subsamples the predictors in each split (RF has to choose the best split value from a subset of predictors). Predictions in RF and BRT are averaged predictions (regression) or classification labels by majority voting.

### DNN

Deep neural networks are based on artificial neural networks. The input predictors are mapped over fully connected layers, each input node is connected to all nodes in the layer, to output nodes whose numbers correspond to the number of response classes (classification) or to one output node (regression). DNNs can consist of hundreds of hidden layers in which each node in a layer is connected to all nodes in the following layers. During training, the weights are updated by backpropagation: The weights in each layer are updated in dependence of the error in the output node by gradient descent (More detailed: A loss function specifies the error, e.g. entropy (classification), mean squared error (regression)).

### CNN

Convolutional neural networks are based on convolutional layers. Compared to DNNs, CNN can handle topological inputs such as 1-D sequences or 2-D images. A convolutional layer consists of kernels, usually small  $n \times n$  weight matrices, that apply the 'convolution' function over the input space, i.e. they run over the input space with a specific step size and use actually cross-correlation. The outputs of these kernels are feature maps. The topological information is still conserved within the feature maps. After a pooling layer (max or average reduction of feature maps with windows (e.g.  $2 \times 2$ ) and a specific step size, additional convolutional layers can follow. After the last convolutional layer, the features maps are collapsed and connected to a fully connected layer, followed by an output layer. Training is congruent to DNNs.

### k-nearest-neighbor (kNN)

The K-nearest-neighbor algorithm computes the distance between all observations in the feature space. A new observation is classified by majority vote of the k-nearest neighbors. For regression, the predicted response is the averaged response of the k-nearest neighbors. If the response is non-linear, the feature space can be transformed with a kernel (e.g. gaussian kernel).

### naïve Bayes

naive Bayes is based on the bayes theorem. It learns the probability of data point  $X_i$  belonging to class  $y_i$  given the input vector  $X_i$  (features/predictors). Naïve bayes is used only for classification.

## Support vector machines (SVM)

Support vector machines optimize hyperplanes to separate the observations into their response classes (classification) or the averaged responses (regression) in the feature space. The term ‘support vector’ refer to the computational benefit that only observations close to the plane are used to optimize the plane. To overcome the issue that only linear separable tasks work well with SVMs, the feature space can be transformed with a kernel (the kernel trick, e.g. gaussian kernel).

## Fitting models for inferring responsible trait-trait interactions

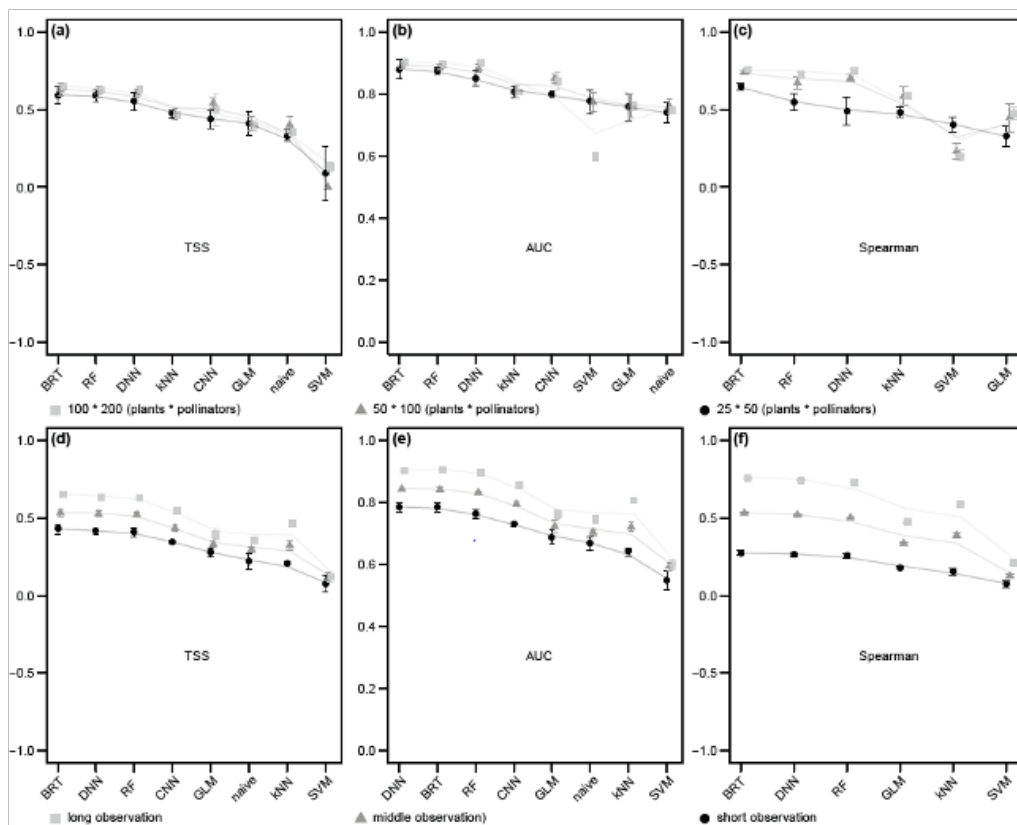
We optimized networks’ observation time in such manner that the class proportion of presence interactions was around 40%. We filtered networks for the condition that at least each insect should have one observed interaction. Only networks with a sample size with 50% of the original size were used. We fitted random forest (RF), boosted regression tree (BRT), DNN and k-nearest-neighbor (kNN) in 50 random tuning steps (Table S2.1). We used holdout validation (split 80:20) for outer and inner validation. We applied the same procedure for the 50 \* 100 networks with DNN and BRT. For inferring responsible traits, we used a grid size equal to the maximum number of rows in the data in the 25 \* 50 (plants \* pollinators) networks. For the 50 \* 100 networks, we used a grid size of 500.

## 5 Additional Results

### Comparison of predictive performance

**TABLE S2.4:** Results from the comparison of machine learning models for their predictive performance in an empirical plant-pollinator network. Rows are sorted according to TSS. AUC = area under the receiver operating characteristic curve; acc = Accuracy; tss = true skill statistic.

auc	f1	bac	acc	fdr	precision	specificity	sensitivity	tss	method
0.577	0.09	0.548	0.736	0.948	0.052	0.75	0.346	0.096	SVM
0.628	0.096	0.59	0.47	0.948	0.052	0.458	0.723	0.181	naive
0.643	0.104	0.602	0.541	0.942	0.058	0.538	0.666	0.204	GLM
0.679	0.131	0.631	0.592	0.925	0.075	0.592	0.67	0.261	BRT
0.696	0.124	0.642	0.656	0.93	0.07	0.658	0.627	0.285	DNN
0.701	0.128	0.665	0.641	0.929	0.071	0.64	0.691	0.331	CNN
0.735	0.149	0.669	0.701	0.914	0.086	0.704	0.633	0.338	RF
0.694	0.162	0.679	0.776	0.905	0.095	0.785	0.574	0.359	kNN



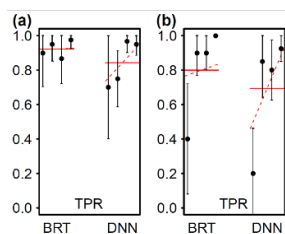
**FIGURE S2.2:** Predictive comparison of machine learning models for varying network sizes (a-c) and varying observation times (d-f). We compared three network sizes (20\*50, 50\*100, and 100\*200 species\*species), for presence-absence plant-pollinator interactions (a, b) and plant-pollinator interaction counts, (c) and three observation times (0.007, 0.0032, 0.12) for presence-absence plant-pollinator interactions (d, e) and plant-pollinator interaction counts (f). We used TSS (true skill statistic) and AUC (area under the curve) for estimating predictive performance for presence-absence (a, b, d, e) and Spearman Rho correlation factor for count frequencies (c, f).

**TABLE S2.5:** Results for baseline models. TSS = True skill statistic. AUC and TSS for presence-absence plant-pollinator interaction models and Spearman Rho correlation factor for plant-pollinator interaction count models.

Measure	dnn	cnn	knn	naive	RF	boost	glm	SVM	glm_step
AUC	0.5	0.51	0.51	0.51	0.52	0.51	0.51	0.51	0.51
TSS	0	0	0.01	0	0	0.01	0	0	0.01
Spearman	-0.02	-	-0.01	-	-0.03	-0.02	-0.03	-0.04	-0.03

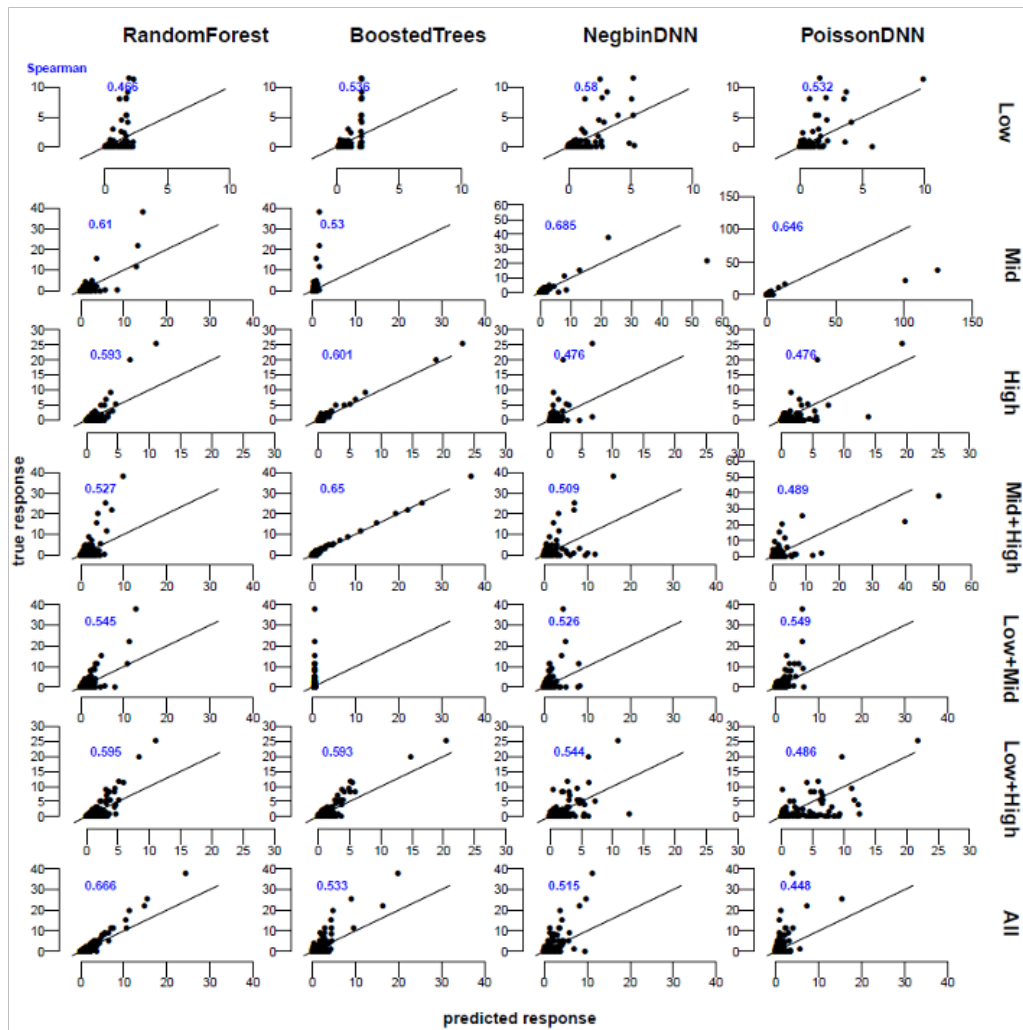
**TABLE S2.6:** Decrease in percent of performance from training to testing (generalization error) for varying network sizes. AUC for presence-absence plant-pollinator models and Spearman Rho correlation factor for plant-pollinator interaction count models.

NetworkSize	DeacraseInPercent	dnn	cnn	knn	naive	RF	boost	glm	SVM	glm_step
20*50	AUC	11	10.4	17.4	7.3	11.8	9.3	13	7	6
50*100	AUC	6.1	5.2	17	2.6	7	5.8	8.1	1	2.7
100*200	AUC	1.5	2.5	15.1	1.7	6.5	3.1	4.2	1.1	2.2
20*50	Spearman	36.7	-	37.1	-	33.4	20.9	40.1	-11.9	19.9
50*100	Spearman	11.9	-	28.6	-	21.4	9.7	19.6	-7.9	8.8
100*200	Spearman	2.6	-	26	-	14.5	4.7	9	-8.7	6.2



**FIGURE S2.3:** Averaged true positive rates for causal inferential performance on a 50\*100 simulated plant-pollinator network. We tested the performance for DNN and BRT on one to four true trait-trait interactions. Red line is the averaged mean of true positive rates. Results were higher for species presence-absence (a) interactions than for species interaction counts (b).





**FIGURE S2.4:** For estimating the predictive power of random forest, BRT, DNN with a negative binomial log-likelihood, and a DNN with a Poisson log-likelihood, we plotted the true observations versus the predictions. BRT and random forest showed best fits. Spearman rho correlation was used to quantify the predictive power and therefore the fit.

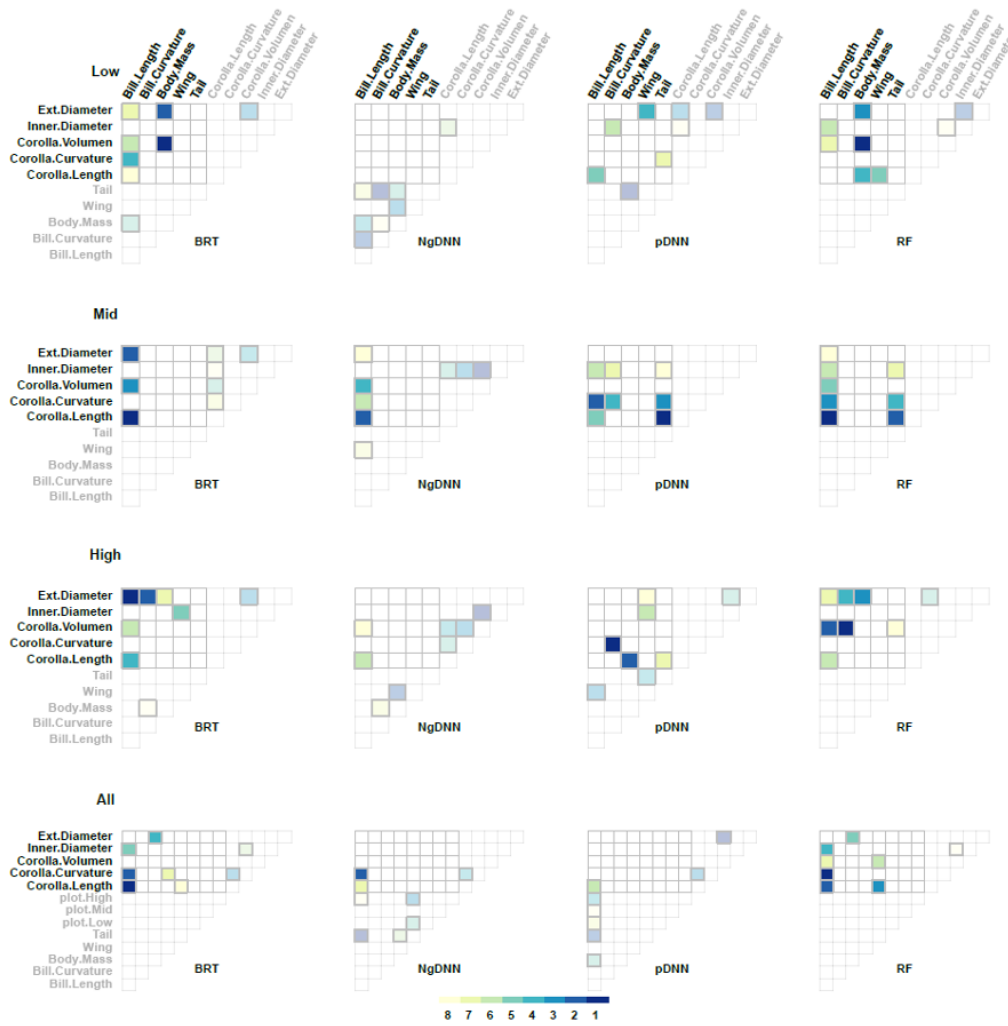


FIGURE S2.5: Enter Caption

FIGURE S2.6: Low network - BRT, DNN with poisson log-likelihood, DNN with negative binomial log-likelihood, and random forest were fit to the low plant-hummingbird network. The four traits with highest interaction strength versus all traits and for each of those the top two pairwise interactions were here visualized. Random forest identified mainly trait combinations between plants and hummingbirds with higher interaction strengths.

# Supporting Information S3 for Chapter 4

## 1 Simulation scenarios

In this section, we describe how the simulations of data under the MVP model were obtained. The MVP can be interpreted as individual GLMs connected by correlated residuals, which are sampled from a multivariate Gaussian, and with a probit link. In the equations below, sites are notated with  $i = 1, \dots, M$ ; species with  $j = 1, \dots, K$ ; and environmental covariates with  $n = 1, \dots, N$ . Environmental covariates and species responses ( $\beta$ ) were uniformly sampled (Eq. 1). The lower triangular covariance matrix was uniformly sampled (Eq. 2), the diagonal was set to one (Eq. 3) and multiplied by the transposed lower triangular to get a symmetric positive definite matrix (Eq. 4). Afterwards, the covariance matrix ( $\Sigma$ ) was normalized to the range of  $[-1, 1]$  (Eq. 6).

$$\beta, X \sim U(-1, +1) \quad (9.1)$$

$$\Sigma_{j \neq j}^{lower} \sim U(-1, +1) \quad (9.2)$$

$$\Sigma_{j=j}^{lower} = 1 \quad (9.3)$$

$$\Sigma = \Sigma^{lower} * \left(\Sigma^{lower}\right)^T \quad (9.4)$$

$$D = \sqrt{\text{diag}(\Sigma)} \quad (9.5)$$

$$\Sigma' = \text{diag}(D)^{-1} \Sigma \text{diag}(D)^{-1} \quad (9.6)$$

$$Z_{ij} = \beta_{0j} + \sum_{n=1}^N X_{in} * \beta_{nj} + e_{ij} \quad (9.7)$$

$$e_i \sim MVN(0, \Sigma') \quad (9.8)$$

$$Y_{ij} = 1 (Z_{ij} > 0) \quad (9.9)$$

Species responses consist of a linear species - environmental response and correlated residuals (Eq. 7). Following a probit link, responses higher than zero were set to 1 and the remaining to 0 (Eq. 9).

## 2 Approximation of multivariate probit model

In this section, we describe our approximation of the multivariate probit model. We denote the multivariate normal PDF is by  $\phi$ . The probability to observe  $\mathbf{Y}_i$  for  $\mathbf{X}_i$  by the environmental coefficient matrix  $\beta$  and the covariance matrix  $\Sigma$  is given by:

$$P(\mathbf{Y}_i | \mathbf{X}_i, \beta, \Sigma) = \int_{A_j} \dots \int_{A_1} \phi(\mathbf{Y}^*, \mathbf{X}_i, \beta, \Sigma) dY_1^* \dots dY_J^* \quad (9.10)$$

with

$$A_j = \begin{cases} (-inf, 0] & y_{ij} = 0 \\ [0, +inf] & y_{ij} = 1 \end{cases} \quad (9.11)$$

We can rewrite this as the cumulative density function ( $\Phi$ ) of the multivariate normal distribution:

$$D_i = \text{diag}(2\mathbf{Y}_i - 1) \quad (9.12)$$

$$\mu_i = D_i(\mathbf{X}_i\beta) \quad (9.13)$$

$$\Sigma^* = D_i\Sigma D_i \quad (9.14)$$

$$P(\mathbf{Y}_i|\mathbf{X}_i\beta, \Sigma) = \Phi(0 | -\mu_i, \Sigma^*) \quad (9.15)$$

To approximate the likelihood in sjSDM, we use a Monte-Carlo approach that was suggested in a slightly different context by CHEN, XUE, and GOMES (2018). In the following, we shortly sketch the idea. With  $\mathbf{r} \sim N(0, \Sigma^*)$ , we can rewrite Eq. 11 as:

$$\Phi(0 | -\mu_i, \Sigma^*) = Pr(r_i - \mu_i \leq 0) \quad (9.16)$$

We can now reparametrize  $\Sigma^* = V + \Sigma^r$  with  $V$  as a diagonal, which means that the random variable  $\mathbf{r}$  depends on two other random variables  $\mathbf{z} \sim N(0, V)$  and  $\mathbf{w} \sim N(0, \Sigma^r)$  and following  $\mathbf{r} = \mathbf{z} + \mathbf{w}$  Eq. 12 is equal to:

$$\begin{aligned} \Phi(0 | -\mu_i^*, \Sigma^*) &= Pr(\mathbf{r}_i - \mu_i \leq 0) \\ &= Pr(\mathbf{z}_i - \mathbf{w}_i \leq \mu_i) \end{aligned} \quad (9.17)$$

In this way we transform the individual Monte-Carlo samples with the covariance matrix, and we can treat them as univariate samples and use the univariate normal CDF:

$$\begin{aligned} &= \mathbb{E}_{\mathbf{w} \sim N(0, \Sigma^r)} [Pr(z \leq (\mathbf{w}_i + \mu_i) | \mathbf{w}_i)] \\ &= \mathbb{E}_{\mathbf{w} \sim N(\mu_i, \Sigma^r)} \left[ \prod_{j=1}^J \left( \frac{\Phi(w_{ij})}{\sqrt{V_{jj}}} \right) \right] \end{aligned} \quad (9.18)$$

As  $V$  is used to rescale the univariate samples, we use w.l.o.g. identity matrix  $I$  instead  $V$  and estimate only the residual  $\Sigma^r$ . Following this, the final approximation is:

$$\approx \frac{1}{M} \sum_{m=1}^M \prod_{j=1}^J \Phi(\mathbf{w}_j^m) \quad (9.19)$$

We can rewrite Eq. 15 as, with  $\Sigma^{1/2}$  being the square-root matrix of  $\Sigma^r$ :

$$\begin{aligned} \mathbf{w}_i^{(m)} &= \mathbf{X}_i\beta + \Sigma^{1/2}\mathbf{z}_i^{(m)}, \mathbf{z} \sim N(0, I) \\ P(\mathbf{Y}_i|\mathbf{X}_i\beta, \Sigma) &\approx \frac{1}{M} \sum_{m=1}^M \prod_{j=1}^J \phi(w_{ij}^{(m)}) Y_{ij} + (1 - Y_{ij}) \left( 1 - \phi(w_{ij}^{(m)}) \right) \end{aligned} \quad (9.20)$$

For optimizing the parameters, we use the automatic derivatives implemented in PyTorch to find the gradient for each Monte-Carlo particle, and average gradients of all particles to obtain a gradient for the optimizer. In short, the core of the algorithm is to generate an approximation of the gradient of the likelihood by drawing from the multivariate normal distribution in the MVP model and propagating the calculations for the resulting draws through the entire model structure.

We said earlier that the random variable  $\mathbf{r}$  consists of actual two random variables  $\mathbf{z}$  and  $\mathbf{w}$ , and thus the covariance matrix  $\Sigma^*$  can be decomposed into  $\Sigma^* = I + \Sigma^r$ . For the actual sampling from the covariances ( $\Sigma^r$ ), we can use the square root matrix:

$$\Sigma^* = I + \Sigma^{\frac{1}{2}} * \left(\Sigma^{\frac{1}{2}}\right)^T \quad (9.21)$$

The square root matrix  $\Sigma^{\frac{1}{2}}$  has dimensions  $J$  (number of species)  $\times d$  and is the actual parameter matrix we optimize in sjSDM.  $d \ll J$  corresponds to a low-rank parametrization, increasing  $d$  increases the overall number of parameters parametrizing  $\Sigma^r$ . The advantage of this re-parametrization is that  $\Sigma^{\frac{1}{2}}$  needs not to be symmetric and positive definite, so we can use it directly for sampling without, for example, using a Cholesky decomposition which is computationally expensive and numerically unstable for high  $J$ . Moreover,  $d \ll J$  can be seen as another way to regularize the covariance matrix (it is similar to the LVM). However, in sjSDM we use an elastic-net regularization on the individual entries of  $\Sigma^*$  (we could also use the precision matrix of  $\Sigma^*$  here (which is implemented in the sjSDM package)) and we found in interim results that for  $d \ll J$  the elastic-net regularization does not work properly (i.e., all entries are regularized to 0 or not). We assume that for  $d \ll J$  the covariances in  $\Sigma^*$  are not ‘independently enough’ parametrized (not in the statistical sense) since the regularization of individual covariances of  $\Sigma^*$  leads to an indirect regularization of  $\Sigma^{\frac{1}{2}}$ . To use elastic-net regularization we set  $d = J/2$  as default in the sjSDM implementation, but future research may be needed to explore the trade-off between regularization via  $d$  and regularization via elastic-net on the covariance matrix.

The model itself is optimized via stochastic gradient descent (BOTTOU, 2010) which means that the estimates of the model are always updated only on a random batch of the dataset and thus one iteration (called epoch in the deep learning field) consists of the number of optimization steps necessary to go once through the full dataset (e.g. for 100 observations and a batch size of 10, an iteration would consist of 10 optimization steps).

The univariate probit link ( $\Phi$  in Eq. 16) can be approximated by scaling the logit link  $F$ :  $\Phi(x) \approx F(x * 1.7)$  BAKER and KIM, 2004, which we found in interim results more beneficial than the analytic probit link. Stochastic gradient descent is numerically often unstable, so it is not uncommon in machine learning approaches to adjust activation functions (XU *et al.*, 2015; ZHAO, ZHANG, *et al.*, 2017) and model structures to improve convergence properties of the algorithm and we assume that the approximation of the probit via scaling of the logit link is numerically more stable than the actual analytic probit link.

### 3 Runtime benchmarks using the data from Wilkinson et al. 2019

Parts of our benchmarks used the compiled datasets from WILKINSON *et al.* (2019). TableS3.1 provides an overview of these datasets.

**TABLE S3.1:** Compiled datasets that were taken from WILKINSON *et al.* (2019)

Dataset	Original paper	Species	Sites	Covariates
Birds	Harris 2015	370	2,752	8
Butterflies	Ovaskainen et al. 2016	55	2,609	4
Eucalypts	Pollock et al. 2014	12	458	7
Frogs	Pollock et al. 2014	9	104	3
Fungi	Ovaskainen et al. 2010	11	800	15
Mosquitos	Golding 2015	16	167	13

#### 4 Model settings and computational environment for the benchmarks

This section provides a more detailed explanation about model settings and the computer setup under which our benchmarks were performed (see Table S3.1 for an overview). Unless stated otherwise, we used default settings for parameters.

**TABLE S3.2:** Overview of the used approaches

Model	Optimization type	Package
Multivariate probit model	MLE	sjSDM
	MCMC	BayesComm
Latent-variable model	MCMC	Hmsc
	Variational bayes / Laplace approximation	gllvm

##### BayesComm

BayesComm models were fitted with 50,000 MCMC sampling iterations, with two chains, thinning = 50, and burn-in of 5000. Prior were not changed from default: normal prior on regression coefficients  $\beta \sim N(0; 10)$  and an inverse Wishart prior on the covariance matrix.

##### Hmsc

Hmsc models were fitted with 50,000 MCMC sampling iterations, with two chains, thinning = 50, and burn-in of 5000. Since the two chains were not run in parallel (although it is supported by Hmsc), the measured runtime was halved. The number of latent variables in Hmsc are automatically inferred by gamma shrinkage priors. The shrinkage priors of Hmsc were not changed from default. We note that there is the option to tune regularization via shrinkage priors in Hmsc: a1 regularizes the lower triangular of the species association and a2 regularizes the number of latent variables (BHATTACHARYA and DUNSON, n.d.). We acknowledge that this might improve accuracy of Hmsc inference. On the other hand, it should be noted that a) these settings were not tuned in recent benchmarks and are likely not tuned by users either. b) the runtime of tuning several combinations would be not practicable (see our results) and c) it is to be expected that a low a2 results in a higher accuracy but then the LVM approach would approximate the MVP model and that would contradict the LVM's unique characteristic.

## gllvm

gllvm models were fitted as binomial models with probit link. The number of latent variables were increased from 2 to 6 with the number of species. If default starting values = “res” caused an error, model was re-run with starting values = “zero” and if another error occurred, the model was re-run with starting values = “random”. Run time was measured individually, not as a sum over possible model fitting tries.

## sjSDM

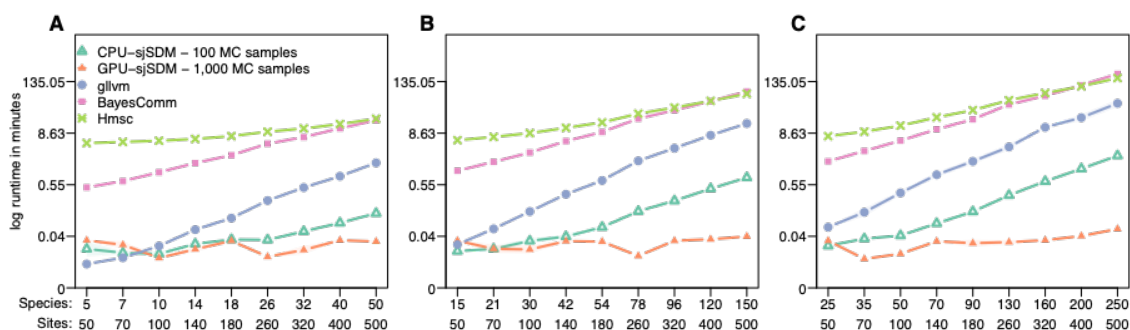
sjSDM models were fitted with 50 iterations (epochs) and a batch size of 10% number of sites. Learning rate was set to 0.01. 50% number of species were set for  $d$  for the parametrization ( $J * dweights$ ) of the covariance matrix (see section about the approximation, default in the sjSDM R-package). For sparse species association matrices, 50 iterations with a learning rate of 0.01 were used and the regularization of the species-species covariances were tuned in 40 random steps and 5-folded cross-validation. 2,000 Monte-Carlo samples were used for the MVP approximation because interim results showed that with too less samples the variance of the models’ log-Likelihoods might interfere with the comparison.

**TABLE S3.3:** Model settings of JSDM implementations Hmsc, BayesComm, gllvm, and sjSDM.

Model	Parameter	Value
Hmsc	Iterations (MCMC samples)	50,000
	Burnin	5,000
	Thinning	50
	Number of chains	2
BayesComm	Distribution	Binomial with probit link
	Iterations (MCMC samples)	50,000
	Burnin	5,000
	Thinning	50
gllvm	Number of chains	2
	Distribution	Multivariate probit link
		2 for 10% species / site proportion
	Number of latent variables	3 for 30% species / site proportion
sjSDM (non-sparse association matrices)		4 for 50% species / site proportion
	Distribution	Binomial with probit link
	Starting values	'res' (residuals). If model did not converge, retry with 'zero' and if model still didn't converge, retry with 'random'
	Learning rate	0.1
sjSDM (sparse associations)	Iterations	50
	MC-samples for each species	100 (CPU); 1,000 (GPU)
	Distribution	Multivariate probit link
	Learning rate	0.01
sjSDM (sparse associations)	Iterations	50
	MC-samples for each species	2,000 (GPU)
	Random tuning steps	40
	n-folded cross-validation	5
	Lambda range (regularization strength)	[9.765e-5, 0.063]
	Alpha (weighting between LASSO and ridge)	[0.0, 1.0]
	Distribution	Multivariate probit link

## Computer setup

All the computations were performed on the same workstation (two Intel Xeon Gold 6128 CPU @3.40 GHz). The number of cores and threads were restricted to 6. GPU computations were carried out on a NVIDIA RTX 2080 Ti. All CPU models had access to 192 GB RAM and the GPU models to 11 GB GPU RAM. Analyses were conducted with the statistical software R and Python (Python Software Foundation. Python Language Reference, version 3.8.1. Available at <http://www.python.org>)



**FIGURE S3.1:** Results for the log runtime of GPU-sjSDM, CPU-sjSDM, gllvm, BayesComm, and Hmsc. Models were fit to different simulated SDM scenarios: 50 to 500 sites with A) 10%, B) 30%, and C) 50% number of species (e.g., for 100 sites and 10% we get 10 species). For each scenario, ten simulations were sampled, and results were averaged. Due to high runtimes, runs for BayesComm, gllvm and Hmsc were aborted at specific points.

## 5 Additional results

### Run time scaling of the algorithms in log plots

In the main paper, we provide the benchmarks on a linear scale. Below, we also provide them in log format, which demonstrates that many other software packages, including the CPU version of sjSDM scale close to exponentially, while GPU-sjSDM scales sub-exponentially for the scenarios that we tested (Fig S3.1).

### Quantifications of inferential accuracy

We additionally calculated the root mean squared error (RMSE) between the true simulated and estimated association matrices (Fig. S3.2). The pattern we found is consistent with the accuracy of the signs of the covariances. For dense and sparse association matrices, BayesComm and sjSDM achieved the lowest RMSE in all scenarios (Fig. S3.2).

Moreover, we calculated the true skill statistic (TSS) for different thresholds (0.005, 0.01, 0.05, and 0.1, entries below the threshold are classified as 0) when evaluating the accuracy in inferring zero and non-zero entries in the covariance matrices (Fig. S3.3). We found that sjSDM and BayesComm achieved the highest TSS for thresholds up to up to 0.05. With a threshold of 0.05, gllvm achieved a similar TSS as sjSDM and BayesComm. It seems the performance of BayesComm and especially for sjSDM decreases for higher thresholds, however, with increasing thresholds higher non-zero values are getting classified as zero resulting in non-sparse association matrices, thereby the decrease in TSS.

### Additional results from fitting sjSDM on large datasets

Beside the runtimes for the large-scale datasets (see main paper), we also calculated the accuracy of the matching signs of predicted and true parameters for the association matrix and environmental coefficients (Table S3.4). Moreover, the RMSE for the environmental coefficients were calculated (Table S3.4).

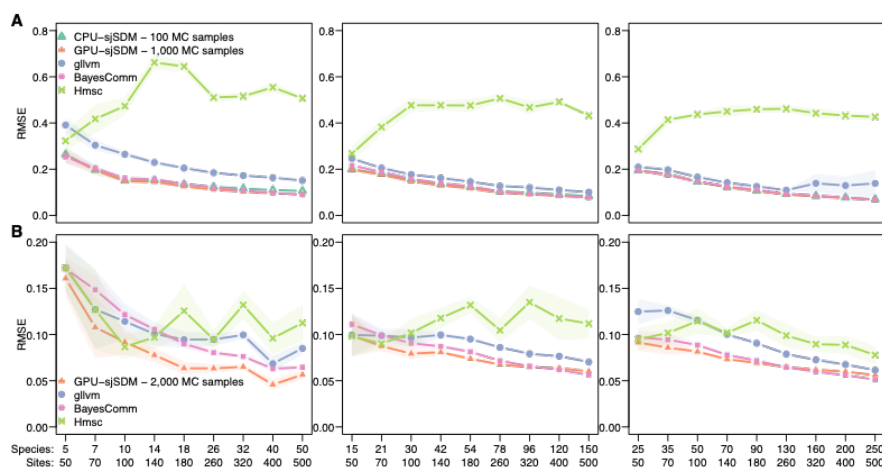
Overall, we found that the association accuracy decreased from 300 to 1000 species with the different number of sites, but overall, the association accuracy increased with the number of sites (Table S3.4). The accuracy of environmental coefficients was close to 1.0 in all scenarios and the



RMSE for the environmental coefficients was close to zero in all scenarios (Table S3.4).

**TABLE S3.4:** Accuracy of matching signs for estimated associations, environmental coefficients, and root mean squared error (RMSE) for environmental coefficients

Sites	5,000			15,000			30,000		
Species	300	500	1000	300	500	1000	300	500	1000
Covariance accuracy	0.744	0.721	0.688	0.750	0.727	0.693	0.750	0.728	0.692
Env accuracy	0.988	0.985	0.987	0.990	0.989	0.991	0.990	0.990	0.990
Env RMSE	0.040	0.037	0.035	0.034	0.029	0.026	0.033	0.029	0.025



**FIGURE S3.2:** Root mean squared error of the inferred A) non-sparse and B) sparse species-species associations. Models were fitted to simulated data with 50 to 500 sites and the number of species set to 0.1, 0.3 and 0.5 times the number of sites. All values are averages from 5 simulated datasets.

For sparse species associations, the models achieved similar performances in inferring the environmental parameters (Fig. S3.4 C-D).

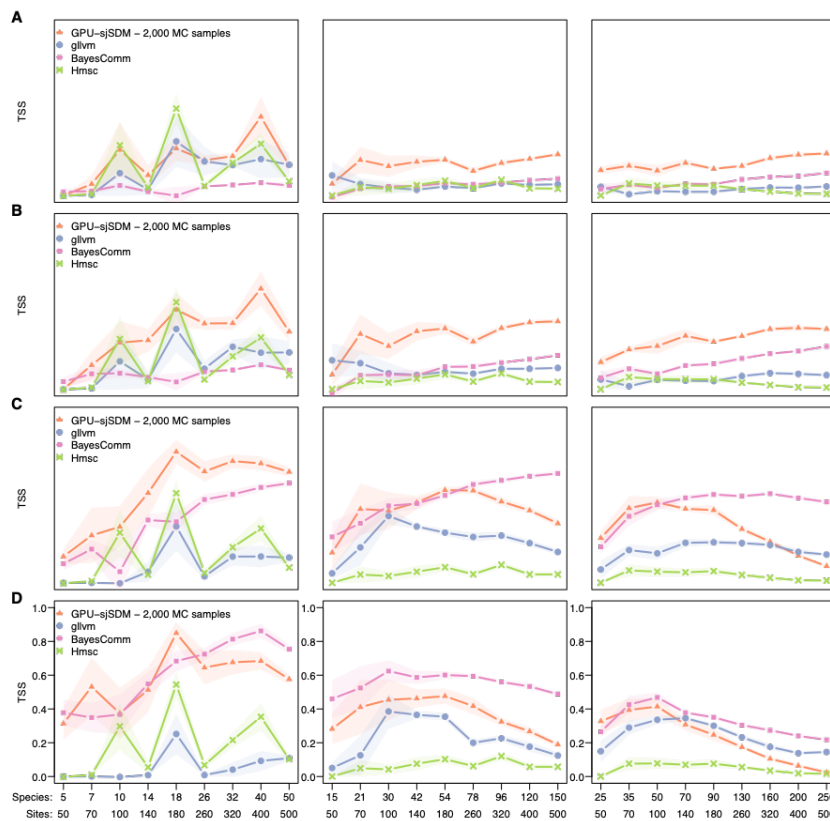
### Convergence check

To check convergence of  $\Sigma$  and  $\beta$ , the potential scale reduction factors (psrf) for Hmsc and BayesComm in the simulation scenarios (dense and sparse association scenarios) were calculated (two chains, burn-in = 5000 and 50,000 sampling iterations). We found no  $psrf > 1.2$  for BayesComm, but for Hmsc in most simulation scenarios at least for one parameter ( $\beta$  or factor loadings) a  $psrf > 1.2$  (Fig. S3.5).

For Hmsc, we removed the psrf factors for the last latent factor loading because interim results showed that the psrfs for the last factor loading were on average always higher than 1.2, although they were estimated to be very small (perhaps a numerical issue).

### Dependence of the inferred associations on the number of sites

To further assess the jSDM's behavior in inferring the species-species association matrix, we set the number of species to 50 and increased the number of sites from 50 to 330. For each, step we



**FIGURE S3.3:** True skill statistic (TSS) for sparse absolute species-species associations with different thresholds: A) 0.005, B) 0.01, C) 0.05, and D) 0.1. Models were fitted to simulated data with 50 to 500 sites and the number of species set to 0.1, 0.3 and 0.5 times the number of sites. All values are averages from 5 simulated datasets.

computed the averaged (we sampled 5 scenarios for each setting) covariance accuracy (matching signs) and environmental RMSE.

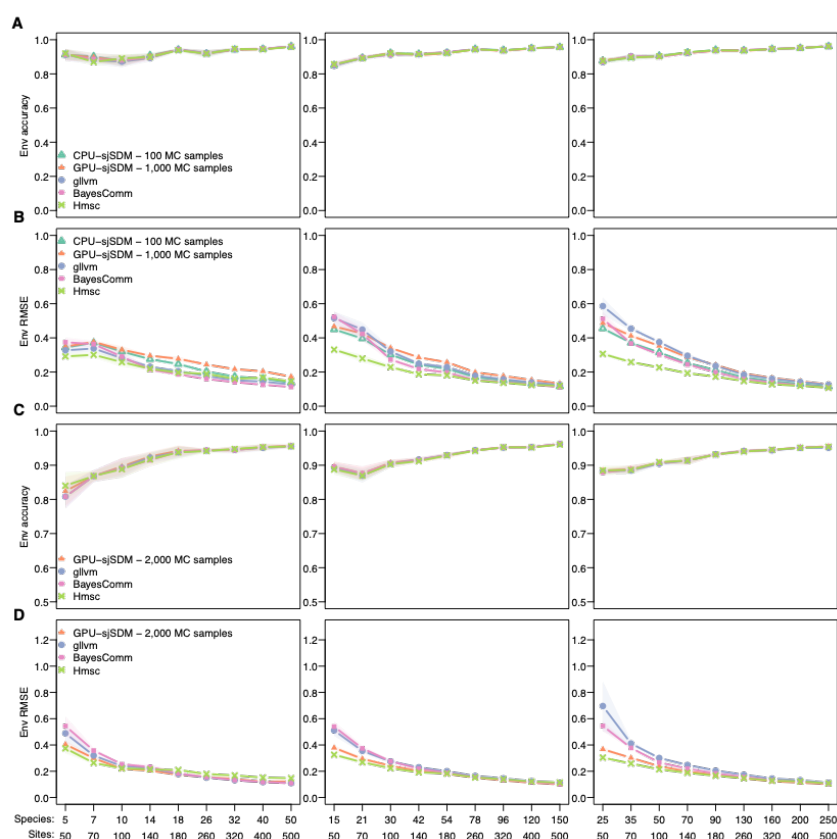
BayesComm achieved at 330 sites around 0.82 accuracy and sjSDM around 0.80. sjSDM and BayesComm increased the covariance accuracy steadily with the number of sites, while Hmsc and gllvm stopped increasing their accuracy at around 0.68 accuracy (Fig. S3.6 A). sjSDM and BayesComm achieved in average 0.1 more accuracy than Hmsc and gllvm (Fig. S3.6 A).

All models achieved a similar RMSE over all scenarios (Fig. S4.6 B). sjSDM showed overall the highest RMSE (Fig. S3.6 B). All models decreased their RMSE with increasing number of sites (Fig. S3.6 B).

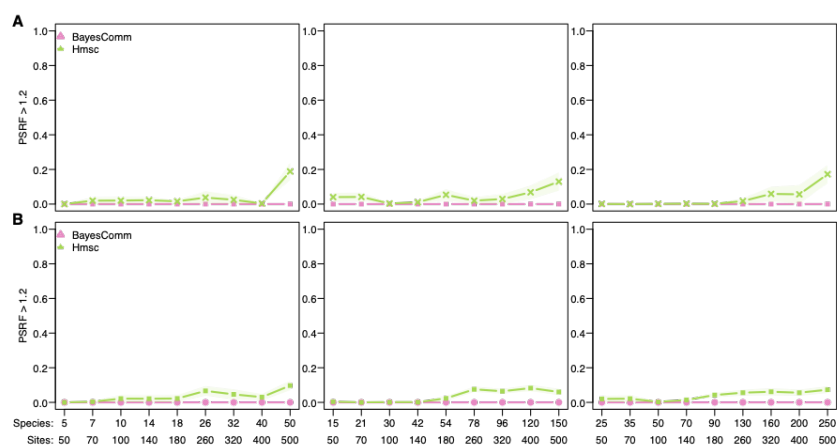
### Ability of the models to recover associations for data simulated from a Latent Variable Model

We also simulated new data from a latent variable model varying the number of species from 10 to 100 (10, 50, and 100) and the number of latent variables (1 - 5) with a constant number of 200 sites. In all simulations, the species' environmental preference was described for two environmental covariates ( $\beta$ ).

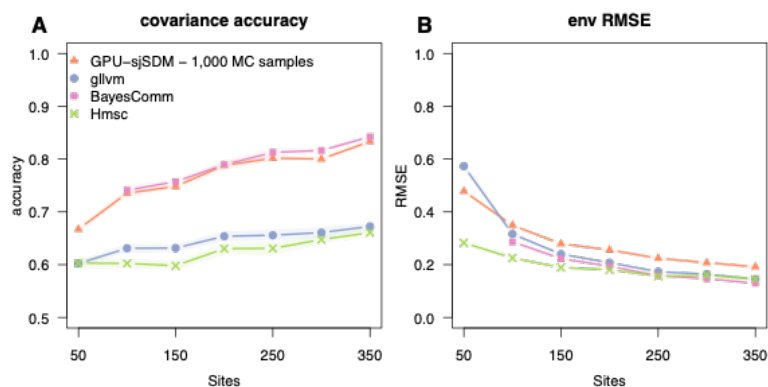
Observed environment was sampled from a uniform distribution with a range of  $[-1, 1]$ . Unob-



**FIGURE S3.4:** Results for inferential benchmarking of G-sjSDM with gllvm, BayesComm, and Hmsc as references. Models were fitted to different simulated SDM scenarios: 50 to 500 sites with 10% (first column), 30% (second column) and 50% (third column) number of species to site proportions (e.g., for 100 sites and 10% we get 10 species). For each scenario, 5 simulations were sampled, and results were averaged. A) and B) show the environmental coefficient accuracy (matching signs) and the corresponding RMSE with full species-species association matrices. C) and D) show the environmental coefficient accuracy (matching signs) and the corresponding RMSE with sparse (50% sparsity) species-species association matrices.



**FIGURE S3.5:** Rate of weights in percent with potential scale reduction factor > 1.2 with non-sparse and sparse association matrices in simulation scenarios for A) Hmsc (for factor loadings and beta estimates) and B) BayesComm (covariance and beta estimates).



**FIGURE S3.6:** Results for examining the ability to recover the covariance structure as a function of the sites for GPU-sjSDM, BC, gllvm, and Hmsc. In the simulated species distribution scenarios, the number of species were constantly set to 50, but the number of sites were changed from 50 to 330 sites. A) Performance was measured by the accuracy of matching sings between estimated covariances matrices and true covariance matrices. B) Moreover, the root mean squared error for the environmental effects with the true coefficients were calculated.

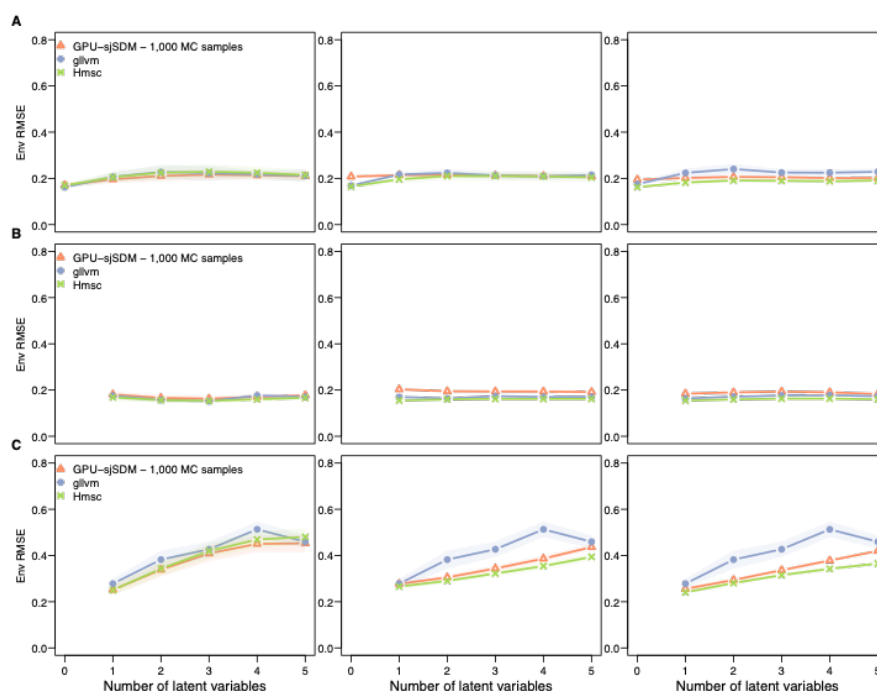
served environment (latent variables) was sampled from a normal distribution with mean equal 0.0 and standard deviation equal 1.0. For each scenario, we simulated 5 communities. The number of latent variables in gllvm was set to the real number of latent variables.

Environmental coefficients and factor loadings were sampled from a uniform distribution. We tested different scenarios: no latent variables at all (i.e. factor loadings equal zero), equal ranges/weighting between environmental coefficients and factor loadings (i.e. both were sampled from a uniform distribution within  $[-1, 1]$ , 5 : 1 weighting of environmental coefficients to factor loadings (i.e. environmental coefficients were sampled from  $[-5, 5]$  and factor loadings from  $[-1, 1]$ ), and a 1 : 5 weight of environmental coefficients to factor loadings (i.e. environmental coefficients were sampled from  $[-1, 1]$  and factors loadings from  $[-5, 5]$ ).

We found for all three scenarios no large differences between the three JSDM (Fig. S3.7-S3.9). sjSDM was able to achieve the same or even better performance than the two LVM JSDMs (Fig. S3.8A, C)

### Interpretation of the results for the eDNA dataset

For the eDNA dataset we found that the strongest negative associations are among the most abundant species while the strongest positive associations are between the rarest species (Fig. 4.5 A-B). Moreover, it appears that the most abundance species differ in their most import environmental coefficients (Fig. 5 D) suggesting that the abundant species occupy different niches within a community (site). Whether this pattern is caused by environmental filtering or biotic interactions (i.e. interspecific competition) is open for discussion (KRAFT *et al.*, 2015; GERMAIN, MAYFIELD, and GILBERT, 2018). Perhaps additional information such as traits or the comparison at different scales is required to disentangle the different factors (e.g. BOET, ARNAN, and RETANA, 2020). Also, we visualized here only the strongest 60 (30 negative and 30 positive associations) and we did not quantify the results. On the other hand, the rare species showed positive associations and more similar important environmental associations than the abundant species, suggesting a signal from biotic interactions rather than from environmental filtering here. However, a more in-depth analysis is required to actually infer meaningful ecological insights from this dataset.

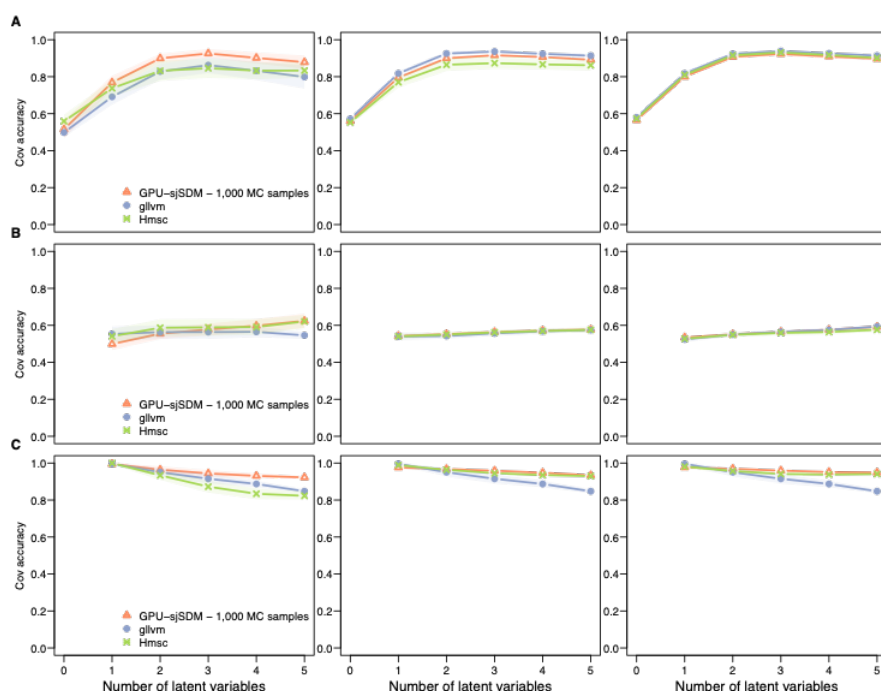


**FIGURE S3.7:** Results for communities simulated by the latent variable model. Models were fitted to different simulation scenarios: 10, 50 and 100 species with 1 - 5 latent variables. Number of environmental coefficients were set to two. For each scenario, the RMSE error between true and estimated environmental coefficients over the five repetitions were averaged. A) shows the results for equal weighting B) for 5:1 (environment to latent) weighting and C) for 1:5 (environment to latent) weighting between the environmental and latent coefficients in the simulation.

## 6 Ability of the models to recover processes from data simulated from a process-based community model

To explore the behavior of our new method compared to other JSDM implementations when using data that was not created according to the assumed model structure, we simulated community data from the process-based community model used in LEIBOLD, RUDOLPH, *et al.* (2022).

A brief description of the process-based community model For a detailed explanation of the process-based community model, see LEIBOLD, RUDOLPH, *et al.* (2022). Briefly, the model is a spatially implicit time-discrete site occupancy model where local colonization depends on the number of immigrants, the environmental suitability, and the ecological interactions, and extinction depends on the environment and ecological interactions. LEIBOLD, RUDOLPH, *et al.* (2022) proposed that JSDMs can be used to analyze the community patterns created by the model and separated them into three main contributions: space (immigration), environmental filtering, and co-occurrences of species (biotic interactions). Data simulation scenarios We simulated data under the same 7 scenarios as LEIBOLD, RUDOLPH, *et al.* (2022). All scenarios had 12 species over 1000 sites over 5 discrete time intervals: A and B without ecological interactions but with a narrow and wide environmental niches, C and D with ecological interactions (interspecific competition effects) and narrow and wide environmental niches, E with half of the species (6) having ecological interactions, F without interactions but with 4 species having low dispersal, 4 species having medium dispersion, and 4 species having high dispersion, and G with different dispersions as in F but with interactions between the 4 species in the three groups (Table S3.5). For each scenario, 5 temporally sequential realizations were simulated.



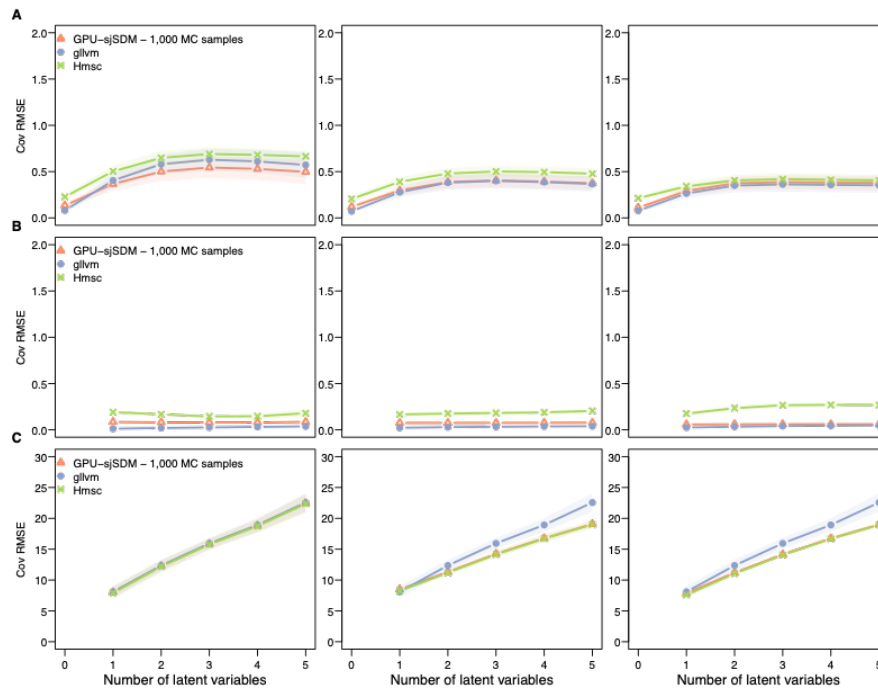
**FIGURE S3.8:** Results for communities simulated by the latent variable model. Models were fitted to different simulation scenarios: 10, 50 and 100 species with 1 - 5 latent variables. Number of environmental coefficients were set to two. For each scenario, the accuracy of matching signs of the covariances in the association matrix between true and estimated covariance matrix over the five repetitions were averaged. A) shows the results for equal weighting B) for 5:1 weighting and C) for 1:5 weighting between the environmental and latent coefficients in the simulation.

**TABLE S3.5:** Different scenarios for the simulations from the process-based community model from LEIBOLD, RUDOLPH, *et al.*, 2022

Scenario	Niche	Interactions	Dispersal
A	0.8	Colonization = extinction = 0.0	0.05
B	2.0	Colonization = extinction = 0.0	0.05
C	0.8	Colonization = extinction = 1.0	0.05
D	2.0	Colonization = extinction = 1.0	0.05
E	0.8	6/12 species: Colonization = extinction = 0.0; 6/12 species: Colonization = extinction = 1.0	0.05 0.05
F	0.8	Colonization = extinction = 0.0	4/12 species: 0.01 4/12 species: 0.05 4/12 species: 0.1
G	0.8	Colonization = extinction = 1.0	4/12 species: 0.01 4/12 species: 0.05 4/12 species: 0.1

### Model estimation

Following LEIBOLD, RUDOLPH, *et al.*, 2022, we fitted the environmental variable E with a linear and a quadratic effect. Also, 50 spatial eigenvectors were fitted as main effects to account for space.



**FIGURE S3.9:** Results for communities simulated by the latent variable model. Models were fitted to different simulation scenarios: 10, 50 and 100 species with 1 - 5 latent variables. Number of environmental coefficients were set to two. For each scenario, the RMSE between the true and estimated covariance matrix (normalized to correlation matrices) over the five repetitions were averaged. A) shows the results for equal weighting B) for 5:1 weighting and C) for 1:5 weighting between the environmental and latent coefficients in the simulation.

We fitted sjSDM, Hmsc, BayesComm, and gllvm on each time step (5 discrete time intervals) for each scenario. The exact parameters of the models are described in Table S3.6.

For each model, the covariance matrix was extracted and if necessary normalized to  $[-1, 1]$ .

**TABLE S3.6:** Model settings of JSJM implementations Hmsc, BayesComm, gllvm, and sjSDM for process-based community simulations.

Model	Parameter	Value
Hmsc	Iterations (MCMC samples)	50,000
	Burnin	5,000
	Thinning	50
	Number of chains	1
	Distribution	Binomial with probit link
	Iterations (MCMC samples)	50,000
BayesComm	Burnin	5,000
	Thinning	50
	Number of chains	1
	Distribution	Multivariate probit link
gllvm	Number of latent variables	2
	Distribution	Binomial with probit link
	Starting values	'zero' (residuals). If model did not converge, retry with 'random' and if model still didn't converge, retry with 'res'
	Learning rate	0.1
sjSDM	Iterations	100
	MC-samples for each species	1,000 (GPU)
	Distribution	Multivariate probit link

## Evaluation of the JSDM for the process-based simulations

While Leibold et al. 2021 was most interested in investigating the ability of JSDM to separate the three functional processes of the simulation (space, biotic, and abiotic effects) across sites and species, we will only concentrate on the association structures found by the JSDMs. Apart from performing an additional test of sjSDM, this analysis also allows to comment on the current discussion about connection between inferred species associations and biotic interactions.

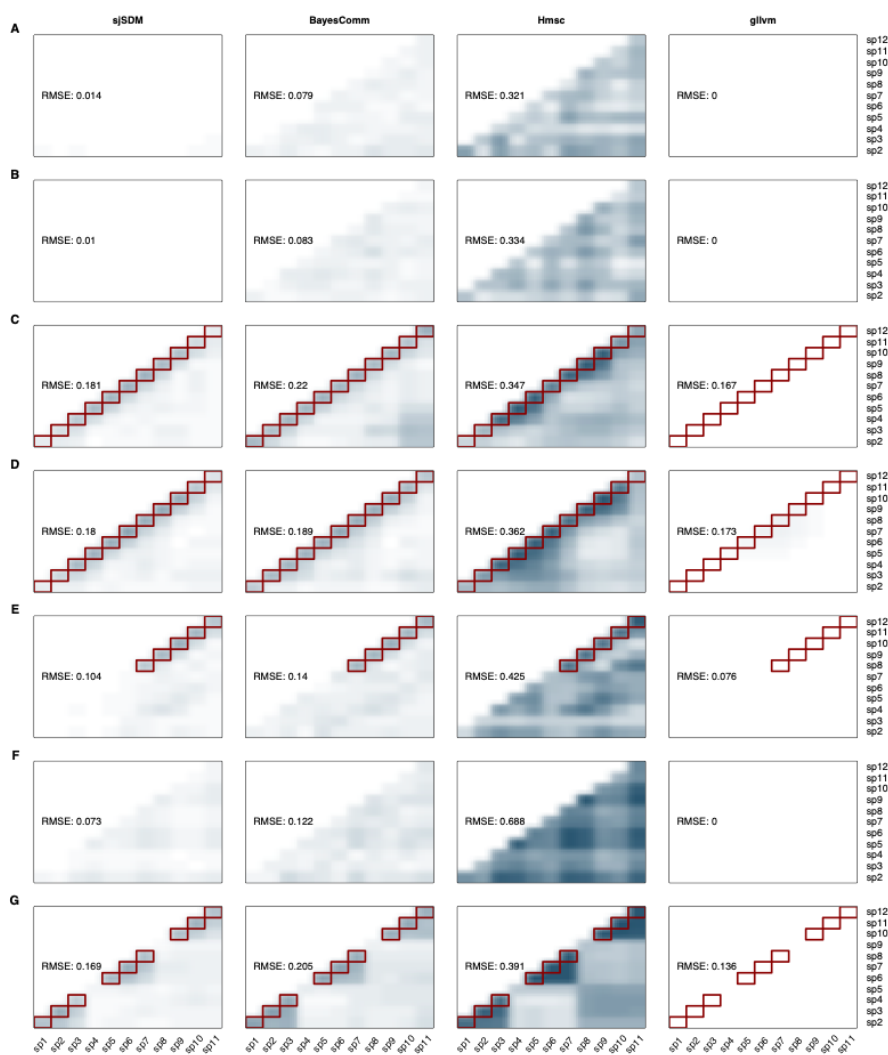
To address this point, the extracted (and normalized) covariance matrices were summed up over the 5 temporal steps and divided by 5. Previous the summarization, we took the absolute values of the covariance matrices since the simulation consisted of asymmetric associations whereas JSDM can only estimate symmetric associations and we were mainly interested in the question whether the models were able to find a signal in the associations – regardless of the sign of the covariance. We then calculated the root mean squared error (RMSE) between the true (simulated) biotic interactions (with entries of 1 between interacting species) and the estimated association matrices from the JSDM.

## Results of the JSDM for the process-based simulations

With no interactions (scenarios A, B, and F), we found that sjSDM, BayesComm and gllvm estimated associations close to zero (sjSDM and BayesComm) or exact zero (gllvm), while Hmsc did not always estimate associations to zero (Fig. S3.10 A, B, F). With interactions (scenarios C - E, G), sjSDM and BayesComm estimated similar association structures as in the true interaction structure (Table S5, red rectangles Fig. S3.10 C - E, G). Hmsc tended again to overall higher associations between all species, but with higher values for the true interactions. Gllvm estimated associations close to zero in all scenarios. Generally, sjSDM and BayesComm achieved the lowest RMSE in all scenarios except for gllvm (Fig. S3.10).

Overall, the LVM-based JSDMs (Hmsc and gllvm) estimated either too many associations unequal to zero (Hmsc) or all associations close to zero (gllvm). We speculate that the differences may be due to the prior settings for the latent variables (or factor loadings): The prior might be too restrictive for gllvm and not restrictive enough for Hmsc. Nevertheless, Hmsc showed again the typical block structures in the estimated covariance matrix (Fig. S3.10G) which is typically for low-rank approximations





**FIGURE S3.10:** Results for simulations by the process-based community model from LEIBOLD, RUDOLPH, *et al.* (2022). Following LEIBOLD, RUDOLPH, *et al.* (2022), we tested 7 different scenarios with different niche widths (A-D), with and without interactions (A, B, and F without interactions), and different dispersal rates (F and G). Red rectangles show the true association matrices. For each scenario, 5 communities were temporally sequentially sampled and each of the JSDM were fit to all 5 realizations. Afterwards, the normalized  $([-1,1])$  covariance matrices were averaged over their absolute values and the root mean squared errors to the true association matrices were calculated.



---

# Supporting Information S4 for Chapter 5

## 1 Extending ACE to two-way interactions

ACE can be extended to  $n$ -dimensions to detect  $n$  way predictor interactions. Here, we extended ACEs to two dimensions to detect two-way predictor interactions by asking what the change is of  $\hat{f}(\cdot)$  when predictors  $x_m$  and  $x_k$  change together:

$$\text{ACE}_{mk} = \frac{\partial^2 \hat{f}(\mathbf{X})}{\partial x_m \partial x_k}$$

We can approximate  $\text{ACE}_{mk}$  with the finite difference method:

$$\begin{aligned} \text{ACE}_{mk} \approx & \frac{\hat{f}(x_1, x_2, \dots, x_m + h, x_k + h, \dots, x_j)}{2(h_m + h_k)} \\ & - \frac{\hat{f}(x_1, x_2, \dots, x_m - h, x_k + h, \dots, x_j)}{2(h_m + h_k)} \\ & - \frac{\hat{f}(x_1, x_2, \dots, x_m + h, x_k - h, \dots, x_j)}{2(h_m + h_k)} \\ & + \frac{\hat{f}(x_1, x_2, \dots, x_m - h, x_k - h, \dots, x_j)}{2(h_m + h_k)} \end{aligned} \tag{9.22}$$

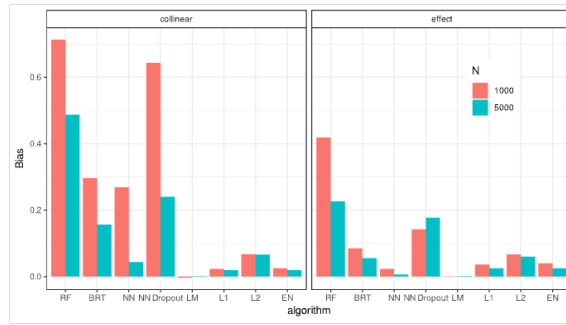
$h_m$  and  $h_k$  are set to  $0.1 \cdot sd(x_m)$  and  $0.1 \cdot sd(x_k)$ . All predictors are centered and standardized.

### Proof of concept simulations for inferring interactions

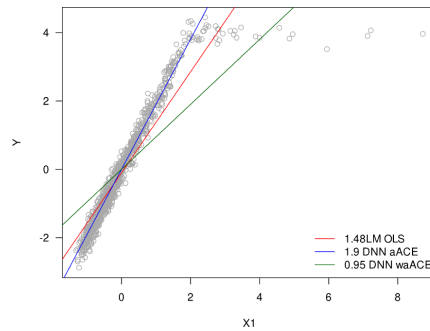
To test the ability of ML algorithms to identify predictor-predictor interactions, we repeated the proof-of-concept simulations, but with an interaction between  $X_1$  and  $X_2$ . The data generation model was  $Y \sim 1.0 \cdot X_1 + 1.0 \cdot X_5 + 1.0 \cdot (X_1 \cdot X_2) + \epsilon$  with  $\epsilon \sim N(0, 1.0)$ . We simulated two scenarios, in the first ("collinear")  $X_1$  and  $X_2$  were collinear (Pearson correlation factor = 0.9) and in the second without collinearity between the predictors.

We sampled 1000 and 5000 observations from each scenario. The ML algorithms (RF, BRT, NN, and NN with dropout) were fit to the data without predictor engineering the predictor interactions (because ML algorithms are known to be able to infer interactions automatically), while the regression algorithms (LM, l1, l2, and elastic-net) received all combinatorially possible predictor interactions as possible predictors. All effects were inferred using ACE. The bias was calculated for the interaction  $X_1 : X_2$ .

We found that for the ML algorithms (RF, BRT, and NN) NN showed the lowest for all scenarios (Fig. S4.1). Also collinearity increased the bias for the ML algorithms. No collinearity or more observations decreased the bias (Fig. S4.1). The regression models, LM, LASSO and Ridge regression, and elastic-net showed the lowest and in case of LM, no bias. However, we want to note here that the regression models received all possible predictor-predictor interactions as predictors while the ML algorithms had to infer the interactions on their own. With this in mind, the performance of the NN is surprising well, even competing with the penalized regression models. On the other hand, NN with dropout showed larger biases than BRT (Fig. S4.1).



**FIGURE S4.1:** Bias of proof of concept simulations in inferring two-way interactions between predictors. First panel shows results for simulations (200 repetitions) for 1000 and 5000 observations with collinear predictors (Pearson correlation factor = 0.9 between  $x_1$  and  $x_2$ ). Second panel shows results for simulations (200 repetitions) for 1000 and 5000 observations with without collinear. Red bars correspond to 1000 observations and blue bars to 5000 observations.



**FIGURE S4.2:** Simulation example with non-uniform sampled predictor  $X_1$  (log normal distributed). The red line is the effect estimated by a LM OLS. The blue line is the effect reported by an unweighted ACE from a NN. The green line is the effect reported by a weighted ACE from a NN.

### Weighted ACE

If the instances of a predictor  $x_j$  are not uniformly distributed, we propose to calculate a weighted  $wACE_k = \sum_{i=1}^N w_i ACE_{ik}$  with the  $w_i$  being, for example, the inverse probabilities of an estimated density function over the predictor space of  $x_k$ .

To demonstrate the idea of weighted ACE, we simulated a scenario with one predictor where the  $\beta_1 = 2$  for values of the predictor  $< 2$  and for the other predictor values  $\beta_1 = 0$  (Fig. S4.2). The predictor was sampled from a log-Normal distribution. We fitted a linear regression model and a NN on the data and compared the effect estimated by the LM, the unweighted ACE, and the weighted ACE.

The LM estimated an effect of 1.48, the unweighted ACE was 1.95, and the weighted ACE was 1.48 (Fig. S4.2).

---

## 2 Boosting and regression trees

### Unbiasedness

Random forest (RF) and boosted regression trees (BRT) showed biased effect estimates in both scenarios, with and without collinearity, raising the question of whether the bias is caused by the boosting/bagging or the regression trees themselves. For RF, we know that the observed spillover effect is caused by the random subsampling (mtry parameter) in the algorithm, which explains the bias.

For BRT, however, it is unclear what is causing the bias (boosting or regression trees) because each member in the ensemble is always presented with all predictors (at least with the default hyperparameters, the BRT implementation in xgboost has options to use bootstrap samples for each tree and also subsamples of columns in each tree (or node), see (CHEN and GUESTRIN, 2016)).

To understand how boosting and regression trees affect effect estimates, we simulated three different scenarios (Fig. S4.3, first column) without collinearity (Fig. S4.3a) and with collinearity (Fig. S4.3a, b) (we sampled 1000 observations from each data generating model (Fig. S4.3, first column) and estimated effects using ACE (500 repetitions)).

We found that the regression tree (RT) is unable to estimate unbiased effects (Fig. S4.3), regardless of the presence or absence of collinearity or the complexity of the RT (depth of the regression trees). Without collinearity, effects in regression trees were biased toward zero, less so with higher complexity (Fig. S4.3). With collinearity, there was a small spillover effect for the RT with high complexity (Fig. S4.3b) to the collinear zero effect ( $X_2$ ), similar to an l2 regularization. When the collinear predictor ( $X_2$ ) had an effect (Fig. S4.3c), we found a stronger absolute bias for the smaller of the two collinear effects ( $X_2$ ), confirming our expectation that RTs show a greedy effect. This greedy behavior was particularly strong for the low complexity RT (Fig. S4.3c).

To answer the question of how boosting affects the greediness and spillover effects of RT, we first investigated the behavior of a linear booster because of the well-known behavior of OLS under collinearity. And indeed, we found that the linear booster was unbiased in all three scenarios (compare LM and linear booster in Fig. S4.3), showing that boosting itself can produce unbiased effects.

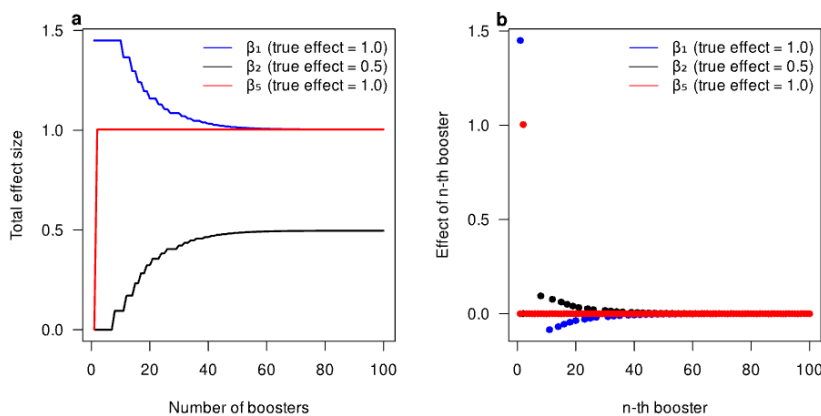
Now, comparing the vanilla BRTs with low and high complexity (depth of individual trees) with the linear booster and the RTs, we found similar biases as for the RTs, in terms of spillover with a collinear zero effect and the greediness effect in the presence of a weaker collinear effect (Fig. S4.3).

### Understanding boosting

Intuitive boosting shouldn't work because it's basically a regression of residuals. That is, and in the case of collinearity, the stronger of two collinear predictors in the first model would absorb the effect of the weaker second predictor that, for example, causes the omitted variable bias (the effect of the missing confounder is absorbed by the collinear effect).



**FIGURE S4.3:** Bias on effect estimates for different ML algorithms (LM = linear regression model (OLS), RT LC = regression tree with low complexity (depth), RT HC = regression tree with high complexity, Linear Booster, Tree Booster LC = tree booster with low complexity, Tree Booster HC = tree booster with high complexity) in three different simulated causal scenarios (a, b, and c). Sample sizes are so large that stochastic effects can be excluded (1000 observations). Effects of the ML models were inferred using average conditional effects. Row a) shows results for simulations with uncorrelated predictors with the true effect sizes. Row b) shows the results for simulations with  $X_1$  and  $X_2$  being strongly correlated (Pearson correlation factor = 0.9) but only  $X_1$  has an effect on  $Y$  (mediator) and row c) shows the results for  $X_1$  and  $X_2$  being strongly correlated (Pearson correlation factor = 0.9) with  $X_1$  and  $X_2$  having effects on  $Y$  (confounder scenario).

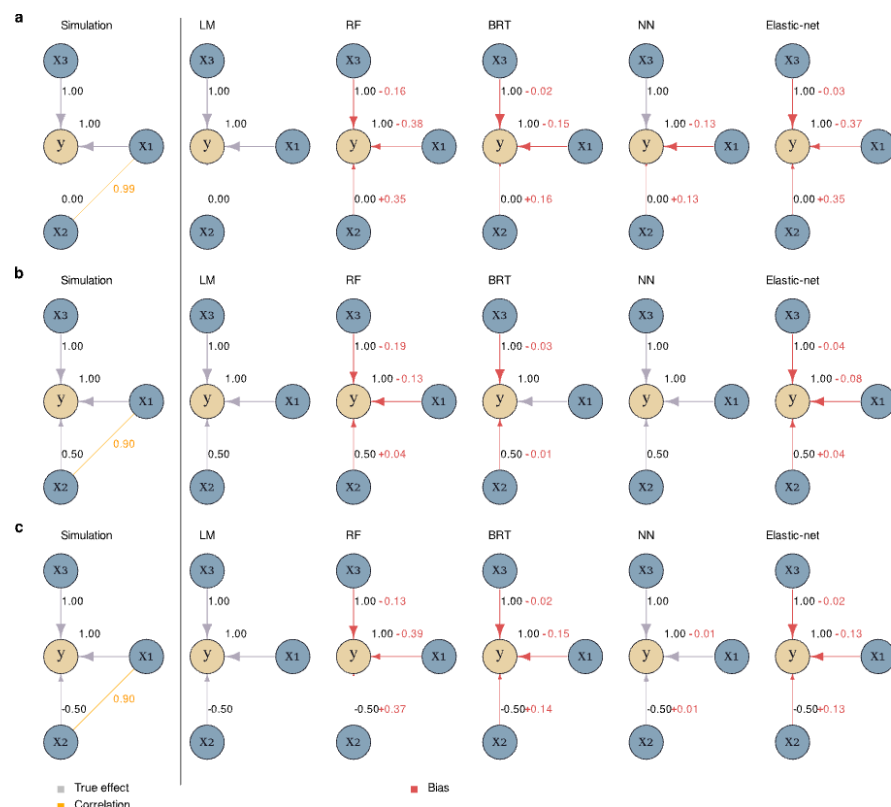


**FIGURE S4.4:** Changes of effects within boosting. (A) shows the total effect of ensemble (linear booster) until the  $n$ -th ensemble member. (B) shows the effects of the  $n$ -th ensemble member.  $X_1$  and  $X_2$  were correlated (Pearson correlation factor = 0.9).

Looking at the development of the total effect within a linear booster model (Fig. S4.4a), we found that the first members of the ensemble absorb the effect of the collinear effect ( $\beta_1$  absorbed  $\beta_2$ , Fig. S4.4a), but as members are added to the ensemble, the collinear effect  $\beta_2$  slowly recovers the effect of the stronger collinear effect until both are at their correct effect estimate (Fig. S4.4a). This retrieval works by reversing the sign of each member's effect, so that  $\beta_1$ , which initially has an effect of 1.5 (because it absorbed the effect of  $\beta_2$ ), has small negative effects in subsequent trees, while  $\beta_2$ , which is initially estimated at 0, has small positive effects (Fig. S4.4b).

### 3 Proof of concept - Additional results

To better understand the ability of ML algorithms in learning unbiased effects, we tested additional scenarios (Fig. S4.5, first column).

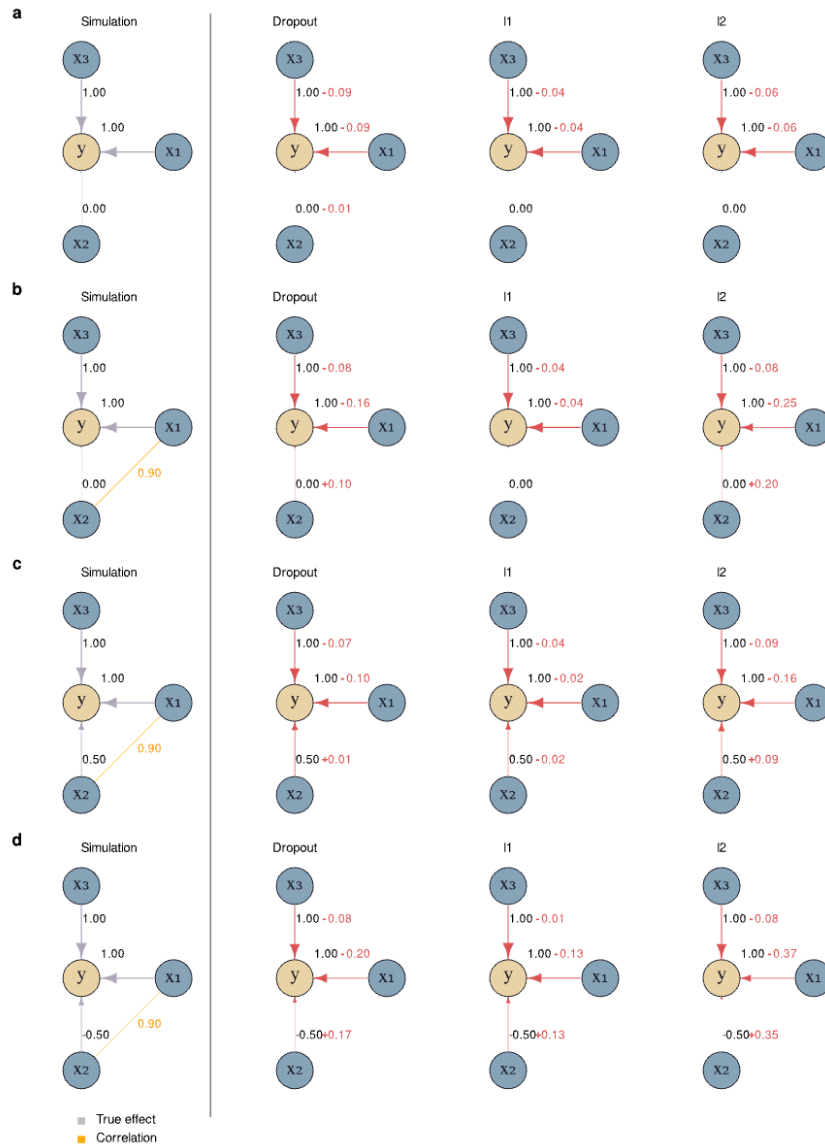


**FIGURE S4.5:** Bias on effect estimates for different ML algorithms in three different simulated causal simulations (a, b, and c). Sample sizes are so large that stochastic effects can be excluded (1000 observations). Effects of the ML models were inferred using average conditional effects. Row a) shows the results for simulations with  $X_1$  and  $X_2$  being strongly correlated (Pearson correlation factor = 0.99) but only  $X_1$  has an effect on  $y$ . Row b) shows results for simulations with with predictors (Pearson correlation factor = 0.5) with effect sizes ( $X_1$ : 1.0,  $X_2$ : 0.5,  $X_3$ : 1.0) and row c) shows results for simulations with with predictors (Pearson correlation factor = 0.5) with effect sizes ( $X_1$ : 1.0,  $X_2$ : -0.5,  $X_3$ : 1.0)

We found that NN cannot separate extreme collinear effects as the OLS (Fig. S4.5a) which, however, may improve with additional observations.

## Additional models

To understand the different effects of regularization in NN (dropout), LASSO regression, and Ridge regression, we tested these models on our theoretical scenarios (Fig. S4.6, first column).



**FIGURE S4.6:** Bias on effect estimates for different ML algorithms in two different simulated causal simulations (a and b). Sample sizes are so large that stochastic effects can be excluded. Effects of the ML models were inferred using average conditional effects. Row a) shows results for simulations with with predictors (Pearson correlation factor = 0.5) with effect sizes ( $X_1$ : 1.0,  $X_2$ : -0.5,  $X_3$ : 1.0). Row b) shows the results for simulations with  $X_1$  and  $X_2$  being strongly correlated (Pearson correlation factor = 0.99) but only  $X_1$  has an effect on  $y$ .

Dropout has a negative effect on the ability to separate collinear effects in NN (Fig. S4.6) while also LASSO and Ridge (as expected) affect negatively the ability to separate collinear effects (Fig. S4.6).



---

## 4 Hyperparameter tuning

We performed a hyperparameter search to check if and how hyperparameters influence differently or equally effect estimates and the prediction error, so does a model tune after the prediction error has biased effects? For that, we created simulation scenarios with 50, 100, 600, and 2000 observations and 100 predictors with effects ( $\beta_i, i = 1, \dots, 100$ )  $\beta_1 = 1.0$ , and  $\beta_2$  to  $\beta_3$  were equally spaced between 0.0 to 1.0 so that  $\beta_2 = 0.0$  and  $\beta_{100} = 1.0$ .

Predictors were sampled from a multivariate normal distribution and all predictors were randomly correlated (Variance-covariance matrix  $\Sigma$  was sampled from a LKJ-distribution with  $\eta = 2.0$ ).

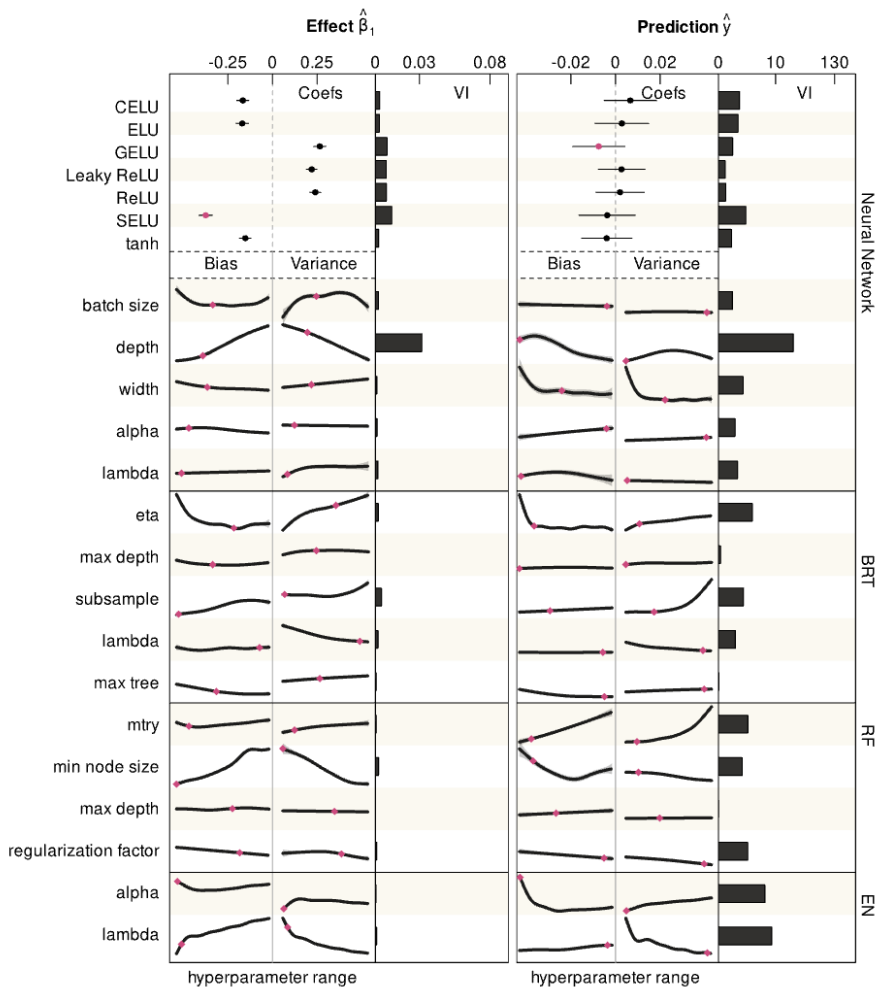
1,000 combinations of hyper-parameters were randomly drawn (Table S4.1). For each draw of hyperparameters, the data simulation and model fitting was repeated 20 times. Effect sizes of  $X_1$  and  $X_2$  were recorded (for each hyperparameter combination and for each repetition). Moreover, bias, variance, and mean square error (MSE) were recorded for the predictions on a holdout of the same size as the training data.

**TABLE S4.1:** Overview over hyper-parameters for Neural Network, Boosted Regression Tree, and Random Forest

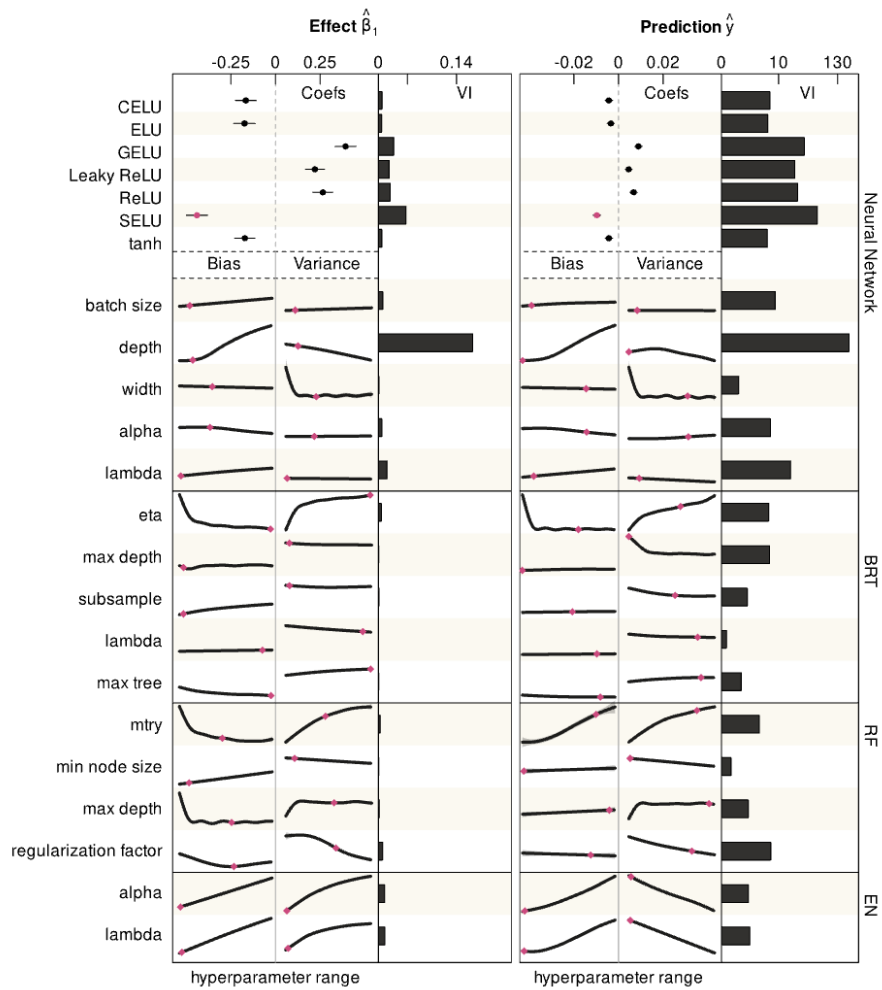
Algorithm	Hyper-parameter	Range
Neural Network	activation function	[relu, leaky_relu, tanh, selu, elu, celu, gelu]
	depth	[1, 8]
	width	[2, 50]
	batch size (sgd)	[1, 100] in percent
	lambda	[2.65e-05, 0.16]
	alpha	[0, 1.0]
Boosted Regression Tree	eta	[0.01, 0.4]
	max depth	[2, 25]
	subsample	[0.5, 1]
	max tree	[30, 125]
	lambda	[1, 20]
Random Forest	mtry	[0, 1] in percent
	min node size	[2, 70]
	max depth	[2, 50]
	regularization factor	[0, 1]
Elastic net	alpha	[0, 1.0]
	lambda	[0, 1.0]

## Results hyperparameter tuning

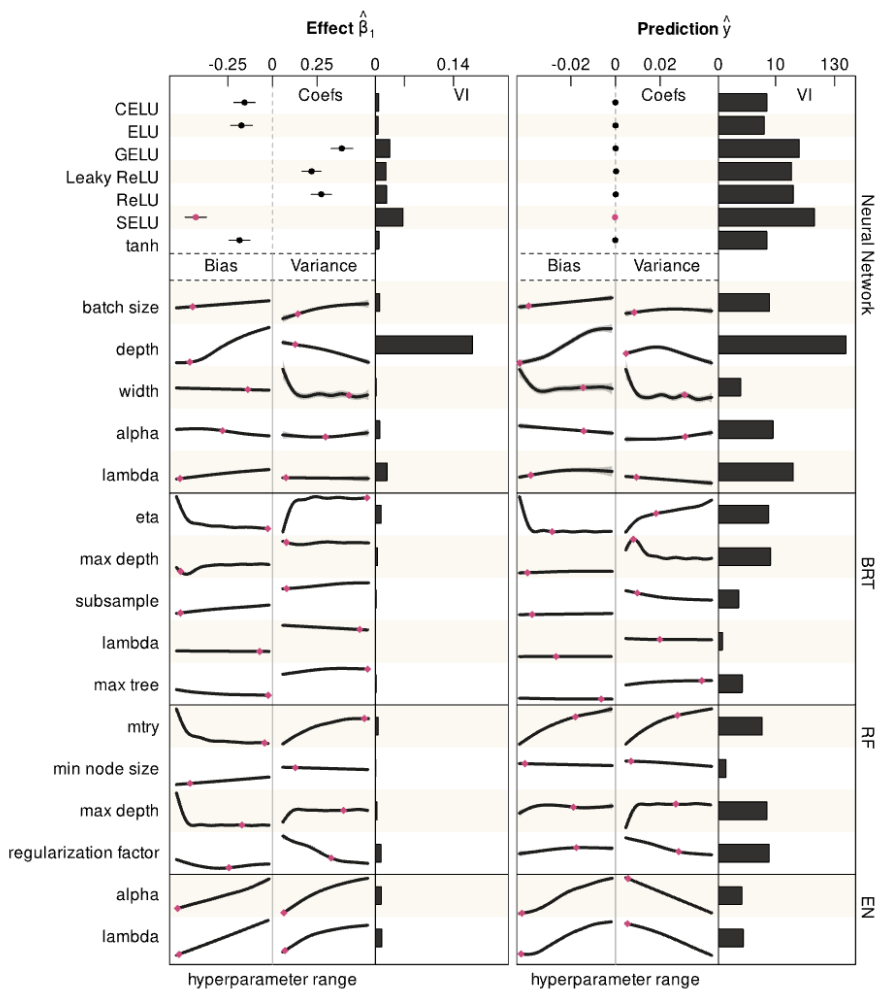
As described in the main text, we analyzed the effects of the hyperparameters on the different errors using GAMs and variable importance of random forest (Fig. S4.7, S4.8, S4.9).



**FIGURE S4.7:** Results of hyperparameter tuning for Neural Networks (NN), Boosted Regression Trees (BRT), Random Forests (RF), and Elastic Net (EN) for 50 observations with 100 predictors. The influence of the hyperparameters on effect  $\hat{\beta}_1$  (bias, variance, and MSE)(true simulated effect  $\beta_1 = 1.0$ ) and the predictions,  $\hat{y}$  of the model (bias, variance, and MSE) were estimated by a multivariate generalized additive model (GAM). Categorical hyperparameters (activation function in NN) were estimated as fixed effects. The responses (bias, variance, MSE) were centered so that the categorical hyperparameters correspond to the intercepts. The variable importance of the hyperparameters was estimated by a random forest with the MSE of the effect  $\hat{\beta}_1$  (first plot) or the prediction  $\hat{y}$  (second plot) as the response. Red dots correspond to the best predicted set of hyperparameters (based on a random forest), in the first plot for the minimum MSE of the effect  $\hat{\beta}_1$  and in the second plot for the minimum MSE of the predictions  $\hat{y}$ .



**FIGURE S4.8:** Results of hyperparameter tuning for Neural Networks (NN), Boosted Regression Trees (BRT), Random Forests (RF), and Elastic Net (EN) for 600 observations with 100 predictors. The influence of the hyperparameters on effect  $\hat{\beta}_1$  (bias, variance, and MSE)(true simulated effect  $\beta_1 = 1.0$ ) and the predictions,  $\hat{y}$  of the model (bias, variance, and MSE) were estimated by a multivariate generalized additive model (GAM). Categorical hyperparameters (activation function in NN) were estimated as fixed effects. The responses (bias, variance, MSE) were centered so that the categorical hyperparameters correspond to the intercepts. The variable importance of the hyperparameters was estimated by a random forest with the MSE of the effect  $\hat{\beta}_1$  (first plot) or the prediction  $\hat{y}$  (second plot) as the response. Red dots correspond to the best predicted set of hyperparameters (based on a random forest), in the first plot for the minimum MSE of the effect  $\hat{\beta}_1$  and in the second plot for the minimum MSE of the predictions  $\hat{y}$ .



**FIGURE S4.9:** Results of hyperparameter tuning for Neural Networks (NN), Boosted Regression Trees (BRT), Random Forests (RF), and Elastic Net (EN) for 2000 observations with 100 predictors. The influence of the hyperparameters on effect  $\hat{\beta}_1$  (bias, variance, and MSE)(true simulated effect  $\beta_1 = 1.0$ ) and the predictions,  $\hat{y}$  of the model (bias, variance, and MSE) were estimated by a multivariate generalized additive model (GAM). Categorical hyperparameters (activation function in NN) were estimated as fixed effects. The responses (bias, variance, MSE) were centered so that the categorical hyperparameters correspond to the intercepts. The variable importance of the hyperparameters was estimated by a random forest with the MSE of the effect  $\hat{\beta}_1$  (first plot) or the prediction  $\hat{y}$  (second plot) as the response. Red dots correspond to the best predicted set of hyperparameters (based on a random forest), in the first plot for the minimum MSE of the effect  $\hat{\beta}_1$  and in the second plot for the minimum MSE of the predictions  $\hat{y}$ .

### Optimal hyperparameters

The hyperparameters were chosen based on the lowest MSE for the predictive performance of the models (Table S4.2) and the lowest MSE for the effect ( $\beta_1$ ) on  $X_1$  (Table S4.3). The selection of the best hyperparameters was done by first fitting a random forest (default parameters) with the MSE as response and the hyperparameters as predictors, and then using the set of hyperparameters that predicted the lowest MSE.

**TABLE S4.2:** Best predicted set of hyperparameter for ML algorithms (tuned after MSE of predictions)

Algorithm	Hyperparameter	n = 50	n = 100	n = 600	n = 2000
NN	activations	celu	selu	selu	selu
	sgd	0.944	0.348	0.098	0.098
	depth	1	1	1	1
	width	24	20	35	35
	alpha	0.939	0.821	0.693	0.693
	lambda	0.003	0.02	0.019	0.019
BRT	eta	0.072	0.126	0.245	0.147
	max_depth	2	2	2	4
	subsample	0.666	0.511	0.77	0.57
	lambda	9.073	8.888	8.21	4.556
	max_tree	117	109	110	114
RF	mtry	0.129	0.466	0.792	0.603
	min.node.size	12	2	3	6
	max.depth	21	19	47	30
	regularization.factor	0.914	0.874	0.736	0.615
EN	alpha	0.007	0.008	0.025	0.025
	lambda	0.286	0.028	0.006	0.006

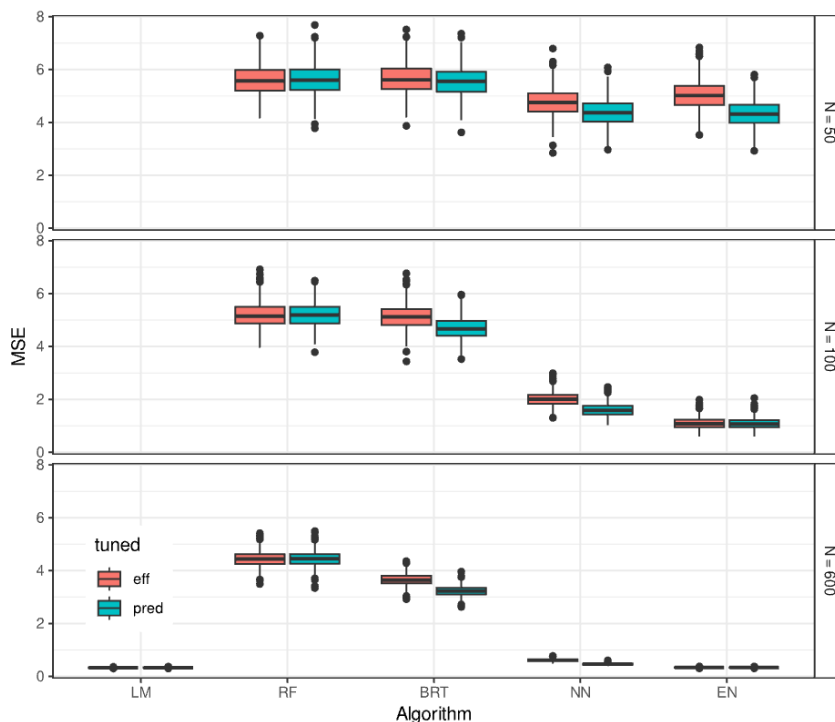
**TABLE S4.3:** Best predicted set of hyperparameter for ML algorithms (tuned after MSE of effect  $X_1$ )

Algorithm	Hyperparameter	n = 50	n = 100	n = 600	n = 2000
NN	activations	selu	selu	selu	selu
	sgd	0.391	0.395	0.112	0.175
	depth	3	3	2	2
	width	18	40	19	39
	alpha	0.135	0.613	0.332	0.498
	lambda	0.009	0.011	0.002	0.006
BRT	eta	0.252	0.327	0.393	0.393
	max_depth	11	17	3	3
	subsample	0.514	0.584	0.523	0.523
	lambda	9.051	7.779	9.053	9.053
	max_tree	71	102	124	124
RF	mtry	0.137	0.926	0.462	0.952
	min.node.size	2	4	9	12
	max.depth	31	29	29	36
	regularization.factor	0.683	0.894	0.587	0.566
EN	alpha	0.011	0	0.011	0.011
	lambda	0.016	0.018	0.009	0.009

## 5 Additional results for data-poor scenarios

### Prediction error of scenarios

Fig. S4.10 shows the MSE of the predictions on the holdouts for the different ML algorithms and different number of observations of the data-poor scenarios (see main text).

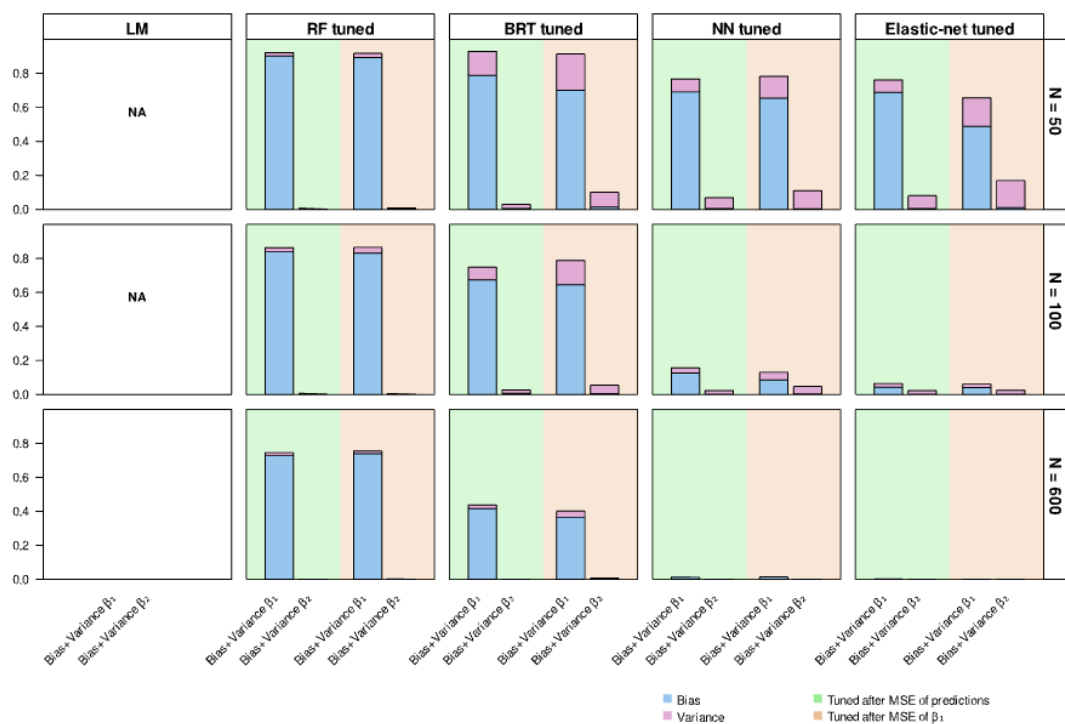


**FIGURE S4.10:** Prediction error (mean square error, MSE) of data poor simulations with optimal hyperparameters either tuned after the best MSE of the effect size (red) or the best MSE of the prediction error (blue).

## 6 Data-poor scenarios without collinearity

### Bias and variance of effects

To assess the effect of collinearity on the data-poor simulations, we repeated the scenarios but without collinearity.  $\Sigma$  which was used in the sampling process of the predictor matrix (multivariate normal distribution) was set to the identity matrix. While it is not ideal, we used the best hyperparameters (Table S4.2, Table S4.3) which were tuned for the collinear scenarios.

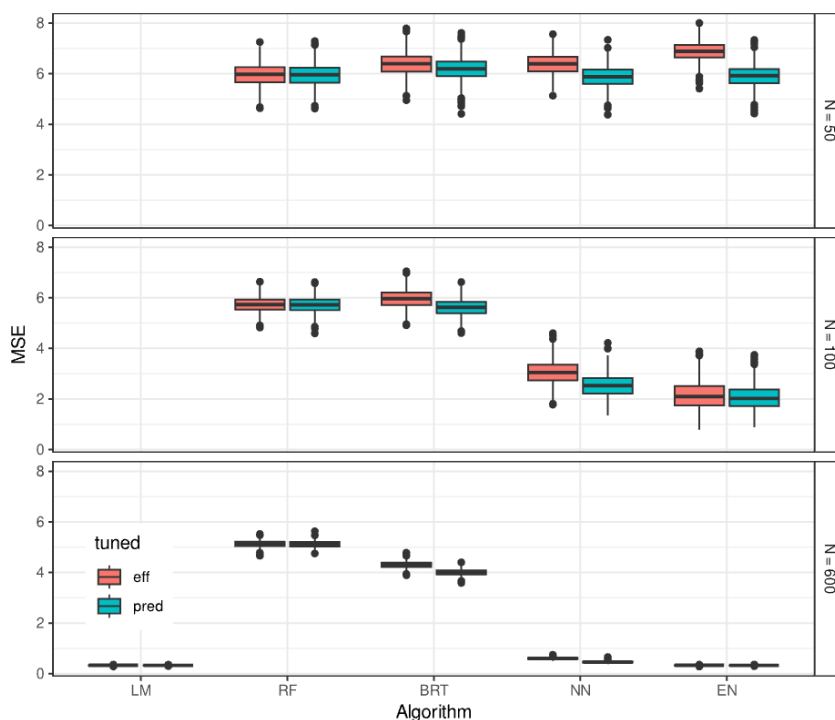


**FIGURE S4.11:** Bias and variance of estimated effects in data-poor situations.  $N = 50, 100,$  and  $600$  observations of  $100$  uncorrelated predictors were simulated. True effects in the data generating model were  $\beta_1=1.0, \beta_2=0.0,$  and the other  $98$  effects were equally spaced between  $0$  and  $1$ . Models were fitted to the simulated data ( $1000$  replicates) with the optimal hyperparameters (except for LM, which doesn't have hyperparameters). Hyperparameters were selected based on the minimum MSE of  $(\hat{\beta}_1)$  (green) or the prediction error (based on  $\hat{y}$ ) (red). Bias and variance were calculated for  $\hat{\beta}_1$  and  $\hat{\beta}_2$ . Effects  $\hat{\beta}_i$  for  $i = 1, \dots, 100$  were approximated using ACE.

We found similar results as for data-poor scenarios with collinearity (Fig. S4.11). NN and elastic-net show the lowest errors and strongest increase in those errors with increasing number of observations (Fig. S4.11).

### Prediction error of scenarios

Fig. S4.12 shows the prediction errors for the ML algorithms for the data-poor simulations without collinearity. We found similar results as for the data-poor simulations with collinearity.



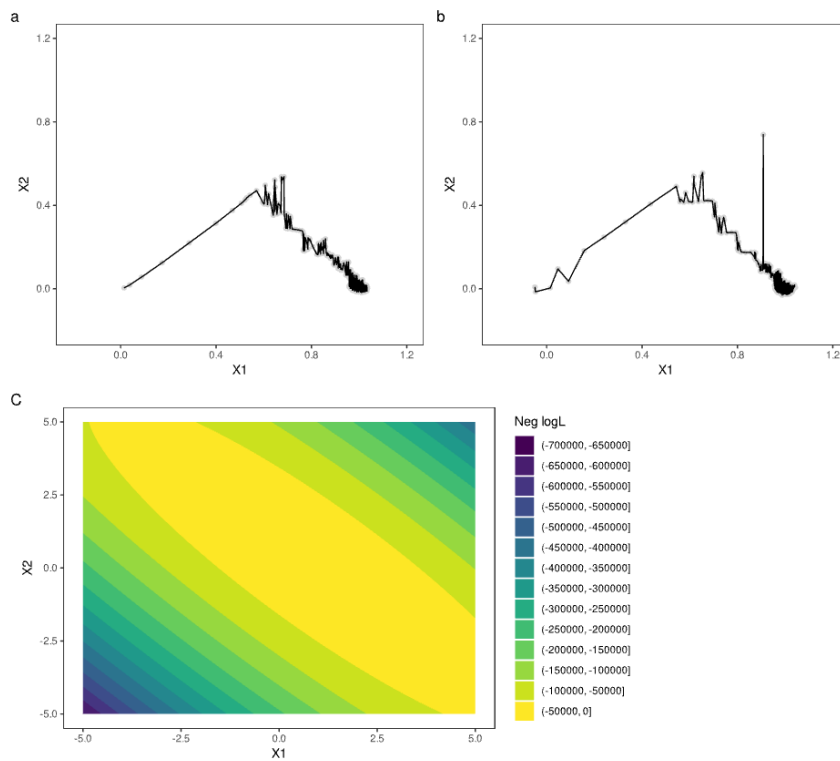
**FIGURE S4.12:** Prediction error (mean square error, MSE) of data poor simulations with optimal hyperparameters either tuned after the best MSE of the effect size (red) or the best MSE of the prediction error (blue).

## 7 Learning in neural networks

To understand the internal learning of neural networks, we trained neural networks of two different sizes (3 layers of 50 units and 3 layers of 500 units) on a simple collinear scenario ( $Y \sim 1.0 \cdot X_1 + 0.0 \cdot X_2 + \epsilon$ ,  $\epsilon \sim N(0, 0.3)$ ;  $X_1$  and  $X_2$  were collinear (Pearson correlation factor = 0.9)) and calculated the ACE after each batch optimization step.

We found that the estimates of the batch effect were initially estimated to be around 0 (Fig. S4.13 A, B), probably due to the initialization of the neural networks, which resembles a shrinkage behavior (weights have to be moved away from 0 step by step in the gradient descent). After this initialization phase, both estimates are within the expected negative log-likelihood surface of OLS (Fig. S4.13C) and are estimated over the training period to the correct estimates ( $X_1 = 1.0$  and  $X_2 = 0.0$ ).

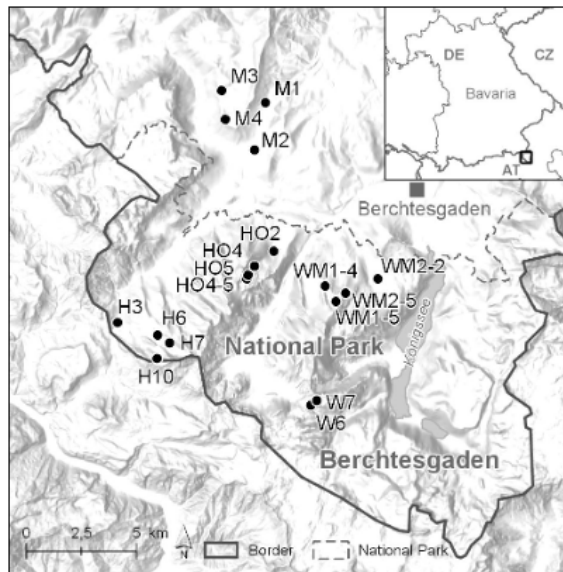




**FIGURE S4.13:** Learning neural networks. Neural networks were trained on simulated data (1000 observations) with 5 predictors,  $X_1$  has a linear effect on  $Y$ , and  $X_2$  is collinear with  $X_1$  (Pearson correlation factor = 0.9). The ACE was computed after each optimization step (i.e., after each batch in stochastic gradient descent) (20 repetitions). Panels A and B show the evolution of the effects for  $X_1$  and  $X_2$  (true effects:  $X_1 = 1.0$  and  $X_2 = 0.0$ ). Panel A shows the results for a neural network with 50 units in each of the 3 hidden layers, while Panel B shows the results for a neural network with 500 units in each of the 3 hidden layers. Panel C shows the negative log likelihood surface for the corresponding OLS.



## Supporting Information S5 for Chapter 7



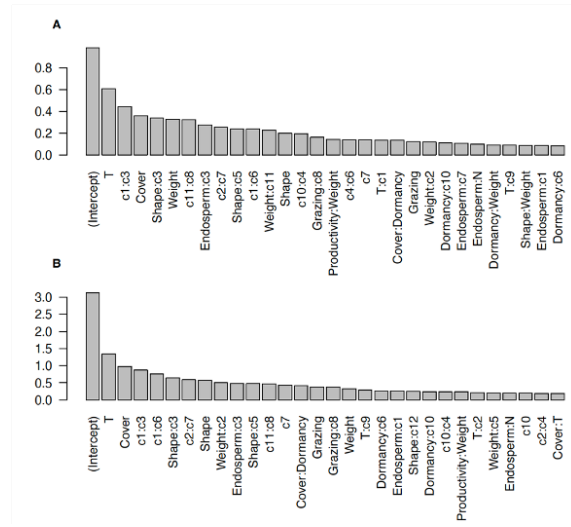
**FIGURE S5.1:** Location of the study sites (n=18) along elevational gradient in the Bavarian Alps, Germany (see Table S5.1 for detailed environmental characteristics). Different letters indicate sampling ‘subregions’: H – ‘Hochkalter’, HO – ‘Hochkalter Ost’, M – Mordaualm, W – ‘Wimbachtal’, WM – ‘Watzmann’.

**TABLE S5.1:** Results of t-tests (p-value) comparing the random forest (RF) and generalized linear model for the different predictor groups for presence/absence of SSB.

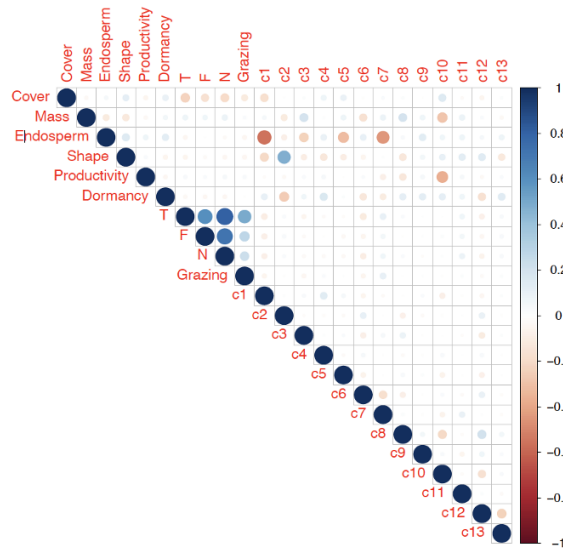
Scenario	lm	lm-SE	RF	RF-SE	p-value
All	0.769	0.022	0.861	0.010	0.003
Phylo	0.694	0.019	0.836	0.013	0.000
Seed	0.615	0.015	0.833	0.010	0.000
Env	0.722	0.025	0.697	0.015	0.405
Phylo.Seed	0.743	0.026	0.837	0.012	0.007
Phylo.Env	0.740	0.021	0.852	0.012	0.001
Env.Seed	0.738	0.025	0.823	0.012	0.010

**TABLE S5.2:** Results of t-tests (p-value) comparing the random forest (RF) and linear regression model for the different predictor groups for SSB density.

Scenario	lm	lm-SE	RF	RF-SE	p-value
All	0.199	0.026	0.416	0.019	0.000
Phylo	0.091	0.022	0.317	0.026	0.000
Seed	0.015	0.007	0.337	0.023	0.000
Env	0.119	0.014	0.124	0.018	0.833
Phylo.Seed	0.132	0.026	0.324	0.028	0.000
Phylo.Env	0.157	0.022	0.383	0.022	0.000
Env.Seed	0.172	0.030	0.329	0.020	0.001



**FIGURE S5.2:** Absolute effect estimates of generalized linear model (A) for presence/absence of SSB and of linear model (B) for SSB density. Only the 30 largest effect estimates are shown here. The elastic-net approach was applied to regularize effects.



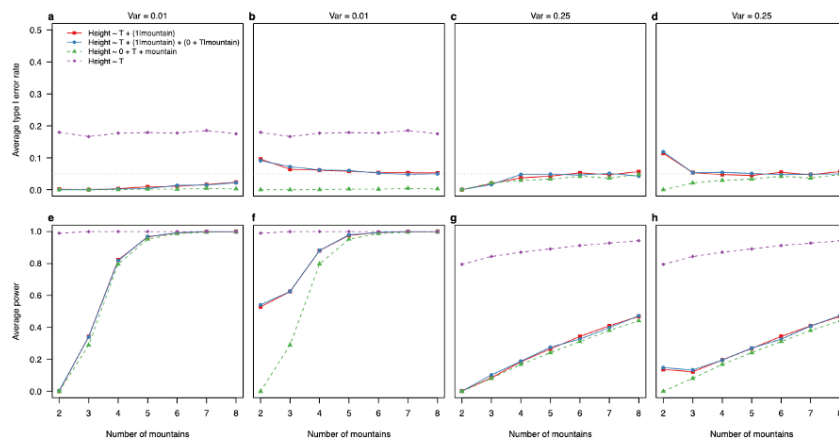
**FIGURE S5.3:** Factors of Pearson correlation between predictors for SSB.

# Supporting Information S6 for Chapter 8

## 1 Additional Results for linear-mixed effect models

### Results for the intercept

We calculated type I error rate and power for the intercept estimates of Scenario A and B (Fig. S6.1, Fig. S6.2) and we found similar patterns as for the slope estimates (Fig. 8.3, Fig. 8.4).

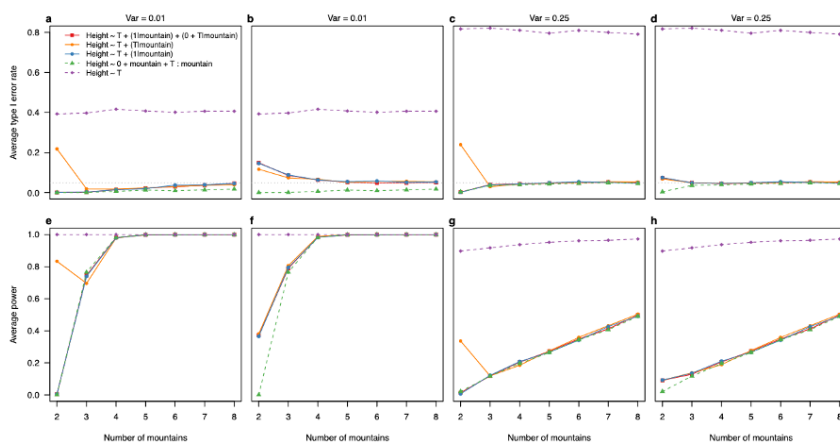


**FIGURE S6.1:** Average type I error rates and average power for the intercept in linear fixed and mixed-effects models fitted to simulated data with 2-8 mountains (random intercept for each mountain - Scenario A) and with 50 observations per mountain. For each scenario, 5000 simulations and models were tested. (a, b, e, f) show results for simulated data with a variance of 0.01 in the random effects. (c, d, g, h) show results for simulated data with a variance of 0.25 in the random effects. (a, c, e, g) show results for mixed-effects models only from datasets in which mixed-effects models converged without presenting singular fit problems and (b, d, f, h) results for mixed-effects models for all datasets. Results for fixed-effects (a-h) model are from all datasets. (a-d) the dotted line represents the 5% alpha level.

### Variance estimates and singular fits

We found that singular fits occurred more often in mixed-effect models when using MLE compared to REML (Table S6.1). The rate of singular fits decreased with increasing number of groups (Table S6.1).

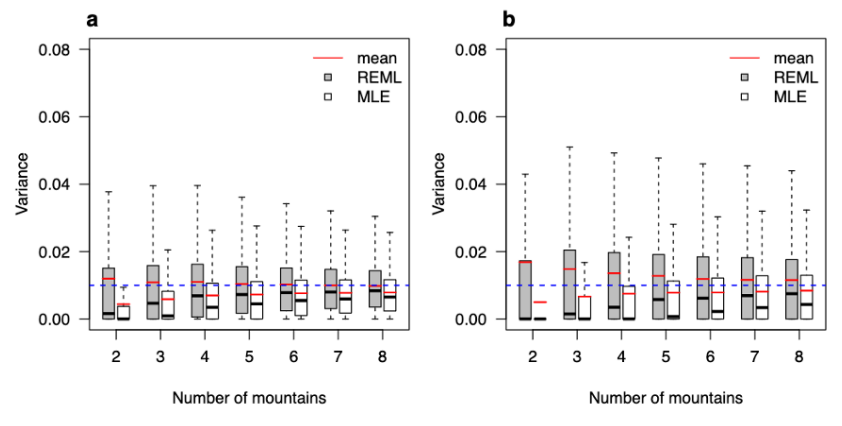
Additional to the rate of singular fits a direct comparison of the variance estimates using REML and MLE is necessary to compare their performance. Using MLE for linear mixed-effect models led to stronger towards zero biased estimates compared to using REML (Fig. S6.3). Estimates for balanced and unbalanced data do not differ within REML and MLE (Fig. S6.4, S6.5)



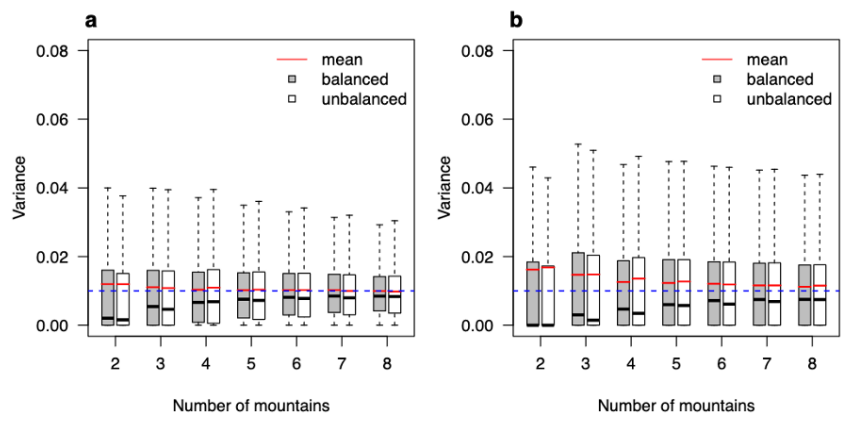
**FIGURE S6.2:** Average type I error rates and average power for the intercept in linear (mixed-effects) models fitted to simulated data with 2-8 mountains for scenario B (random intercept and random slope for each mountain range). For each scenario, 5,000 simulations and models were tested. (a, b, e, f) show results for simulated data with a variance of 0.01 in the random effects. (c, d, g, h) show results for simulated data with a variance of 0.25 in the random effects. (a, c, e, g) show results for mixed-effects models only from datasets in which mixed-effects models converged without presenting singular fit problems and (b, d, f, h) results for mixed-effects models for all datasets. Results for fixed-effects (a-h) model are from all datasets. In (a-d) the dotted line represents the 5% alpha level.

**TABLE S6.1:** Rate of models with singular fit convergence problem in lme4 when using maximum likelihood (MLE) and restricted maximum likelihood (REML) fitting algorithms. Notice that for GLMMs in lme4, REML is not implemented.

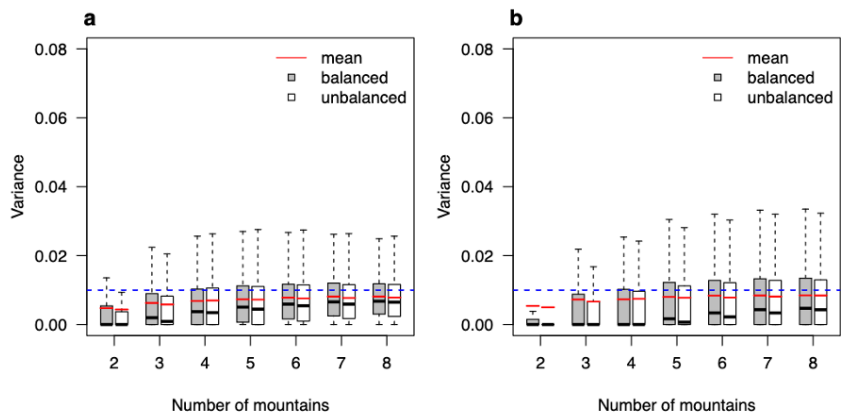
Number of groups	LMM		GLMM
	REML	MLE	MLE
2	77%	92%	95%
3	64%	80%	89%
4	55%	70%	84%
5	45%	61%	81%
6	40%	54%	78%
7	36%	48%	76%
8	32%	43%	72%



**FIGURE S6.3:** Variance estimates of the random intercepts (a) and random slopes (b) for linear mixed-effect models (LMM) fitted to simulated data with 2-8 numbers of artificial mountain ranges. For each scenario, 5,000 simulations and models were tested. The blue line represents the true variance used in the simulation (0.01). The grey boxes show results for the models fitted by restricted maximum likelihood estimation (REML) and the white boxes shows results for the models fitted by maximum likelihood estimation (MLE).



**FIGURE S6.4:** Variance estimates of the random intercepts (a) and random slopes (b) for linear mixed-effect models (LMM) fitted to simulated data with 2-8 numbers of artificial mountain ranges using REML. For each scenario, 5,000 simulations and models were tested. The blue line represents the true variance used in the simulation (0.01). The grey boxes show the results for the models with unbalanced data (number of observation) among mountains and the white boxes shows results for the models fitted with balanced data among mountains.



**FIGURE S6.5:** Variance estimates of the random intercepts (a) and random slopes (b) for linear mixed-effect models (LMM) fitted to simulated data with 2-8 numbers of artificial mountain ranges using MLE. For each scenario, 5,000 simulations and models were tested. The blue line represents the true variance used in the simulation (0.01). The grey boxes show the results for the models with unbalanced data (number of observation) among mountains and the white boxes shows results for the models fitted with balanced data among mountains.

## 2 Results for generalized linear-mixed effect models

If we would have also hypothesized that higher temperatures increase the reproductive success (either yes or no) of a plant species (H2), we would have to test also generalized mixed-effect models. To do so, we also simulated an unbalanced study design for this hypothesis with 2-8 mountains from a varying number of plants for each mountain (H2: expected range between 40-360 plants per mountain) while keeping the overall number of plants constant along altitudinal transects.

Again, we simulated 5000 datasets for each case. We used the inverse logit link function and sampled from a binomial distribution. We used models GLMs and GLMMs to fit the models to the simulated data with binomial distribution.

We found similar patterns as for the LM and LMMs in the main text. For scenario A, the number of mountains didn't affect the type I error rates or the statistical power, regardless of if the singular fits were included in the results for the mixed-effect models or not (Fig. S6). Here, we didn't test the overparametrized model or higher variances (larger effect sizes) in the simulated random intercepts.

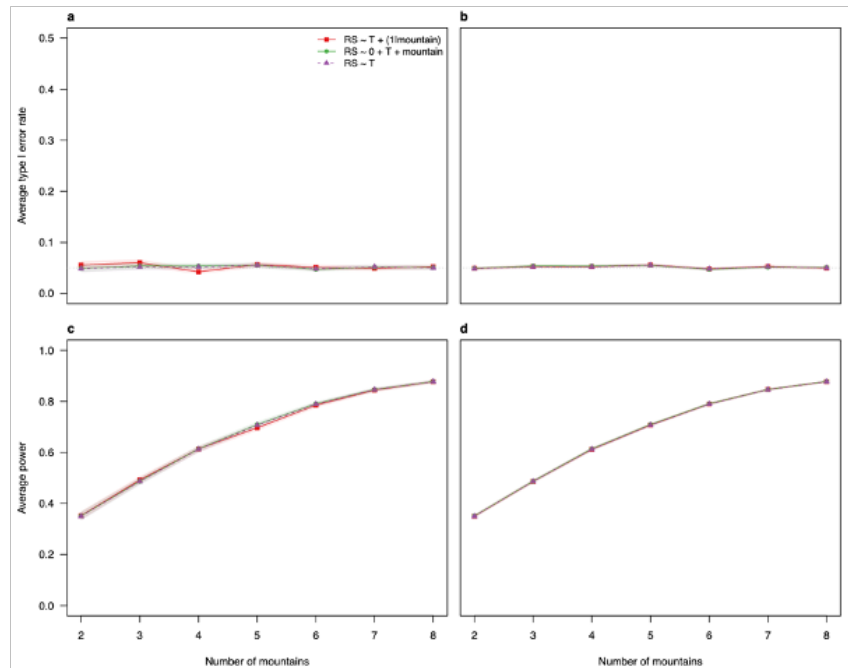
For scenario B, the overall pattern was again very similar to the findings for the LM and LMMs (Fig. 8.4). Type I error rates were robust to the number of the levels in the grouping variable (Fig. S6.7a-d), but the rates for all models were affected by the variance of the random effects (the size of random effects), and the mixed-effect models showed increased type I error rates when the singular fits were included (Fig. S6.7b, d).

The fixed-effect model without the grouping variable showed for low variance in the random effects close to the nominal level average type I error rates. However, for high variances, the average type I error rates were highly increased.

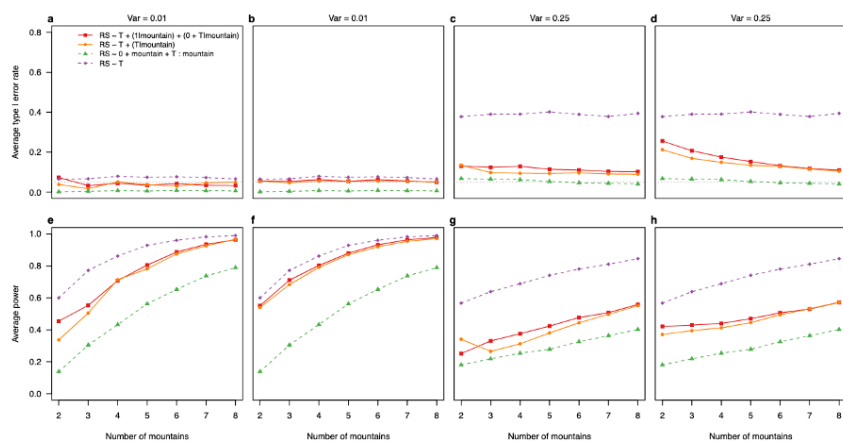
One notable difference to LM and LMMs is that the overall power of the mixed-effect models was always higher than when modeling the grouping variable as a fixed effect (Fig. S6.7e-h). This



difference in power was here higher than for LMs and LMMs (Fig. 8.4).



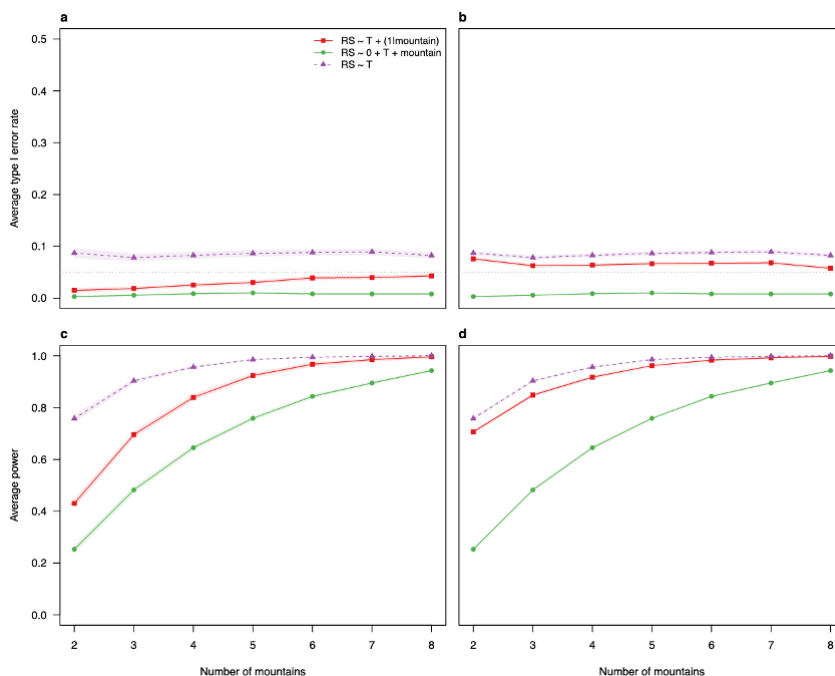
**FIGURE S6.6:** Average type I error rates and average power for generalized linear fixed and mixed-effect models fitted to simulated data with 2-8 mountains (random intercept for each mountain - Scenario A) and with 200 observations per mountain. For each scenario, 5000 simulations and models were tested. (a, c) show results for mixed-effects models only from datasets in which mixed-effects models converged without presenting singular fit problems and (b, d) results for mixed-effects models for all datasets. Results for fixed-effects (a-d) model are from all datasets. (a-d) the dotted line represents the 5% alpha level.



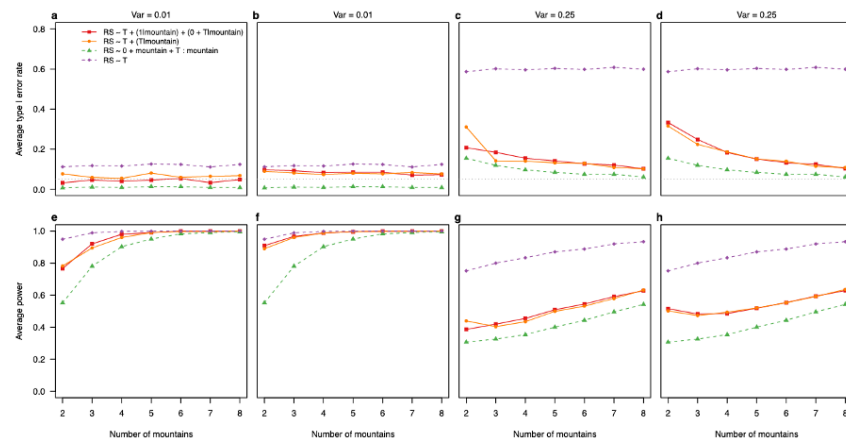
**FIGURE S6.7:** Average type I error rates and average power for generalized linear (mixed-effect) models fitted to simulated data with 2-8 mountains for scenario B (random intercept and random slope for each mountain range). For each scenario, 5,000 simulations and models were tested. (a, b, e, f) show results for simulated data with a variance of 0.01 in the random effects. (c, d, g, h) show results for simulated data with a variance of 0.25 in the random effects. (a, c, e, g) show results for mixed-effects models only from datasets in which mixed-effects models converged without presenting singular fit problems and (b, d, f, h) results for mixed-effects models for all datasets. Results for fixed-effects (a-h) model are from all datasets. In (a-d) the dotted line represents the 5% alpha level.

We calculated the statistical properties for the intercept estimates of Scenario A and B (Fig. S6.8, Fig. S6.9) and we found similar patterns as for the slope estimates (Fig. 8.3, Fig. 8.4).

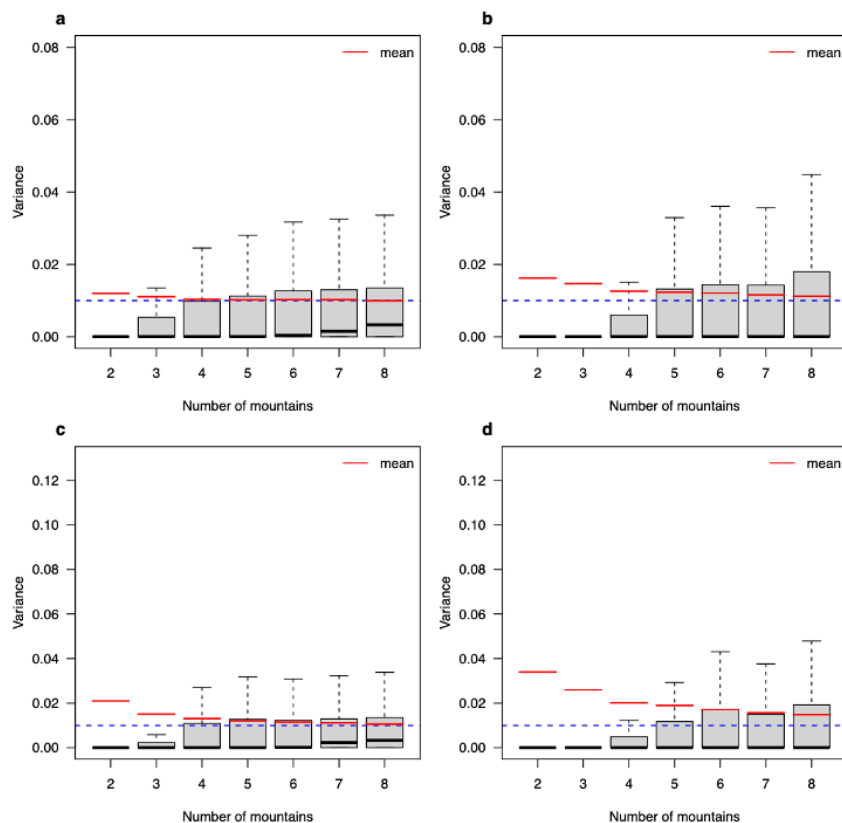
Also, the distribution of the variance estimates of the random effects matched with the results of the LMMs (Fig. 8.5, Fig. S6.10)



**FIGURE S6.8:** Average type I error rates and average power for the intercept in generalized linear fixed and mixed-effect models fitted to simulated data with 2-8 mountains (random intercept for each mountain - Scenario A) and with 200 observations per mountain. For each scenario, 5000 simulations and models were tested. (a, c) show results for mixed-effects models only from datasets in which mixed-effects models converged without presenting singular fit problems and (b, d) results for mixed-effects models for all datasets. Results for fixed-effects (a-d) model are from all datasets. (a-d) the dotted line represents the 5% alpha level.



**FIGURE S6.9:** Average type I error rates and average power for the intercept in generalized linear (mixed-effect) models fitted to simulated data with 2-8 mountains for scenario B (random intercept and random slope for each mountain range). For each scenario, 5,000 simulations and models were tested. (a, b, e, f) show results for simulated data with a variance of 0.01 in the random effects. (c, d, g, h) show results for simulated data with a variance of 0.25 in the random effects. (a, c, e, g) show results for mixed-effects models only from datasets in which mixed-effects models converged without presenting singular fit problems and (b, d, f, h) results for mixed-effects models for all datasets. Results for fixed-effects (a-h) model are from all datasets. In (a-d) the dotted line represents the 5% alpha level.



**FIGURE S6.10:** Variance estimates of random intercepts (a, c) and random slopes (b, d) for generalized linear mixed-effects models in Scenario B, fitted with lme4 using MLE to simulated data with 2-8 mountains. Figures (a) and (b) show the results for all models (singular and non-singular fits) and figures (c) and (d) show the results for only non-singular fits. For each scenario, 5,000 simulations and models were tested. The blue dotted lines represent the true variance used in the simulation (0.01) and the red lines the average variance estimates.

### 3 Calculation of the mean temperature effect in fixed-effect models with interaction

As fixed-effect models with interactions estimate the effect of one level and its contrasts to the other levels, the population effect of temperature itself is not estimated in the R default parametrization. To calculate the population effect and its significance to be able to compare it to mixed-effect models result, we estimate the grand mean and its standard error via bootstrapping. To do so, we sample from the multivariate normal distribution reported from the fitted linear for the interactions. Let  $S$  be a big enough sample from this distribution, then the mean of  $S$  is the grand mean temperature effect, and the standard error of  $S$  is the standard error of the mean. In R language we can calculate p-values using 2000 samples given the effect estimates effects and the covariance matrix of the individual level effects  $V$  from the linear model in the following way:

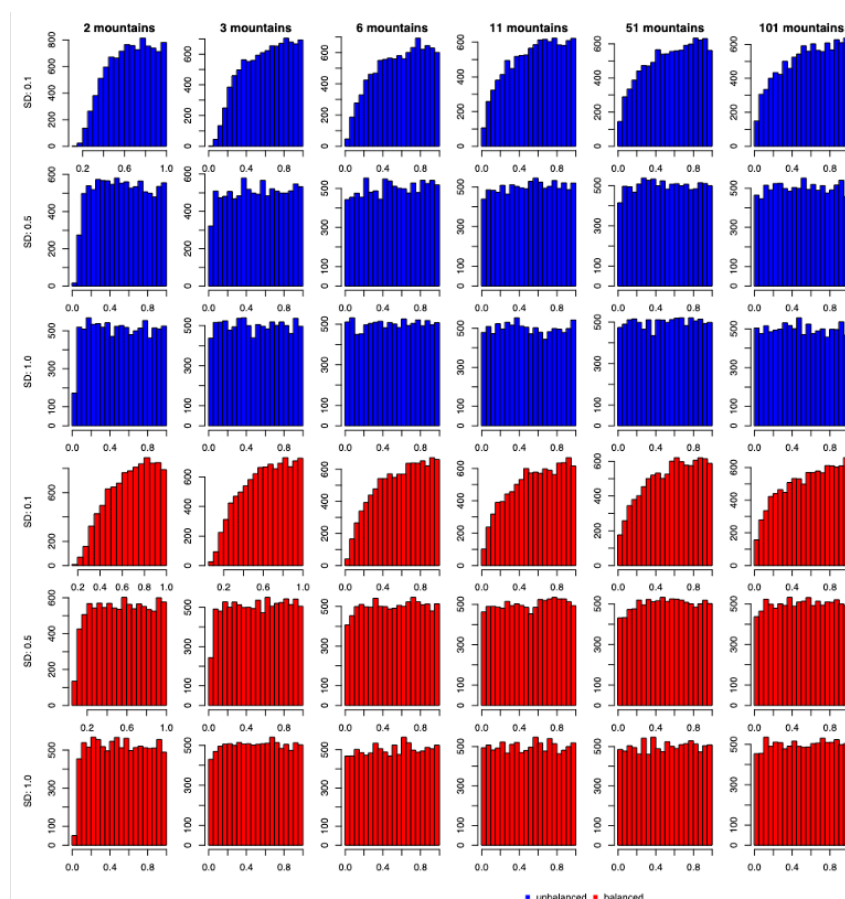
```
S = mvtnorm::rmvnorm(2000, mean = effects, sigma = V)
eff = mean(S)
se = sd(S) / sqrt(mountain - 1)
```

We then used t-values for LMs and z-values for GLMs for a better comparison with the respective packages to calculate p-values. To check whether our calculation of the grand mean behaves approximatively correctly, i.e. with no effect we would expect a uniform distribution of the

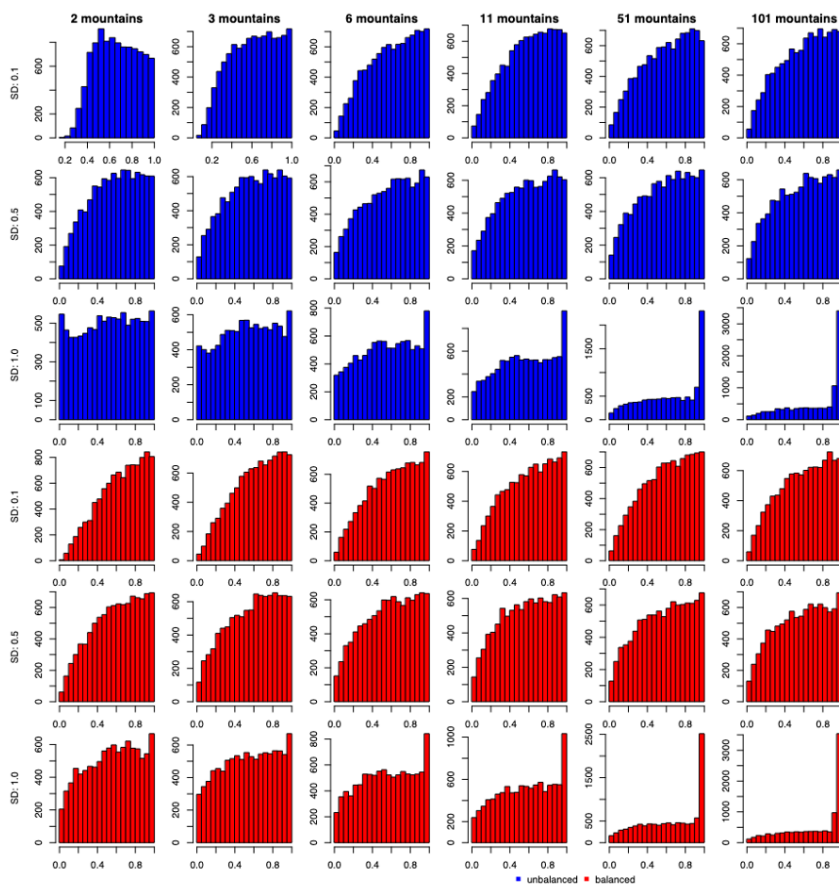
p-values, we simulated scenario B with no grand mean (effect was set to 0), different standard deviations of the varying random intercept and slope (0.1, 0.5, and 1.0), and unbalanced (30 observations for each group but 1000 observations for the last one) and balanced (30 observations for each group) study design. We found that the distribution of p-values is approximately correct for larger standard deviations (which is linked to the number of observations in each group, Fig. S11).

We repeated the simulations for GLMs and we found that the distributions of p-values were on average left-skewed (Fig. S12). The nature of the logit link in the GLM explains the increase in left skewness with higher standard deviations of the random effects (Fig. S12).

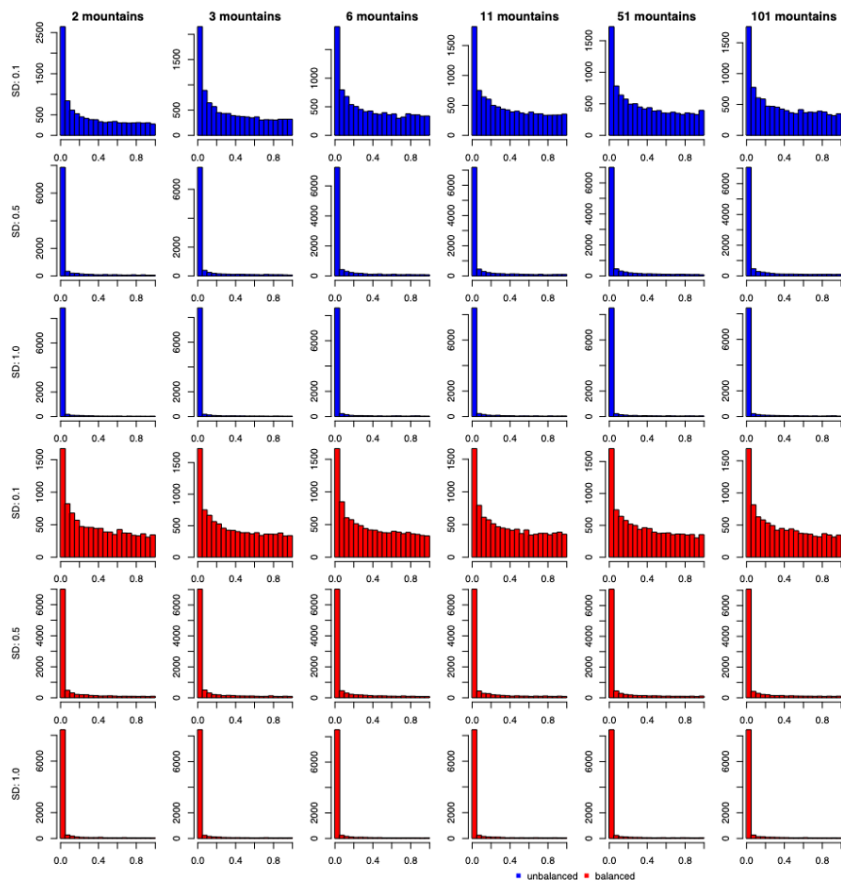
Using zero sum contrasts is another way to calculate the average affects over a grouping variable. We repeated the above-mentioned simulation to compare the zero-sum contrasts method to our bootstrapping approach. However, we found highly right-skewed distribution of p-values for the zero-sum contrasts for both models, the LM and the GLM (Fig. S13, S14). Thus, the zero-sum contrasts method leads to high type I errors and very high power for LMs and GLMs, while our bootstrapping method leads to correct type I errors and power for LMs, but higher than expected type I error rates and less power for the GLMs. Therefore, we think that our bootstrapping approach is preferable over the zero-sum contrast method.



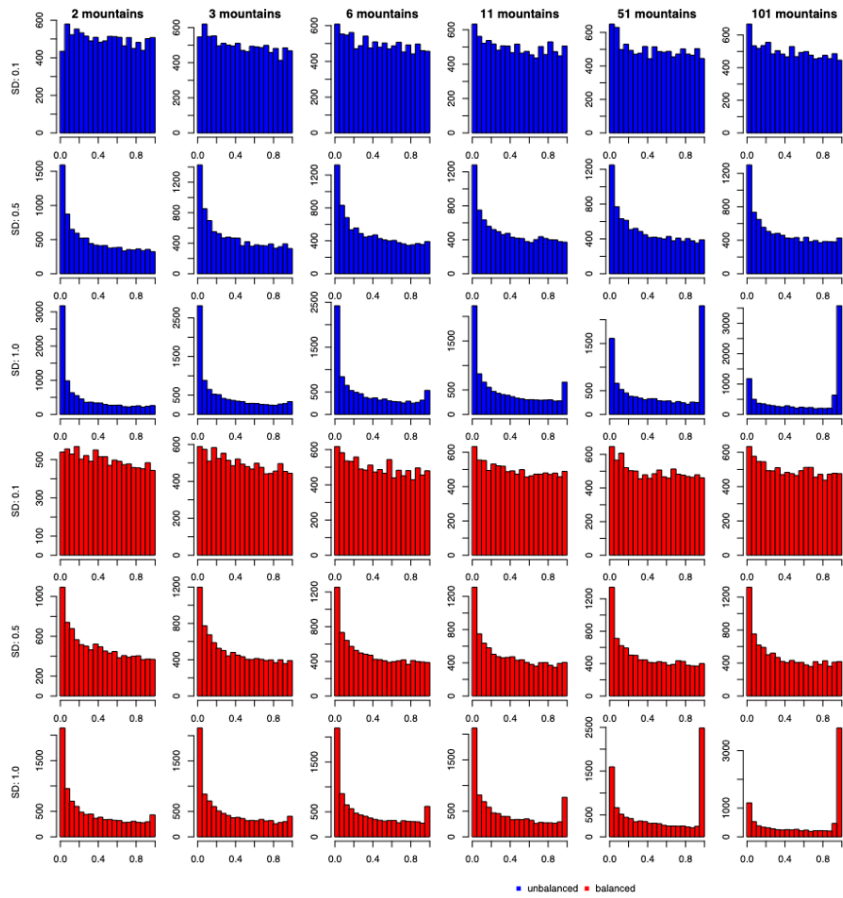
**FIGURE S6.11:** The distribution of p-values (based on t-statistics) for the grand mean calculation of linear models for different simulation scenarios: 2 – 101 mountains, different SDs for the random effects (0.1, 0.5, and 1.0), and for unbalanced (blue) and balanced (red) study design. Each scenario was simulated 10,000 times.



**FIGURE S6.12:** The distribution of p-values (based on z-statistics) for the grand mean calculation of generalized linear models for different simulation scenarios: 2 – 101 mountains, different SDs for the random effects (0.1, 0.5, and 1.0), and for unbalanced (blue) and balanced (red) study design. Each scenario was simulated 10,000 times.



**FIGURE S6.13:** The distribution of p-values (based on t-statistics) for the grand mean calculation in linear models (based on zero sum contrasts) for different simulation scenarios: 2 – 101 mountains, different SDs for the random effects (0.1, 0.5, and 1.0), and for unbalanced (blue) and balanced (red) study design. Each scenario was simulated 10,000 times.

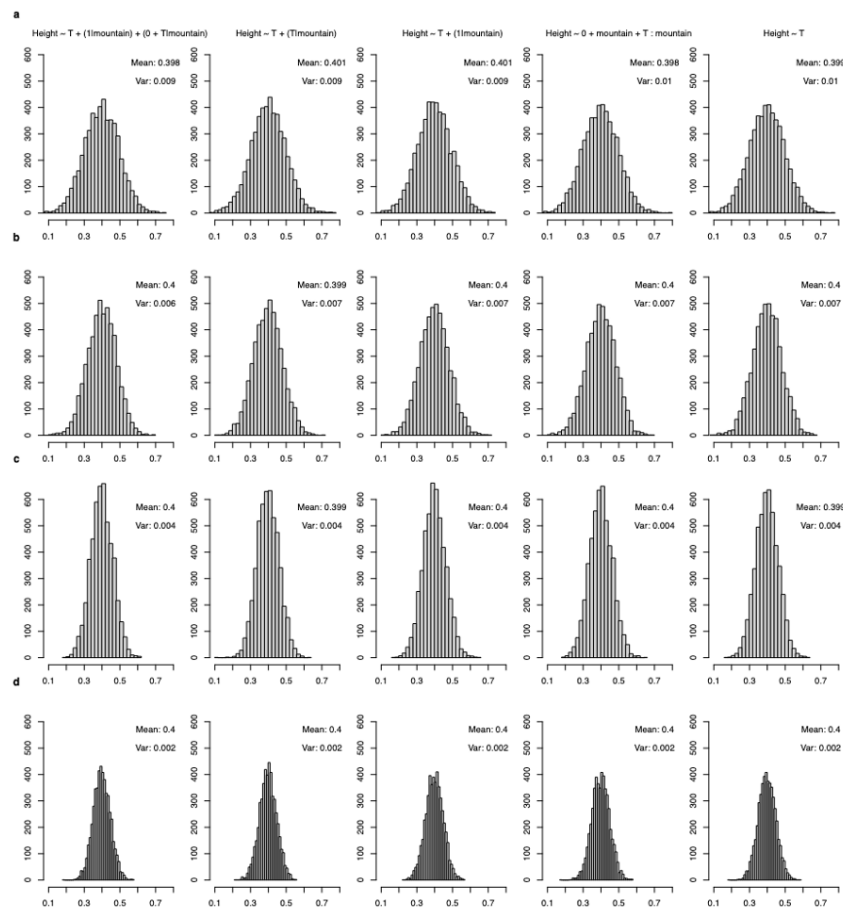


**FIGURE S6.14:** The distribution of p-values (based on t-statistics) for the grand mean calculation in generalized models (based on zero sum contrasts) for different simulation scenarios: 2 – 101 mountains, different SDs for the random effects (0.1, 0.5, and 1.0), and for unbalanced (blue) and balanced (red) study design. Each scenario was simulated 10,000 times.

### Distribution of population-level effect estimates

To check what causes different type I error rates, we plotted the distribution of estimated population mean effects for scenario B for a different number of mountains (2, 3, 5, and 7, corresponding to panels a-d in Fig. S6.15 and S6.16) for the intercept (Fig. S6.15) as well as for the slope (Fig. S6.16). We found no differences for the different models for the estimates of the population effect (Fig. S6.15, S6.16). We found that with increasing number of mountains the variance of the distribution declines.





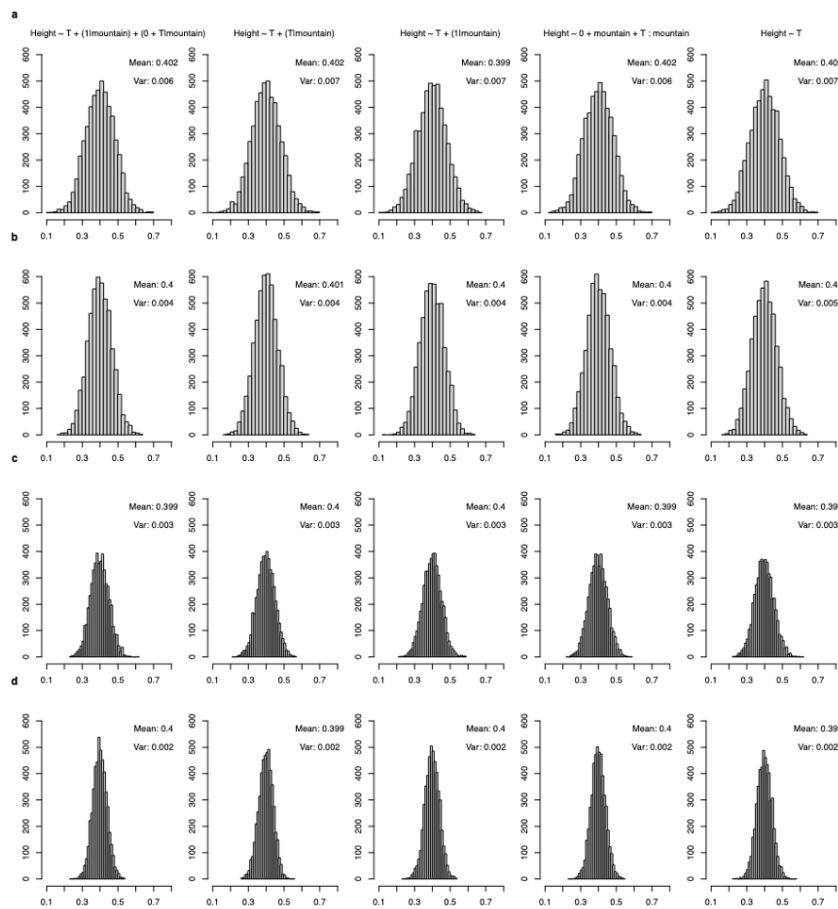
**FIGURE S6.15:** The distribution of the population slope effect (grand mean, ecological effect) of temperature for scenario B. a-d show different number of mountains (2, 3, 5, and 7) and each column corresponds for a different model which we tested for scenario B (see Figure 2 in main text)

#### 4 Mixed-effect model implementations in R, default settings and convergence issues

The most used packages in R to fit mixed-effect models are *lme4* (BATES *et al.*, 2014) and *glmmTMB* (BROOKS, KRISTENSEN, BENTHEM, MAGNUSSON, BERG, NIELSEN, SKAUG, MACHLER, *et al.*, 2017). These packages differ in their optimization routines and the calculation of p-values for linear mixed-effect models (LMMs). While *lme4* uses standard optimizers, *glmmTMB* relies on automatic differentiation implemented in TMB package (KRISTENSEN *et al.*, 2016). Another difference is that *glmmTMB* offers to fit linear and generalized linear models with both the maximum likelihood (MLE) and the restricted maximum likelihood estimation (REML), while *lme4* offers only REML for LMMs but not for GLMMs. By default, *lme4* uses restricted maximum likelihood (REML) for LMMs (*lmer* function) and unrestricted maximum likelihood (MLE) for GLMMs (*glmer* function), while *glmmTMB* uses MLE by default for any kind of data.

#### 5 Results for linear-mixed effect models

We repeated the experiments from the main text with *glmmTMB* instead of *lme4*. Because *lme4* defines a singular fit when the estimated variance of a random effect is smaller than  $10^{-4}$  we decided to use the same threshold for *glmmTMB* (which has no default threshold).

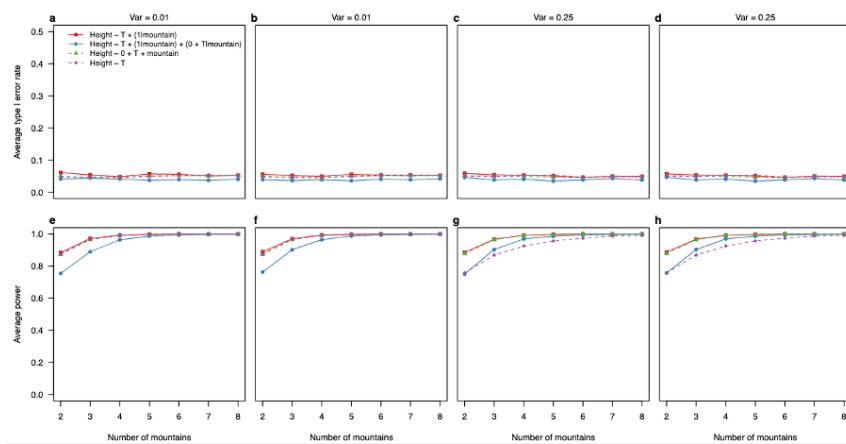


**FIGURE S6.16:** The distribution of the population intercept effect for scenario B. a-d show different number of mountains (2, 3, 5, and 7) and each column corresponds for a different model which we tested for scenario B (see Figure 2 in main text).

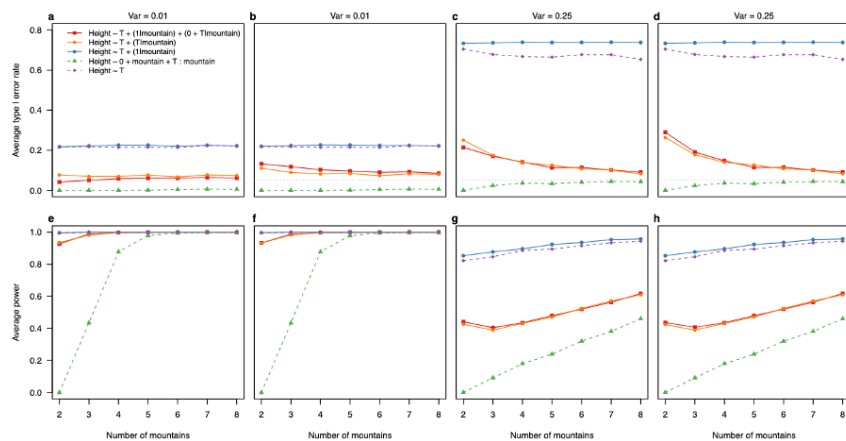
### Results for the slope

For scenario A, the patterns we found for average type I error rate and average statistical power of the population-level effect (effect size of the temperature predictor) were similar to the findings of lme4 (Fig. 8.3, Fig. S6.17). However, the overparametrized model was less affected by the variance of the random effects or if the singular fits were included or not. Also, the average power of the overparametrized model was higher than when using lme4.

For scenario B, again the patterns were similar to lme4 with the exception that for higher variances, the average type I error rates of the mixed-effect models fitted by glmmTMB were higher than the nominal level, regardless of with or without singular fits. Here, the average type I error rates decreased with the number of the number of mountains to the nominal level (Fig. S6.18a-d). In lme4 the average type I error rates were closer to the nominal level (more conservative, Fig. S6.18). The less conservative average type I error rates led also to higher power (Fig. S6.18e-h).



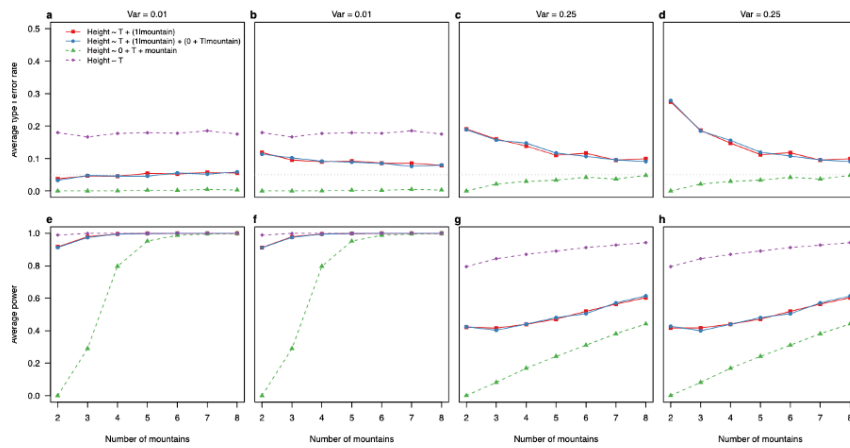
**FIGURE S6.17:** Average type I error rates and average power for linear fixed and mixed-effect models (glmmTMB) fitted to simulated data with 2-8 mountains (random intercept for each mountain - Scenario A) and with 50 observations per mountain. For each scenario, 5000 simulations and models were tested. (a, b, e, f) show results for simulated data with a variance of 0.01 in the random effects. (c, d, g, h) show results for simulated data with a variance of 0.25 in the random effects. (a, c, e, g) show results for mixed-effects models only from datasets in which mixed-effects models converged without presenting singular fit problems and (b, d, f, h) results for mixed-effects models for all datasets. Results for fixed-effects (a-h) model are from all datasets. (a-d) the dotted line represents the 5% alpha level.



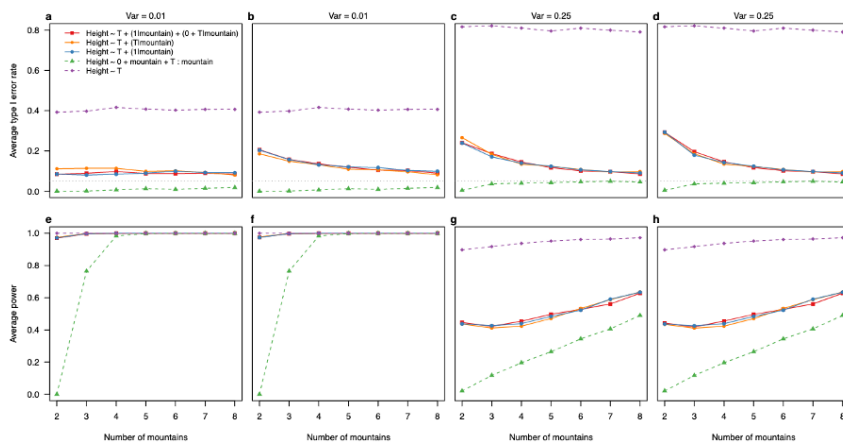
**FIGURE S6.18:** Average type I error rates and average power for linear (mixed-effect) models fitted to simulated data with 2-8 mountains for scenario B (random intercept and random slope for each mountain range) using glmmTMB. For each scenario, 5,000 simulations and models were tested. (a, b, e, f) show results for simulated data with a variance of 0.01 in the random effects. (c, d, g, h) show results for simulated data with a variance of 0.25 in the random effects. (a, c, e, g) show results for mixed-effects models only from datasets in which mixed-effects models converged without presenting singular fit problems and (b, d, f, h) results for mixed-effects models for all datasets. Results for fixed-effects (a-h) model are from all datasets. In (a-d) the dotted line represents the 5% alpha level.

## Results for the intercept

For the intercept, we found a similar pattern as for the slopes: The average type I error rates of the mixed-effect models fitted by glmmTMB were on average higher for low number of levels and stronger affected by larger random effect sizes than when fitted by lme4 (Fig. S6.19, S6.20).



**FIGURE S6.19:** Average type I error rates and average power for the intercept in linear fixed and mixed-effect models (glmmTMB) fitted to simulated data with 2-8 mountains (random intercept for each mountain - Scenario A) and with 50 observations per mountain. For each scenario, 5000 simulations and models were tested. (a, b, e, f) show results for simulated data with a variance of 0.01 in the random effects. (c, d, g, h) show results for simulated data with a variance of 0.25 in the random effects. (a, c, e, g) show results for mixed-effects models only from datasets in which mixed-effects models converged without presenting singular fit problems and (b, d, f, h) results for mixed-effects models for all datasets. Results for fixed-effects (a-h) model are from all datasets. (a-d) the dotted line represents the 5% alpha level.



**FIGURE S6.20:** Average type I error rates and average power for the intercept in linear mixed-effect models (glmmTMB) fitted to simulated data with 2-8 mountains for scenario B (random intercept and random slope for each mountain range). For each scenario, 5,000 simulations and models were tested. (a, b, e, f) show results for simulated data with a variance of 0.01 in the random effects. (c, d, g, h) show results for simulated data with a variance of 0.25 in the random effects. (a, c, e, g) show results for mixed-effects models only from datasets in which mixed-effects models converged without presenting singular fit problems and (b, d, f, h) results for mixed-effects models for all datasets. Results for fixed-effects (a-h) model are from all datasets. In (a-d) the dotted line represents the 5% alpha level.

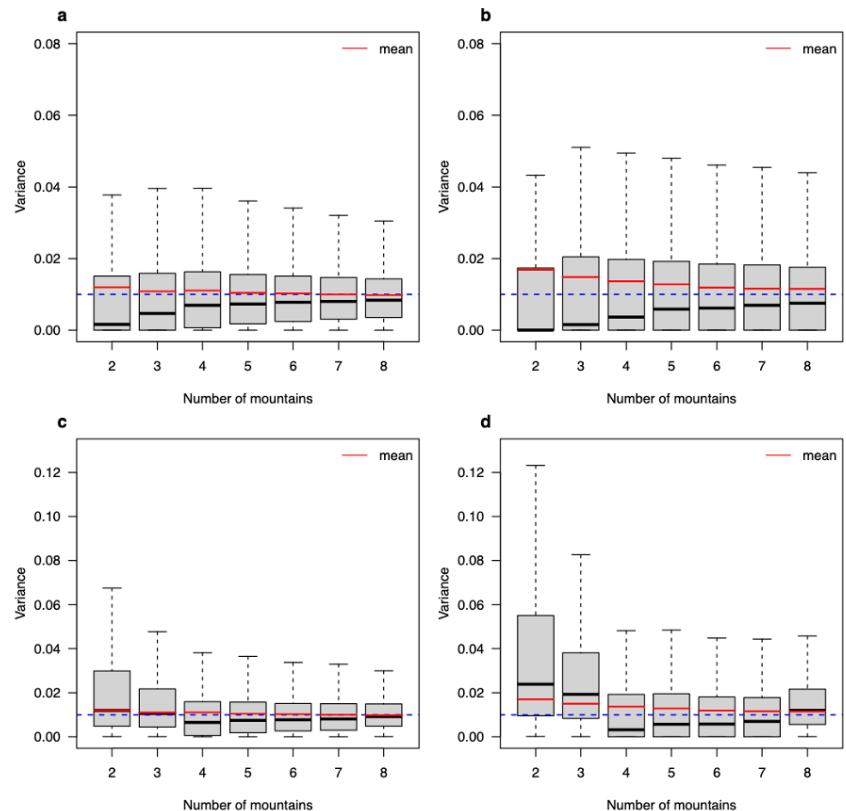
### Variance estimates and singular fits using glmmTMB

We found that singular fit occurred more often in mixed-effect models when using MLE compared to REML (Table S6.2). Also, the rate of singular fits decreased with increasing number of mountains (Table S6.2)). When using a threshold of  $10^{-4}$  (the same as for lme4) to detect a singular fit, the

rate of singular fits to non-singular fits was the same as for lme4.

For non-singular fits, we found that the average variance estimate in the mixed-effect models were closed to the true value of the data generating process than in lme4 (Fig. 8.5), Fig. S6.21). Estimates for balanced and unbalanced data do not differ within REML and MLE (Fig. S6.23, S6.24)

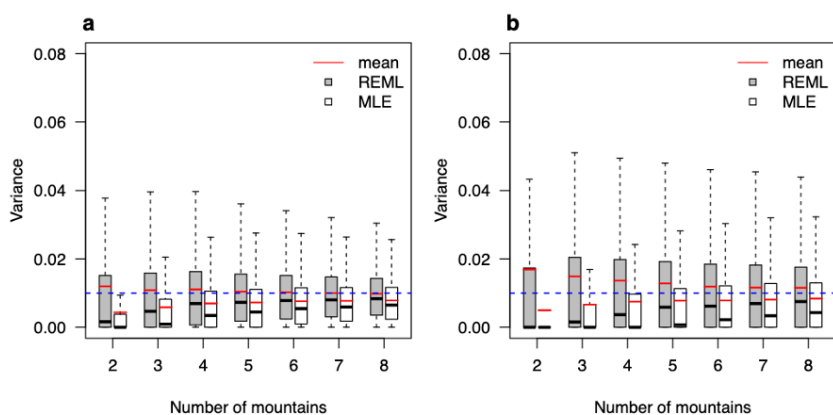
Additional to the rate of singular fits a direct comparison of the variance estimates using REML and MLE is necessary to compare their performance. Using MLE for linear mixed-effect models led to stronger towards zero biased estimates compared to using REML (Fig. S6.22).

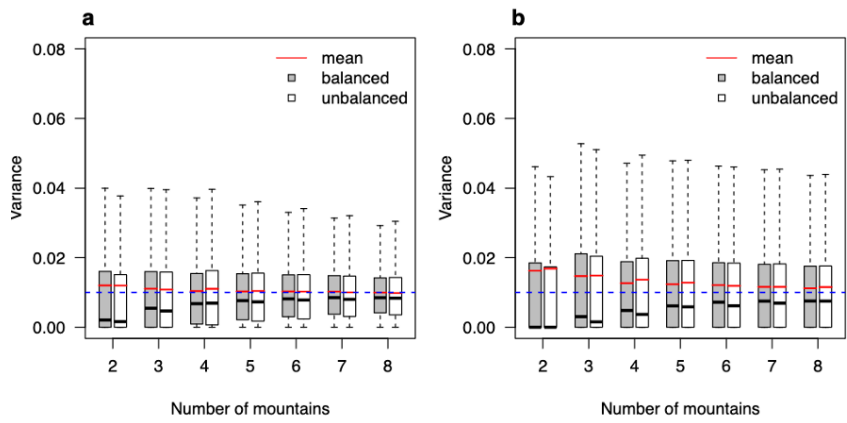


**FIGURE S6.21:** Variance estimates of random intercepts (a, c) and random slopes (b, d) for linear mixed-effects models (LMM, Table 1. Eq. M10) in Scenario B, fitted with glmmTMB using REML to simulated data with 2-8 mountains. Figures (a) and (b) show the results for all models (singular and non-singular fits) and figures (c) and (d) show the results for only non-singular fits. For each scenario, 5,000 simulations and models were tested. The blue dotted lines represent the true variance used in the simulation (0.01) and the red lines the average variance estimates.

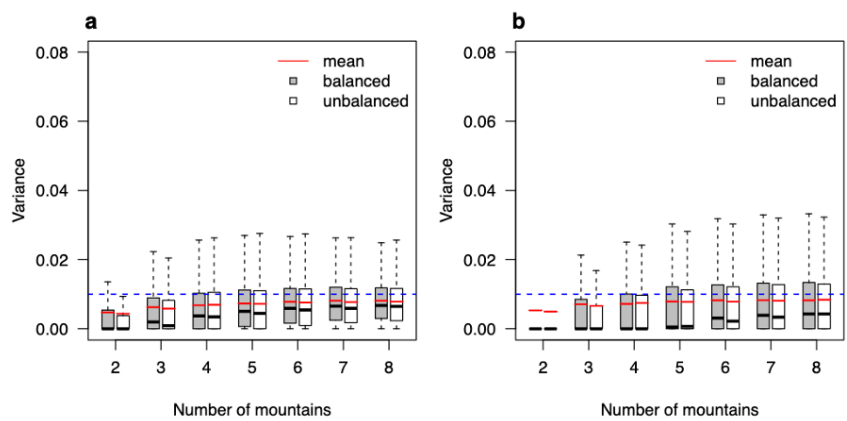
**TABLE S6.2:** Proportion of models ran in glmmTMB that presented singular fit convergence problem when using maximum likelihood (MLE) and restricted maximum likelihood (REML) fitting algorithms.

Number of groups	LMM		GLMM	
	REML	MLE	MLE	REML
2	77%	92%	87%	97%
3	65%	81%	80%	92%
4	55%	71%	76%	88%
5	46%	62%	72%	84%
6	41%	54%	69%	81%
7	37%	48%	68%	80%
8	33%	43%	64%	76%

**FIGURE S6.22:** Variance estimates of the random intercepts (a) and random slopes (b) for linear mixed-effect models (LMM) fitted to simulated data with 2-8 numbers of artificial mountain ranges. For each scenario, 5,000 simulations and models were tested. The blue line represents the true variance used in the simulation (0.01). The grey boxes show results for the models fitted by restricted maximum likelihood estimation (REML) and the white boxes shows results for the models fitted by maximum likelihood estimation (MLE).



**FIGURE S6.23:** Variance estimates of the random intercepts (a) and random slopes (b) for linear mixed-effect models (LMM) fitted to simulated data with 2-8 numbers of artificial mountain ranges using REML. For each scenario, 5,000 simulations and models were tested. The blue line represents the true variance used in the simulation (0.01). The grey boxes show the results for the models with unbalanced data (number of observation) among mountains and the white boxes shows results for the models fitted with balanced data among mountains.



**FIGURE S6.24:** Variance estimates of the random intercepts (a) and random slopes (b) for linear mixed-effect models (LMM) fitted to simulated data with 2-8 numbers of artificial mountain ranges using MLE. For each scenario, 5,000 simulations and models were tested. The blue line represents the true variance used in the simulation (0.01). The grey boxes show the results for the models with unbalanced data (number of observation) among mountains and the white boxes shows results for the models fitted with balanced data among mountains.

## 6 Discussion

At least to some extent, the differences between `glmmTMB` and `lme4` might be explained by the arbitrary chosen threshold to detect singular fits. It is doubtful if the singular fits rate of the two different packages are truly comparable when they are classified with the same threshold but rely on different implementations and optimization routines.

However, even if we include the singular fits in the results of the mixed-effect models, we found that `glmmTMB` showed on average a higher type I error rate for larger variances in the random effect than `lme4` (Fig. 8.3, 8.4, Fig. S6.17, S6.18, S6.19, S6.20) indicating that `lme4` can handle in general singular fits better than `glmmTMB` because the average type I error rate of `lme4` was here

closer to the nominal level (Fig. 8.3, 8.4, Fig. S6.17, S6.18, S6.19, S6.20).

Future work should focus on exploring and understanding the cause of this difference between the two mixed-effect model implementations.



# References

- AARTS, Emmeke *et al.* (Apr. 2014). “A solution to dependency: using multilevel analysis to accommodate nested data”. en. In: *Nature Neuroscience* 17.4. Number: 4 Publisher: Nature Publishing Group, pp. 491–496. ISSN: 1546-1726. DOI: 10.1038/nn.3648.
- ABADI, Mart?in *et al.* (2015). “TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems”. In: Software available from tensorflow.org.
- ABEDI, M, M BARTELHEIMER, and P POSCHLOD (2014). “Effects of substrate type, moisture and its interactions on soil seed survival of three Rumex species”. In: *Plant and soil* 374.1, pp. 485–495.
- ADERHOLD, Andrej *et al.* (Sept. 2012). “Hierarchical Bayesian models in ecology: Reconstructing species interaction networks from non-homogeneous species abundance data”. In: *Ecological Informatics* 11, pp. 55–64. ISSN: 1574-9541.
- ALBERY, Gregory F. *et al.* (Dec. 2021). “The science of the host–virus network”. en. In: *Nature Microbiology* 6.12. Number: 12 Publisher: Nature Publishing Group, pp. 1483–1492. ISSN: 2058-5276. DOI: 10.1038/s41564-021-00999-5.
- ALLOUCHE, Omri, Asaf TSOAR, and Ronen KADMON (2006). “Assessing the accuracy of species distribution models: prevalence, kappa and the true skill statistic (TSS)”. en. In: *Journal of Applied Ecology* 43.6, pp. 1223–1232. ISSN: 1365-2664. DOI: 10.1111/j.1365-2664.2006.01214.x.
- ALVES DE OLIVEIRA, Vinicius *et al.* (2021). “Reduced-complexity end-to-end variational autoencoder for on board satellite image compression”. In: *Remote Sensing* 13.3, p. 447.
- ALVI, Mohsan, Andrew ZISSERMAN, and Christoffer NELLAAKER (Sept. 2018). “Turning a Blind Eye: Explicit Removal of Biases and Variation from Deep Neural Network Embeddings”. In: *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*.
- AMESÖDER, Christian and Maximilian PICHLER (2023). *cito: Building and Training Neural Networks*. R package version 1.0.1.
- ANDERSON, T Michael, Martin SCHÜTZ, and Anita C RISCH (2012). “Seed germination cues and the importance of the soil seed bank across an environmental gradient in the Serengeti”. In: *Oikos* 121.2, pp. 306–312.
- ANGELOV, Boyan (Sept. 2020). *boyanangelov/interpretable\_sdm: Reproducibility fix*. Version v1.3. DOI: 10.5281/zenodo.4048271.
- AODHA, Oisin Mac *et al.* (Aug. 2018). “Bat detective—Deep learning tools for bat acoustic signal detection”. en. In: *PLOS Computational Biology* 14.3, e1005995. ISSN: 1553-7358. DOI: 10.1371/journal.pcbi.1005995.
- APLEY, Daniel W (2016). “Visualizing the Effects of Predictor Variables in Black Box Supervised Learning Models”. In: *arXiv preprint arXiv:1612.08468*.
- APLEY, Daniel W and Jingyu ZHU (2020). “Visualizing the effects of predictor variables in black box supervised learning models”. In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 82.4, pp. 1059–1086.
- ARIF, Suchinta and Aaron MACNEIL (2022a). “Predictive models aren’t for causal inference”. en. In: *Ecology Letters* n/a.n/a. ISSN: 1461-0248. DOI: 10.1111/ele.14033.
- ARIF, Suchinta and M Aaron MACNEIL (2022b). “Predictive models aren’t for causal inference”. In: *Ecology Letters* 25.8, pp. 1741–1745.
- ARIK, Sercan O. and Tomas PFISTER (Dec. 2020). “TabNet: Attentive Interpretable Tabular Learning”. In: *arXiv:1908.07442 [cs, stat]*. arXiv: 1908.07442.
- ARNQVIST, Göran (Apr. 2020a). “Mixed Models Offer No Freedom from Degrees of Freedom”. en. In: *Trends in Ecology & Evolution* 35.4, pp. 329–335. ISSN: 01695347. DOI: 10.1016/j.tree.2019.12.004.

- ARNQVIST, Göran (Apr. 2020b). “Mixed Models Offer No Freedom from Degrees of Freedom”. en. In: *Trends in Ecology & Evolution* 35.4, pp. 329–335. ISSN: 01695347. DOI: 10.1016/j.tree.2019.12.004.
- ARORA, Sanjeev *et al.* (Nov. 2019). “On Exact Computation with an Infinitely Wide Neural Net”. In: *arXiv:1904.11955 [cs, stat]*. arXiv: 1904.11955.
- ASHUKHA, Arsenii *et al.* (July 2021). “Pitfalls of In-Domain Uncertainty Estimation and Ensembling in Deep Learning”. In: *arXiv:2002.06470 [cs, stat]*. arXiv: 2002.06470.
- BAAYEN, R. H., D. J. DAVIDSON, and D. M. BATES (Nov. 2008). “Mixed-effects modeling with crossed random effects for subjects and items”. en. In: *Journal of Memory and Language*. Special Issue: Emerging Data Analysis 59.4, pp. 390–412. ISSN: 0749-596X. DOI: 10.1016/j.jml.2007.12.005.
- BAILES, Emily J. *et al.* (Aug. 2015). “How can an understanding of plant-pollinator interactions contribute to global food security?” In: *Current Opinion in Plant Biology* 26, pp. 72–79. ISSN: 1369-5266.
- BAKER, Frank B and Seock-Ho KIM (2004). *Item response theory: Parameter estimation techniques*. CRC Press.
- BAKKER, JP *et al.* (1996). “Seed banks and seed dispersal: important topics in restoration ecology”. In: *Acta botanica neerlandica* 45.4, pp. 461–490.
- BÁLINT, Miklós *et al.* (Dec. 2018). “Environmental DNA Time Series in Ecology”. en. In: *Trends in Ecology & Evolution* 33.12, pp. 945–957. ISSN: 0169-5347. DOI: 10.1016/j.tree.2018.09.003.
- BARBEDO, Jayme G.A. (Aug. 2018). “Factors influencing the use of deep learning for plant disease recognition”. en. In: *Biosystems Engineering* 172, pp. 84–91. ISSN: 15375110. DOI: 10.1016/j.biosystemseng.2018.05.013.
- BARBET-MASSIN, Morgane *et al.* (Mar. 2018). “Can species distribution models really predict the expansion of invasive species?” en. In: *PLOS ONE* 13.3. Publisher: Public Library of Science, e0193085. ISSN: 1932-6203. DOI: 10.1371/journal.pone.0193085.
- BARR, Dale J. *et al.* (Apr. 2013). “Random effects structure for confirmatory hypothesis testing: Keep it maximal”. en. In: *Journal of Memory and Language* 68.3, pp. 255–278. ISSN: 0749-596X. DOI: 10.1016/j.jml.2012.11.001.
- BARRAQUAND, Frédéric *et al.* (Mar. 2021). “Inferring species interactions using Granger causality and convergent cross mapping”. en. In: *Theoretical Ecology* 14.1, pp. 87–105. ISSN: 1874-1746. DOI: 10.1007/s12080-020-00482-7.
- BARREDO ARRIETA, Alejandro *et al.* (June 2020). “Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI”. en. In: *Information Fusion* 58, pp. 82–115. ISSN: 1566-2535. DOI: 10.1016/j.inffus.2019.12.012.
- BARSOUM, N. *et al.* (June 2019). “The devil is in the detail: Metabarcoding of arthropods provides a sensitive measure of biodiversity response to forest stand composition compared with surrogate measures of biodiversity”. en. In: *Ecological Indicators* 101, pp. 313–323. ISSN: 1470-160X. DOI: 10.1016/j.ecolind.2019.01.023.
- BARTHOLOMEW, David J, Martin KNOTT, and Irimi MOUSTAKI (2011). *Latent variable models and factor analysis: A unified approach*. Vol. 904. John Wiley & Sons.
- BARTOLDSON, Brian R. *et al.* (Oct. 2020). “The Generalization-Stability Tradeoff In Neural Network Pruning”. In: *arXiv:1906.03728 [cs, stat]*. arXiv: 1906.03728.
- BARTOMEUS, Ignasi *et al.* (2016). “A common framework for identifying linkage rules across different types of interactions”. en. In: *Functional Ecology* 30.12, pp. 1894–1903. ISSN: 1365-2435. DOI: 10.1111/1365-2435.12666.
- BATES, Douglas *et al.* (2014). “Fitting linear mixed-effects models using lme4”. In: *arXiv preprint arXiv:1406.5823*.
- BECKER, Daniel J *et al.* (Jan. 2022). “Optimising predictive models to prioritise viral discovery in zoonotic reservoirs”. en. In: *The Lancet Microbe*. ISSN: 2666-5247. DOI: 10.1016/S2666-5247(21)00245-7.

- BEERY, Sara, Elijah COLE, *et al.* (2021). “Species distribution modeling for machine learning practitioners: A review”. In: *ACM SIGCAS conference on computing and sustainable societies*, pp. 329–348.
- BEERY, Sara, Grant VAN HORN, and Pietro PERONA (2018). “Recognition in terra incognita”. In: *Proceedings of the European conference on computer vision (ECCV)*, pp. 456–473.
- BEERY, Sara, Guanhang WU, *et al.* (2020). “Context r-cnn: Long term temporal context for per-camera object detection”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13075–13085.
- BEHR, Merle *et al.* (2022). “Provable Boolean interaction recovery from tree ensemble obtained via random forests”. In: *Proceedings of the National Academy of Sciences* 119.22, e2118636119.
- BEKKER, RM *et al.* (1998). “Seed size, shape and vertical distribution in the soil: indicators of seed longevity”. In: *Functional Ecology* 12.5, pp. 834–842.
- BELGIU, Mariana and Lucian DRĂGUȚ (2016). “Random forest in remote sensing: A review of applications and future directions”. In: *ISPRS journal of photogrammetry and remote sensing* 114, pp. 24–31.
- BELKIN, Mikhail *et al.* (Sept. 2019). “Reconciling modern machine learning practice and the bias-variance trade-off”. In: *arXiv:1812.11118 [cs, stat]*. arXiv: 1812.11118.
- BELL, Andrew, Malcolm FAIRBROTHER, and Kelvyn JONES (Mar. 2019). “Fixed and random effects models: making an informed choice”. en. In: *Quality & Quantity* 53.2, pp. 1051–1074. ISSN: 1573-7845. DOI: 10.1007/s11135-018-0802-x.
- BENDER, Irene M. A. *et al.* (2017). “Functionally specialised birds respond flexibly to seasonal changes in fruit availability”. en. In: *Journal of Animal Ecology* 86.4, pp. 800–811. ISSN: 1365-2656. DOI: 10.1111/1365-2656.12683.
- BENGIO, Yoshua *et al.* (2019). “A meta-transfer objective for learning to disentangle causal mechanisms”. In: *arXiv preprint arXiv:1901.10912*.
- BERGLER, Christian *et al.* (July 2019). “ORCA-SPOT: An Automatic Killer Whale Sound Detection Toolkit Using Deep Learning”. En. In: *Scientific Reports* 9.1, p. 10997. ISSN: 2045-2322. DOI: 10.1038/s41598-019-47335-w.
- BERGSTRA, James and Yoshua BENGIO (2012). “Random search for hyper-parameter optimization”. In: *Journal of Machine Learning Research* 13.Feb, pp. 281–305.
- BERLOW, Eric L. *et al.* (Jan. 2009). “Simple prediction of interaction strengths in complex food webs”. In: *Proc Natl Acad Sci USA* 106.1, p. 187.
- BERNARDO, José M and Adrian FM SMITH (2009). *Bayesian theory*. Vol. 405. John Wiley & Sons.
- BERNER, Christopher *et al.* (2019). “Dota 2 with large scale deep reinforcement learning”. In: *arXiv preprint arXiv:1912.06680*.
- BHATTACHARYA, Anirban and David B DUNSON (n.d.). “Sparse Bayesian infinite factor models”. In: *Biometrika* (), pp. 291–306.
- BIRDAL, Tolga *et al.* (2021). “Intrinsic dimension, persistent homology and generalization in neural networks”. In: *Advances in Neural Information Processing Systems* 34.
- BISCHL, Bernd *et al.* (2016). “mlr: Machine Learning in R”. In: *Journal of Machine Learning Research* 17.170, pp. 1–5.
- BLANCHET, F. Guillaume, Kevin CAZELLES, and Dominique GRAVEL (2020). “Co-occurrence is not evidence of ecological interactions”. en. In: *Ecology Letters* 23.7, pp. 1050–1063. ISSN: 1461-0248. DOI: 10.1111/ele.13525.
- BLÜTHGEN, Nico *et al.* (Feb. 2007). “Specialization, Constraints, and Conflicting Interests in Mutualistic Networks”. In: *Current Biology* 17.4, pp. 341–346. ISSN: 0960-9822. DOI: 10.1016/j.cub.2006.12.039.
- BOET, Olga, Xavier ARNAN, and Javier RETANA (Feb. 2020). “The role of environmental vs. biotic filtering in the structure of European ant communities: A matter of trait type and spatial scale”.

- en. In: *PLOS ONE* 15.2. Publisher: Public Library of Science, e0228625. ISSN: 1932-6203. DOI: 10.1371/journal.pone.0228625.
- BOHMANN, Kristine *et al.* (June 2014). “Environmental DNA for wildlife biology and biodiversity monitoring”. en. In: *Trends in Ecology & Evolution* 29.6, pp. 358–367. ISSN: 0169-5347. DOI: 10.1016/j.tree.2014.04.003.
- BOISGONTIER, Matthieu P. and Boris CHEVAL (Sept. 2016). “The anova to mixed model transition”. en. In: *Neuroscience & Biobehavioral Reviews* 68, pp. 1004–1005. ISSN: 0149-7634. DOI: 10.1016/j.neubiorev.2016.05.034.
- BOLKER, Benjamin M (2015). “Linear and generalized linear mixed models”. In: *Ecological Statistics: Contemporary theory and application*. Publisher: Oxford University Press Oxford, New York, pp. 309–333.
- BOLKER, Benjamin M. *et al.* (Mar. 2009). “Generalized linear mixed models: a practical guide for ecology and evolution”. en. In: *Trends in Ecology & Evolution* 24.3, pp. 127–135. ISSN: 01695347. DOI: 10.1016/j.tree.2008.10.008.
- BOLLEN, Kenneth A and Mark D NOBLE (2011). “Structural equation models and the quantification of behavior”. In: *Proceedings of the National Academy of Sciences* 108.supplement\_3, pp. 15639–15646.
- BOOTH, James G and James P HOBERT (1999). “Maximizing generalized linear mixed model likelihoods with an automated Monte Carlo EM algorithm”. In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 61.1, pp. 265–285.
- BORDT, Sebastian *et al.* (2022). “Post-hoc explanations fail to achieve their purpose in adversarial contexts”. In: *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*, pp. 891–905.
- BOROWIEC, Marek L., Rebecca B. DIKOW, *et al.* (2022). “Deep learning as a tool for ecology and evolution”. en. In: *Methods in Ecology and Evolution* n/a.n/a. ISSN: 2041-210X. DOI: 10.1111/2041-210X.13901.
- BOROWIEC, Marek L., Paul FRANSEN, *et al.* (June 2021). *Deep learning as a tool for ecology and evolution*. en-us. Tech. rep. type: article. EcoEvoRxiv. DOI: 10.32942/osf.io/nt3as.
- BOSER, Bernhard E., Isabelle M. GUYON, and Vladimir N. VAPNIK (July 1992). “A training algorithm for optimal margin classifiers”. In: *Proceedings of the fifth annual workshop on Computational learning theory*. COLT '92. New York, NY, USA: Association for Computing Machinery, pp. 144–152. ISBN: 978-0-89791-497-0. DOI: 10.1145/130385.130401.
- BOTTOU, Léon (2010). “Large-Scale Machine Learning with Stochastic Gradient Descent”. en. In: *Proceedings of COMPSTAT'2010*. Ed. by Yves LECHEVALIER and Gilbert SAPORTA. Heidelberg: Physica-Verlag HD, pp. 177–186. ISBN: 978-3-7908-2604-3. DOI: 10.1007/978-3-7908-2604-3\_16.
- BRAMBOR, Thomas, William Roberts CLARK, and Matt GOLDR (2006). “Understanding interaction models: Improving empirical analyses”. In: *Political analysis* 14.1, pp. 63–82.
- BREIMAN, Leo (Aug. 1996). “Bagging predictors”. en. In: *Machine Learning* 24.2, pp. 123–140. ISSN: 1573-0565. DOI: 10.1007/BF00058655.
- (2001a). “Random Forests”. In: *Machine Learning* 45.1, pp. 5–32. ISSN: 1573-0565.
- (Aug. 2001b). “Statistical Modeling: The Two Cultures (with comments and a rejoinder by the author)”. en. In: *Statistical Science* 16.3, pp. 199–231. ISSN: 0883-4237, 2168-8745. DOI: 10.1214/ss/1009213726.
- BRIEUC, Marine S. O. *et al.* (July 2018). “A practical introduction to Random Forest for genetic association studies in ecology and evolution”. In: *Mol Ecol Resour* 18.4, pp. 755–766. ISSN: 1755-098X. DOI: 10.1111/1755-0998.12773.
- BROOKS E., Mollie, Kasper KRISTENSEN, van Koen BENTHEM J., Arni MAGNUSSON, Casper BERG W., Anders NIELSEN, Hans SKAUG J., Martin MÄCHLER, *et al.* (2017). “glmmTMB Balances Speed and

- Flexibility Among Packages for Zero-inflated Generalized Linear Mixed Modeling”. en. In: *The R Journal* 9.2, p. 378. ISSN: 2073-4859. DOI: 10.32614/RJ-2017-066.
- BROOKS, Mollie E, Kasper KRISTENSEN, Koen J van BENTHEM, Arni MAGNUSSON, Casper W BERG, Anders NIELSEN, Hans J SKAUG, Martin MACHLER, *et al.* (2017). “glmmTMB balances speed and flexibility among packages for zero-inflated generalized linear mixed modeling”. In: *The R Journal* 9.2. Publisher: Technische Universitaet Wien, pp. 378–400.
- BROUSSEAU, Pierre-Marc, Dominique GRAVEL, and I. Tanya HANDA (Jan. 2018a). “Trait matching and phylogeny as predictors of predator-prey interactions involving ground beetles”. In: *Funct Ecol* 32.1, pp. 192–202. ISSN: 0269-8463. DOI: 10.1111/1365-2435.12943.
- (2018b). “Trait matching and phylogeny as predictors of predator–prey interactions involving ground beetles”. en. In: *Functional Ecology* 32.1, pp. 192–202. ISSN: 1365-2435. DOI: 10.1111/1365-2435.12943.
- BROWN, Alexandra M, David I WARTON, *et al.* (2014). “The fourth-corner solution—using predictive models to understand how species traits interact with the environment”. In: *Methods in Ecology and Evolution* 5.4. Publisher: Wiley Online Library, pp. 344–352.
- BROWN, Tom, Benjamin MANN, *et al.* (2020). “Language models are few-shot learners”. In: *Advances in neural information processing systems* 33, pp. 1877–1901.
- BRUNBJERG, Ane Kirstine *et al.* (Feb. 2017). “Ecospace: A unified framework for understanding variation in terrestrial biodiversity”. en. In: *Basic and Applied Ecology* 18, pp. 86–94. ISSN: 1439-1791. DOI: 10.1016/j.baae.2016.09.002.
- BRYSSBAERT, Marc and Michaël STEVENS (2018). “Power Analysis and Effect Size in Mixed Effects Models: A Tutorial”. In: *Journal of Cognition* 1.1. ISSN: 2514-4820. DOI: 10.5334/joc.10.
- BURNHAM, Kenneth P. and David R. ANDERSON (Nov. 2004). “Multimodel Inference: Understanding AIC and BIC in Model Selection”. en. In: *Sociological Methods & Research* 33.2. Publisher: SAGE Publications Inc, pp. 261–304. ISSN: 0049-1241. DOI: 10.1177/0049124104268644.
- BURY, Thomas M *et al.* (2021). “Deep learning for early warning signals of tipping points”. In: *Proceedings of the National Academy of Sciences* 118.39, e2106140118.
- BYSTROVA, Daria *et al.* (Mar. 2021). “Clustering Species With Residual Covariance Matrix in Joint Species Distribution Models”. en. In: *Frontiers in Ecology and Evolution* 9, p. 601384. ISSN: 2296-701X. DOI: 10.3389/fevo.2021.601384.
- CAI, Wang *et al.* (2023). “Environmental DNA captures the internal structure of a pond metacom- munity”. In: *bioRxiv*, pp. 2023–12.
- CALATAYUD, Joaquín *et al.* (Dec. 2019). “Positive associations among rare species and their persis- tence in ecological assemblages”. en. In: *Nature Ecology & Evolution*, pp. 1–6. ISSN: 2397-334X. DOI: 10.1038/s41559-019-1053-5.
- CALISKAN, Aylin, Joanna J. BRYSON, and Arvind NARAYANAN (Apr. 2017). “Semantics derived automatically from language corpora contain human-like biases”. In: *Science* 356.6334. Publisher: American Association for the Advancement of Science, pp. 183–186. DOI: 10.1126/science. aal4230.
- CANARD, E. F. *et al.* (Apr. 2014). “Empirical Evaluation of Neutral Interactions in Host-Parasite Networks.” In: *The American Naturalist* 183.4, pp. 468–479. ISSN: 0003-0147. DOI: 10.1086/675363.
- CARDOSO, Pedro *et al.* (2020). “Automated Discovery of Relationships, Models, and Principles in Ecology”. In: *Frontiers in Ecology and Evolution* 8. ISSN: 2296-701X.
- CHEN, Di, Yexiang XUE, and Carla P. GOMES (Mar. 2018). “End-to-End Learning for the Deep Multivariate Probit Model”. In: *arXiv:1803.08591 [cs, stat]*. arXiv: 1803.08591.
- CHEN, Ricky T. Q., Yulia RUBANOVA, *et al.* (Dec. 2019). “Neural Ordinary Differential Equations”. In: *arXiv:1806.07366 [cs, stat]*. arXiv: 1806.07366.
- CHEN, Tianqi and Carlos GUESTRIN (2016). “XGBoost: A Scalable Tree Boosting System”. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and*

- Data Mining*. KDD '16. event-place: San Francisco, California, USA. New York, NY, USA: ACM, pp. 785–794. ISBN: 978-1-4503-4232-2. DOI: 10.1145/2939672.2939785.
- CHEN, Zhen and David B. DUNSON (2003). “Random Effects Selection in Linear Mixed Models”. en. In: *Biometrics* 59.4, pp. 762–769. ISSN: 1541-0420. DOI: <https://doi.org/10.1111/j.0006-341X.2003.00089.x>.
- CHENG, Feixiong and Zhongming ZHAO (2014). “Machine learning-based prediction of drug-drug interactions by integrating drug phenotypic, therapeutic, chemical, and genomic properties”. In: *Journal of the American Medical Informatics Association* 21.e2, e278–e286. ISSN: 1067-5027. DOI: 10.1136/amiajnl-2013-002512.
- CHENG, S.h., C. AUGUSTIN, *et al.* (2018). “Using machine learning to advance synthesis and use of conservation and environmental evidence”. In: *Conservation Biology* 32.4, pp. 762–764. ISSN: 1523-1739. DOI: 10.1111/cobi.13117.
- CHERNOZHUKOV, Victor *et al.* (2018). *Double/debiased machine learning for treatment and structural parameters*.
- CHESSON, Peter (2000). “Mechanisms of Maintenance of Species Diversity”. In: *Annual Review of Ecology and Systematics* 31.1, pp. 343–366. DOI: 10.1146/annurev.ecolsys.31.1.343.
- CHIB, Siddhartha and Edward GREENBERG (June 1998). “Analysis of multivariate probit models”. en. In: *Biometrika* 85.2, pp. 347–361. ISSN: 0006-3444. DOI: 10.1093/biomet/85.2.347.
- CHOLLET, François, JJ ALLAIRE, *et al.* (2017). “R Interface to Keras”. In: tex.publisher: GitHub.
- CHRISTIN, Sylvain, Éric HERVET, and Nicolas LECOMTE (2019). “Applications for deep learning in ecology”. en. In: *Methods in Ecology and Evolution* 10.10, pp. 1632–1644. ISSN: 2041-210X. DOI: <https://doi.org/10.1111/2041-210X.13256>.
- CIRILLO, Davide *et al.* (June 2020). “Sex and gender differences and biases in artificial intelligence for biomedicine and healthcare”. en. In: *npj Digital Medicine* 3.1. Number: 1 Publisher: Nature Publishing Group, pp. 1–11. ISSN: 2398-6352. DOI: 10.1038/s41746-020-0288-5.
- CLAESEN, Marc and Bart DE MOOR (2015). “Hyperparameter search in machine learning”. In: *arXiv preprint arXiv:1502.02127*.
- CLARK, James S. (2005). “Why environmental scientists are becoming Bayesians”. en. In: *Ecology Letters* 8.1, pp. 2–14. ISSN: 1461-0248. DOI: 10.1111/j.1461-0248.2004.00702.x.
- CLARK, James S., Alan E. GELFAND, *et al.* (2014). “More than the sum of the parts: forest climate response from joint species distribution models”. en. In: *Ecological Applications* 24.5, pp. 990–999. ISSN: 1939-5582. DOI: 10.1890/13-1015.1.
- CLARK, James S., Diana NEMERGUT, *et al.* (2017). “Generalized joint attribute modeling for biodiversity analysis: median-zero, multivariate, multifarious data”. en. In: *Ecological Monographs* 87.1, pp. 34–56. ISSN: 1557-7015. DOI: 10.1002/ecm.1241.
- COMO, F. *et al.* (Jan. 2017). “Predicting acute contact toxicity of pesticides in honeybees (*Apis mellifera*) through a k-nearest neighbor model”. en. In: *Chemosphere* 166, pp. 438–444. ISSN: 0045-6535. DOI: 10.1016/j.chemosphere.2016.09.092.
- COTTENIE, Karl (Nov. 2005). “Integrating environmental and spatial processes in ecological community dynamics: Meta-analysis of metacommunities”. en. In: *Ecology Letters* 8.11, pp. 1175–1182. ISSN: 1461023X. DOI: 10.1111/j.1461-0248.2005.00820.x.
- CRISTESCU, Melania E. (Oct. 2014). “From barcoding single individuals to metabarcoding biological communities: towards an integrative approach to the study of global biodiversity”. en. In: *Trends in Ecology & Evolution* 29.10, pp. 566–571. ISSN: 0169-5347. DOI: 10.1016/j.tree.2014.08.001.
- CRISTIANINI, Nello, John SHAWE-TAYLOR, *et al.* (2000). *An introduction to support vector machines and other kernel-based learning methods*. Cambridge university press.
- D’ASCOLI, Stéphane *et al.* (Jan. 2022). “Deep Symbolic Regression for Recurrent Sequences”. In: *arXiv:2201.04600 [cs]*. arXiv: 2201.04600.

- DALLAS, Tad, Andrew W. PARK, and John M. DRAKE (Feb. 2017). “Predictability of helminth parasite host range using information on geography, host traits and parasite community structure”. en. In: *Parasitology* 144.2. Publisher: Cambridge University Press, pp. 200–205. ISSN: 0031-1820, 1469-8161. DOI: 10.1017/S0031182016001608.
- DALLAS, Tad, Sadie Jane RYAN, *et al.* (Nov. 2021). *Predicting the tripartite network of mosquito-borne disease*. en-us. DOI: 10.32942/osf.io/xzmp8.
- DANIELI-SILVA, Aline *et al.* (2012). “Do pollination syndromes cause modularity and predict interactions in a pollination network in tropical high-altitude grasslands?” In: *Oikos* 121.1, pp. 35–43. ISSN: 1600-0706.
- DAVIES, Alex *et al.* (Dec. 2021). “Advancing mathematics by guiding human intuition with AI”. en. In: *Nature* 600.7887, pp. 70–74. ISSN: 1476-4687. DOI: 10.1038/s41586-021-04086-x.
- DE PALMA, Giacomo, Bobak Toussi KIANI, and Seth LLOYD (Oct. 2019). “Random deep neural networks are biased towards simple functions”. In: *arXiv:1812.10156 [cond-mat, physics:math-ph, physics:quant-ph, stat]*. arXiv: 1812.10156.
- DEHLING, D. Matthias, Pedro JORDANO, *et al.* (Jan. 2016). “Morphology predicts species’ functional roles and their degree of specialization in plant-frugivore interactions”. In: *Proc Biol Sci* 283.1823.
- DEHLING, D. Matthias, Till TÖPFER, *et al.* (2014). “Functional relationships beyond species richness patterns: trait matching in plant-bird mutualisms across scales”. In: *Global Ecology and Biogeography* 23.10, pp. 1085–1093. ISSN: 1466-8238.
- DEINER, Kristy *et al.* (2017). “Environmental DNA metabarcoding: Transforming how we survey animal and plant communities”. en. In: *Molecular Ecology* 26.21, pp. 5872–5895. ISSN: 1365-294X. DOI: 10.1111/mec.14350.
- DENEU, Benjamin *et al.* (Apr. 2021). “Convolutional neural networks improve species distribution modelling by capturing the spatial structure of the environment”. en. In: *PLOS Computational Biology* 17.4. Publisher: Public Library of Science, e1008856. ISSN: 1553-7358. DOI: 10.1371/journal.pcbi.1008856.
- DERKARABETIAN, Shahan *et al.* (2019). “A demonstration of unsupervised machine learning in species delimitation”. In: *Molecular phylogenetics and evolution* 139, p. 106562.
- DERSIMONIAN, Rebecca and Nan LAIRD (Sept. 1986). “Meta-Analysis in Clinical Trials”. en. In: *Controlled Clinical Trials* 7.3, pp. 177–188. ISSN: 01972456. DOI: 10.1016/0197-2456(86)90046-2.
- DESJARDINS-PROULX, Philippe *et al.* (Aug. 2017). “Ecological interactions and the Netflix problem”. In: *PeerJ* 5. Ed. by Yuriy ORLOV, e3644. ISSN: 2167-8359.
- DESJONQUÈRES, Camille, Toby GIFFORD, and Simon LINKE (2019). “Passive acoustic monitoring as a potential tool to survey animal and ecosystem processes in freshwater environments”. en. In: *Freshwater Biology* 0.0 (). ISSN: 1365-2427. DOI: 10.1111/fwb.13356.
- DÍAZ, Sandra *et al.* (2013). “Functional traits, the phylogeny of function, and ecosystem service vulnerability”. en. In: *Ecology and Evolution* 3.9, pp. 2958–2975. ISSN: 2045-7758. DOI: 10.1002/ece3.601.
- DIETTERICH, Thomas G *et al.* (2012). “Machine learning for computational sustainability.” In: *IGCC*, p. 1.
- DIETTERICH, Thomas G. (2000). “Ensemble Methods in Machine Learning”. en. In: *Multiple Classifier Systems*. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, pp. 1–15. ISBN: 978-3-540-45014-6. DOI: 10.1007/3-540-45014-9\_1.
- DISTLER, Trisha *et al.* (May 2015). “Stacked species distribution models and macroecological models provide congruent projections of avian species richness under climate change”. en. In: *Journal of Biogeography* 42.5. Ed. by Richard LADLE, pp. 976–988. ISSN: 03050270. DOI: 10.1111/jbi.12479.
- DIXON, Philip (May 2016). “SHOULD BLOCKS BE FIXED OR RANDOM?” en. In: *Conference on Applied Statistics in Agriculture*. ISSN: 2475-7772. DOI: 10.4148/2475-7772.1474.

- DORMANN, Carsten F., Maria BOBROWSKI, *et al.* (2018). “Biotic interactions in species distribution modelling: 10 questions to guide interpretation and avoid false conclusions”. en. In: *Global Ecology and Biogeography* 27.9, pp. 1004–1016. ISSN: 1466-8238. DOI: 10.1111/geb.12759.
- DORMANN, Carsten F., Justin M. CALABRESE, *et al.* (2018). “Model averaging in ecology: a review of Bayesian, information-theoretic, and tactical approaches for predictive inference”. en. In: *Ecological Monographs* 88.4, pp. 485–504. ISSN: 1557-7015. DOI: 10.1002/ecm.1309.
- DORMANN, Carsten F., Jane ELITH, *et al.* (Jan. 2013). “Collinearity: a review of methods to deal with it and a simulation study evaluating their performance”. In: *Ecography* 36.1, pp. 27–46. ISSN: 0906-7590. DOI: 10.1111/j.1600-0587.2012.07348.x.
- DORMANN, Carsten F., Jochen FRÜND, and H. Martin SCHAEFER (2017). “Identifying Causes of Patterns in Ecological Networks: Opportunities and Limitations”. In: *Annual Review of Ecology, Evolution, and Systematics* 48.1, pp. 559–584. DOI: 10.1146/annurev-ecolsys-110316-022928.
- DORMANN, Carsten F., Stanislaus J. SCHYMANSKI, *et al.* (2012). “Correlation and process in species distribution models: bridging a dichotomy”. en. In: *Journal of Biogeography* 39.12, pp. 2119–2131. ISSN: 1365-2699. DOI: 10.1111/j.1365-2699.2011.02659.x.
- DORMANN, Carsten F. and Rouven STRAUSS (2014). “A method for detecting modules in quantitative bipartite networks”. en. In: *Methods in Ecology and Evolution* 5.1, pp. 90–98. ISSN: 2041-210X. DOI: 10.1111/2041-210X.12139.
- DRAKE, John M., Christophe RANDIN, and Antoine GUISAN (2006). “Modelling ecological niches with support vector machines”. en. In: *Journal of Applied Ecology* 43.3, pp. 424–432. ISSN: 1365-2664. DOI: 10.1111/j.1365-2664.2006.01141.x.
- DUHART, Clement *et al.* (2019). “Deep learning for wildlife conservation and restoration efforts”. In: *36th International Conference on Machine Learning, Long Beach*. Vol. 5.
- DUNKER, Susanne *et al.* (2020). “Pollen analysis using multispectral imaging flow cytometry and deep learning”. en. In: *New Phytologist* n/a.n/a. ISSN: 1469-8137. DOI: <https://doi.org/10.1111/nph.16882>.
- DURKA, Walter and Stefan G MICHALSKI (2012). “Daphne: a dated phylogeny of a large European flora for phylogenetically informed ecological analyses: Ecological Archives E093-214”. In: *Ecology* 93.10, pp. 2297–2297.
- DUSHOFF, Jonathan, Morgan P. KAIN, and Benjamin M. BOLKER (2019). “I can see clearly now: Reinterpreting statistical significance”. en. In: *Methods in Ecology and Evolution* 10.6, pp. 756–759. ISSN: 2041-210X. DOI: 10.1111/2041-210X.13159.
- EFRON, Bradley (1992). “Bootstrap Methods: Another Look at the Jackknife”. en. In: *Breakthroughs in Statistics: Methodology and Distribution*. Ed. by Samuel KOTZ and Norman L. JOHNSON. Springer Series in Statistics. New York, NY: Springer, pp. 569–593. ISBN: 978-1-4612-4380-9. DOI: 10.1007/978-1-4612-4380-9\_41.
- EKLÖF, Anna *et al.* (2013). “The dimensionality of ecological networks”. In: *Ecol Lett* 16.5, pp. 577–583. ISSN: 1461-0248.
- ELITH, J., J. R. LEATHWICK, and T. HASTIE (2008). “A working guide to boosted regression trees”. en. In: *Journal of Animal Ecology* 77.4, pp. 802–813. ISSN: 1365-2656. DOI: 10.1111/j.1365-2656.2008.01390.x.
- ELITH, Jane and John R. LEATHWICK (2009). “Species Distribution Models: Ecological Explanation and Prediction Across Space and Time”. In: *Annual Review of Ecology, Evolution, and Systematics* 40.1, pp. 677–697. DOI: 10.1146/annurev.ecolsys.110308.120159.
- FAEGRI, K. and L. van der PIJL (1979). *The principles of pollination ecology*. ID - 19790561579. English. Ed. 3. Oxford: Pergamon Press.
- FAIRBRASS, A. J. *et al.* (2018). “CityNet - Deep Learning Tools for Urban Ecoacoustic Assessment”. In: *Methods Ecol Evol* 0.ja. ISSN: 2041-210X. DOI: 10.1111/2041-210x.13114.



- FAISAL, Ali *et al.* (Nov. 2010). “Inferring species interaction networks from species abundance data: A comparative evaluation of various statistical and machine learning methods”. en. In: *Ecological Informatics* 5.6, pp. 451–464. ISSN: 1574-9541. DOI: 10.1016/j.ecoinf.2010.06.005.
- FAIST, Akasha M, Scott FERRENBURG, and Sharon K COLLINGE (2013). “Banking on the past: seed banks as a reservoir for rare and native species in restored vernal pools”. In: *AoB Plants* 5.
- FALBEL, Daniel (2023). *luz: Higher Level 'API' for 'torch'*. R package version 0.4.0.
- FALBEL, Daniel and Javier LURASCHI (2023). *torch: Tensors and Neural Networks with 'GPU' Acceleration*. R package version 0.11.0.
- FANG, Jiansong *et al.* (Nov. 2013). “Predictions of BuChE Inhibitors Using Support Vector Machine and Naive Bayesian Classification Techniques in Drug Discovery”. In: *Journal of Chemical Information and Modeling* 53.11, pp. 3009–3020. ISSN: 1549-9596. DOI: 10.1021/ci400331p.
- FASIOLO, Matteo *et al.* (2020). “qgam: Bayesian non-parametric quantile regression modelling in R”. In: *arXiv preprint arXiv:2007.03303*.
- FEINERER, Ingo, Kurt HORNIK, and David MEYER (2008). “Text mining infrastructure in R”. In: *Journal of statistical software* 25, pp. 1–54.
- FENNER, Michael K, Michael FENNER, Ken THOMPSON, *et al.* (2005). *The ecology of seeds*. Cambridge University Press.
- FENSTER, Charles B. *et al.* (Nov. 2004). “Pollination Syndromes and Floral Specialization”. In: *Annual Review of Ecology, Evolution, and Systematics* 35.1, pp. 375–403. ISSN: 1543-592X. DOI: 10.1146/annurev.ecolsys.34.011802.132347.
- FERREIRA, André C. *et al.* (2020). “Deep learning-based methods for individual recognition in small birds”. en. In: *Methods in Ecology and Evolution* 11.9, pp. 1072–1085. ISSN: 2041-210X. DOI: <https://doi.org/10.1111/2041-210X.13436>.
- FEUERRIEGEL, Stefan, Mateusz DOLATA, and Gerhard SCHWABE (Aug. 2020). “Fair AI”. en. In: *Business & Information Systems Engineering* 62.4, pp. 379–384. ISSN: 1867-0202. DOI: 10.1007/s12599-020-00650-3.
- FISHER, Aaron, Cynthia RUDIN, and Francesca DOMINICI (2018). “All Models are Wrong but many are Useful: Variable Importance for Black-Box, Proprietary, or Misspecified Prediction Models, using Model Class Reliance”. In: *ArXiv e-prints*.
- FONTANA, Matteo, Gianluca ZENI, and Simone VANTINI (2023). “Conformal prediction: a unified review of theory and new challenges”. In: *Bernoulli* 29.1, pp. 1–23.
- FOODY, Giles M. (Sept. 1995). “Land cover classification by an artificial neural network with ancillary information”. In: *International Journal of Geographical Information Systems* 9.5. Publisher: Taylor & Francis \_eprint: <https://doi.org/10.1080/02693799508902054>, pp. 527–542. ISSN: 0269-3798. DOI: 10.1080/02693799508902054.
- FORSTMEIER, Wolfgang, Eric-Jan WAGENMAKERS, and Timothy H PARKER (2017). “Detecting and avoiding likely false-positive findings—a practical guide”. In: *Biological Reviews* 92.4, pp. 1941–1968.
- FOURCADE, Yoan, Aurélien G BESNARD, and Jean SECONDI (2018). “Paintings predict the distribution of species, or the challenge of selecting environmental predictors and evaluation statistics”. In: *Global Ecology and Biogeography* 27.2, pp. 245–256.
- FRANKLE, Jonathan and Michael CARBIN (Mar. 2019). “The Lottery Ticket Hypothesis: Finding Sparse, Trainable Neural Networks”. In: *arXiv:1803.03635 [cs]*. arXiv: 1803.03635.
- FRENCH, M and F RECKNAGEL (1970). “Modeling of algal blooms in freshwaters using artificial neural networks”. In: *WIT Transactions on Ecology and the Environment* 6. Publisher: WIT Press.
- FREUND, Yoav and Robert E SCHAPIRE (Aug. 1997). “A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting”. en. In: *Journal of Computer and System Sciences* 55.1, pp. 119–139. ISSN: 0022-0000. DOI: 10.1006/jcss.1997.1504.

- FRIEDMAN, Jerome, Trevor HASTIE, and Robert TIBSHIRANI (2010). “Regularization Paths for Generalized Linear Models via Coordinate Descent”. en. In: *Journal of Statistical Software* 33.1. ISSN: 1548-7660. DOI: 10.18637/jss.v033.i01.
- FRIEDMAN, Jerome H. (2001). “Greedy Function Approximation: A Gradient Boosting Machine”. In: *The Annals of Statistics* 29.5, pp. 1189–1232. ISSN: 00905364.
- FRIEDMAN, Jerome H. and Bogdan E. POPESCU (Sept. 2008). “Predictive learning via rule ensembles”. en. In: *Ann. Appl. Stat.* 2.3, pp. 916–954. ISSN: 1932-6157.
- FRICTSCH, Stefan, Frauke GUENTHER, and Marvin N. WRIGHT (2019). *neuralnet: Training of Neural Networks*. R package version 1.44.2.
- FRITZLER, Andreas, Sven KOITKA, and Christoph M FRIEDRICH (2017). “Recognizing Bird Species in Audio Files Using Transfer Learning”. en. In: *LEF (Working Notes)*, p. 14.
- FRØSLEV, Tobias Guldberg *et al.* (May 2019). “Man against machine: Do fungal fruitbodies and eDNA give similar biodiversity assessments across broad environmental gradients?” en. In: *Biological Conservation* 233, pp. 201–212. ISSN: 0006-3207. DOI: 10.1016/j.biocon.2019.02.038.
- FRYDA, Tomas *et al.* (2023). *h2o: R Interface for the 'H2O' Scalable Machine Learning Platform*. R package version 3.42.0.2.
- FUKUSHIMA, Kunihiko (Apr. 1980). “Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position”. en. In: *Biological Cybernetics* 36.4, pp. 193–202. ISSN: 1432-0770. DOI: 10.1007/BF00344251.
- FUNES, Guillermo *et al.* (2003). “Seed bank dynamics in tall-tussock grasslands along an altitudinal gradient”. In: *Journal of Vegetation Science* 14.2, pp. 253–258.
- GALIANA, Nuria *et al.* (2018). “The spatial scaling of species interaction networks”. In: *Nature Ecology & Evolution* 2.5, pp. 782–790. ISSN: 2397-334X.
- GALLIEN, Laure *et al.* (2012). “Invasive species distribution models – how violating the equilibrium assumption can create new insights”. en. In: *Global Ecology and Biogeography* 21.11, pp. 1126–1136. ISSN: 1466-8238. DOI: 10.1111/j.1466-8238.2012.00768.x.
- GANAIE, M. A. *et al.* (Apr. 2021). “Ensemble deep learning: A review”. In: *arXiv:2104.02395 [cs]*. arXiv: 2104.02395.
- GARIBALDI, Lucas A. *et al.* (2015). “REVIEW: Trait matching of flower visitors and crops predicts fruit set better than trait diversity”. In: *J Appl Ecol* 52.6, pp. 1436–1444. ISSN: 1365-2664.
- GELMAN, A and J HILL (2007). *Data analysis using regression and multilevel/hierarchical models*. Cambridge university press. ISBN: 0-521-86706-1.
- GELMAN, Andrew (Feb. 2005a). “Analysis of variance—why it is more important than ever”. en. In: *The Annals of Statistics* 33.1, pp. 1–53. ISSN: 0090-5364. DOI: 10.1214/009053604000001048.
- (2005b). *Why I don't use the term "fixed and random effects"*.
- GELMAN, Andrew and Eric LOKEN (2014). “The statistical crisis in science data-dependent analysis—a “garden of forking paths”—explains why many statistically significant comparisons don't hold up”. In: *American scientist* 102.6. Publisher: Sigma Xi, The Scientific Research Society, p. 460.
- GENUER, Robin, Jean-Michel POGGI, and Christine TULEAU-MALOT (Oct. 2010). “Variable selection using random forests”. en. In: *Pattern Recognition Letters* 31.14, pp. 2225–2236. ISSN: 0167-8655. DOI: 10.1016/j.patrec.2010.03.014.
- GERMAIN, Rachel M., Margaret M. MAYFIELD, and Benjamin GILBERT (Aug. 2018). “The ‘filtering’ metaphor revisited: competition and environment jointly structure invasibility and coexistence”. In: *Biology Letters* 14.8. Publisher: Royal Society, p. 20180460. DOI: 10.1098/rsbl.2018.0460.
- GIESSELMANN, Marco and Alexander W. SCHMIDT-CATRAN (Apr. 2020). “Interactions in Fixed Effects Regression Models”. en. In: *Sociological Methods & Research*. Publisher: SAGE Publications Inc, p. 0049124120914934. ISSN: 0049-1241. DOI: 10.1177/0049124120914934.

- GILBERT, Benjamin and Joseph R. BENNETT (2010). “Partitioning variation in ecological communities: do the numbers add up?” en. In: *Journal of Applied Ecology* 47.5, pp. 1071–1082. ISSN: 1365-2664. DOI: 10.1111/j.1365-2664.2010.01861.x.
- GIORIA, Margherita, Johannes J LE ROUX, *et al.* (2019). “Characteristics of the soil seed bank of invasive and non-invasive plants in their native and alien distribution range”. In: *Biological Invasions* 21.7, pp. 2313–2332.
- GIORIA, Margherita, Petr PYŠEK, *et al.* (2020). “Phylogenetic relatedness mediates persistence and density of soil seed banks”. In: *Journal of Ecology* 108.5, pp. 2121–2131.
- GOLDING, Nick (May 2015). *Mosquito community data for Golding et al. 2015 (Parasites & Vectors)*. DOI: 10.6084/m9.figshare.1420528.v1.
- (Aug. 2019). “greta: simple and scalable statistical modelling in R”. en. In: *Journal of Open Source Software* 4.40, p. 1601. ISSN: 2475-9066. DOI: 10.21105/joss.01601.
- GOLDING, Nick and David J. HARRIS (2015). *BayesComm: Bayesian community ecology analysis*.
- GOMES, Dylan G. E. (Apr. 2021). “Including random effects in statistical models in ecology: fewer than five levels?” en. In: *bioRxiv*. Publisher: Cold Spring Harbor Laboratory Section: New Results, p. 2021.04.11.439357. DOI: 10.1101/2021.04.11.439357.
- GONZÁLEZ, Martín Ana M., Bo DALSGAARD, and Jens M. OLESEN (Mar. 2010). “Centrality measures and the importance of generalist species in pollination networks”. In: *Ecological Complexity* 7.1, pp. 36–43. ISSN: 1476-945X. DOI: 10.1016/j.ecocom.2009.03.008.
- GOODFELLOW, Ian, Yoshua BENGIO, and Aaron COURVILLE (2016). *Deep learning*. MIT press.
- GRAVEL, Dominique *et al.* (2013). “Inferring food web structure from predator–prey body size relationships”. en. In: *Methods in Ecology and Evolution* 4.11, pp. 1083–1090. ISSN: 2041-210X. DOI: 10.1111/2041-210X.12103.
- GRAVING, Jacob M. *et al.* (Apr. 2019). “Fast and robust animal pose estimation”. en. In: *bioRxiv*, p. 620245. DOI: 10.1101/620245.
- GRAY, Patrick C. *et al.* (2019). “A convolutional neural network for detecting sea turtles in drone imagery”. en. In: *Methods in Ecology and Evolution* 10.3, pp. 345–355. ISSN: 2041-210X. DOI: 10.1111/2041-210X.13132.
- GREEN, Peter and Catriona J. MACLEOD (2016). “SIMR: an R package for power analysis of generalized linear mixed models by simulation”. en. In: *Methods in Ecology and Evolution* 7.4, pp. 493–498. ISSN: 2041-210X. DOI: <https://doi.org/10.1111/2041-210X.12504>.
- GREENLAND, Sander (2003). “Quantifying Biases in Causal Models: Classical Confounding vs Collider-Stratification Bias”. In: *Epidemiology* 14.3. Publisher: Lippincott Williams & Wilkins, pp. 300–306. ISSN: 1044-3983.
- GREGORUTTI, Baptiste, Bertrand MICHEL, and Philippe SAINT-PIERRE (2017). “Correlation and variable importance in random forests”. In: *Statistics and Computing* 27, pp. 659–678.
- GRIME, J Philip (2006). *Plant strategies, vegetation processes, and ecosystem properties*. John Wiley & Sons.
- GRIMM, Volker *et al.* (Nov. 2005). “Pattern-Oriented Modeling of Agent-Based Complex Systems: Lessons from Ecology”. In: *Science* 310.5750. Publisher: American Association for the Advancement of Science, pp. 987–991. DOI: 10.1126/science.1116681.
- GROOT, R. S. de *et al.* (Sept. 2010). “Challenges in integrating the concept of ecosystem services and values in landscape planning, management and decision making”. en. In: *Ecological Complexity. Ecosystem Services – Bridging Ecology, Economy and Social Sciences* 7.3, pp. 260–272. ISSN: 1476-945X. DOI: 10.1016/j.ecocom.2009.10.006.
- GUALTIERI, J. Anthony and Robert F. CROMP (Jan. 1999). “Support vector machines for hyperspectral remote sensing classification”. In: *27th AIPR Workshop: Advances in Computer-Assisted Recognition*. Vol. 3584. SPIE, pp. 221–232. DOI: 10.1117/12.339824.

- GUIDOTTI, Riccardo *et al.* (2018). “A survey of methods for explaining black box models”. In: *ACM Computing Surveys (CSUR)* 51.5, p. 93.
- GUIRADO, Emilio *et al.* (Jan. 2018). “Automatic whale counting in satellite images with deep learning”. In: *bioRxiv*. DOI: 10.1101/443671.
- GUNASEKARA, Fiona Imlach *et al.* (Feb. 2014). “Fixed effects analysis of repeated measures data”. In: *International Journal of Epidemiology* 43.1, pp. 264–269. ISSN: 0300-5771. DOI: 10.1093/ije/dyt221.
- HAIJIVASSILIOU, Vassilis A and Paul A RUUD (1994). “Classical estimation methods for LDV models using simulation”. In: *Handbook of econometrics* 4. Publisher: Elsevier, pp. 2383–2441.
- HAN, Barbara A. *et al.* (June 2015). “Rodent reservoirs of future zoonotic diseases”. In: *Proceedings of the National Academy of Sciences* 112.22. Publisher: Proceedings of the National Academy of Sciences, pp. 7039–7044. DOI: 10.1073/pnas.1501598112.
- HARDT, Moritz *et al.* (2016). “Equality of Opportunity in Supervised Learning”. In: *Advances in Neural Information Processing Systems*. Vol. 29. Curran Associates, Inc.
- HARPER, John L *et al.* (1977). “Population biology of plants.” In: *Population biology of plants*.
- HARRIS, David J. (2015). “Generating realistic assemblages with a joint species distribution model”. en. In: *Methods in Ecology and Evolution* 6.4, pp. 465–473. ISSN: 2041-210X. DOI: 10.1111/2041-210X.12332.
- HARRIS, David J., Shawn D. TAYLOR, and Ethan P. WHITE (Feb. 2018). “Forecasting biodiversity in breeding birds using best practices”. en. In: *PeerJ* 6, e4278. ISSN: 2167-8359. DOI: 10.7717/peerj.4278.
- HARRISON, Xavier A. (Oct. 2014). “Using observation-level random effects to model overdispersion in count data in ecology and evolution”. en. In: *PeerJ* 2, e616. ISSN: 2167-8359. DOI: 10.7717/peerj.616.
- (July 2015). “A comparison of observation-level random effect and Beta-Binomial models for modelling overdispersion in Binomial data in ecology & evolution”. en. In: *PeerJ* 3. Publisher: PeerJ Inc., e1114. ISSN: 2167-8359. DOI: 10.7717/peerj.1114.
- HARRISON, Xavier A. *et al.* (May 2018). “A brief introduction to mixed effects modelling and multi-model inference in ecology”. en. In: *PeerJ* 6, e4794. ISSN: 2167-8359. DOI: 10.7717/peerj.4794.
- HARTIG, Florian (2019). “DHARMA: residual diagnostics for hierarchical (multi-level/mixed) regression models”. In: *R package version 0.2 4*.
- HARTIG, Florian, Nerea ABREGO, *et al.* (2024). “Novel community data in ecology-properties and prospects”. In: *Trends in Ecology & Evolution*.
- HARTIG, Florian and Frédéric BARRAQUAND (2022). “The evidence contained in the P-value is context dependent”. In: *Trends in ecology & evolution*, S0169–5347.
- HARTIG, Florian, Justin M. CALABRESE, *et al.* (Aug. 2011). “Statistical inference for stochastic simulation models - theory and application: Inference for stochastic simulation models”. en. In: *Ecology Letters* 14.8, pp. 816–827. ISSN: 1461023X. DOI: 10.1111/j.1461-0248.2011.01640.x.
- HAUENSTEIN, Severin *et al.* (2019). “Calibrating an individual-based movement model to predict functional connectivity for little owls”. en. In: *Ecological Applications* 29.4, e01873. ISSN: 1939-5582. DOI: 10.1002/eap.1873.
- HE, Kaiming, Georgia GKIOXARI, *et al.* (Oct. 2017). “Mask R-CNN”. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.
- HE, Kaiming, Xiangyu ZHANG, *et al.* (Dec. 2015). “Deep Residual Learning for Image Recognition”. In: *arXiv:1512.03385 [cs]*. arXiv: 1512.03385.
- HE, Tong, Marten HEIDEMEYER, *et al.* (Apr. 2017). “SimBoost: a read-across approach for predicting drug–target binding affinities using gradient boosting machines”. In: *Journal of Cheminformatics* 9.1, p. 24. ISSN: 1758-2946. DOI: 10.1186/s13321-017-0209-z.

- HEDGES, Larry V. and Jack L. VEVEA (1998). “Fixed- and random-effects models in meta-analysis”. In: *Psychological Methods* 3.4. Place: US Publisher: American Psychological Association, pp. 486–504. ISSN: 1939-1463(Electronic),1082-989X(Print). DOI: 10.1037/1082-989X.3.4.486.
- HOERL, Arthur E. and Robert W. KENNARD (Feb. 1970). “Ridge Regression: Biased Estimation for Nonorthogonal Problems”. In: *Technometrics* 12.1. Publisher: Taylor & Francis \_eprint: <https://www.tandfonline.com/doi/pdf/10.1080/00401706.1970.10488634>, pp. 55–67. ISSN: 0040-1706. DOI: 10.1080/00401706.1970.10488634.
- HOFFMAN, Matthew D, Andrew GELMAN, *et al.* (2014). “The No-U-Turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo.” In: *J. Mach. Learn. Res.* 15.1, pp. 1593–1623.
- HÖLZEL, Norbert and Annette OTTE (2004). “Assessing soil seed bank persistence in flood-meadows: The search for reliable traits”. In: *Journal of Vegetation Science* 15.1, pp. 93–100.
- HONDA, Yukio (2008). “Ecological correlations between the persistence of the soil seed bank and several plant traits, including seed dormancy”. In: *Plant Ecology* 196.2, pp. 301–309.
- HONNAY, Olivier *et al.* (2008). “Can a seed bank maintain the genetic variation in the above ground plant population?” In: *Oikos* 117.1, pp. 1–5.
- HOOKE, Giles and Lucas MENTCH (May 2019). “Please Stop Permuting Features: An Explanation and Alternatives”. In: *arXiv:1905.03151 [cs, stat]*. arXiv: 1905.03151.
- HOOKE, Giles, Lucas MENTCH, and Siyu ZHOU (2021). “Unrestricted permutation forces extrapolation: variable importance requires at least one more model, or there is no free variable importance”. In: *Statistics and Computing* 31, pp. 1–16.
- HOX, J, M MOERBEEK, and R van de SCHOOT (2017). *Multilevel Analysis: Techniques and Applications*. en. 3rd edition. Quantitative methodology series. Routledge/Taylor & Francis Group.
- HU, Jun *et al.* (Feb. 2016). “GPCR–drug interactions prediction using random forest with drug-association-matrix-based post-processing procedure”. In: *Computational Biology and Chemistry* 60, pp. 59–71. ISSN: 1476-9271. DOI: 10.1016/j.compbiolchem.2015.11.007.
- HUETTMANN, Falk (2018). “Machine Learning for ‘Strategic Conservation and Planning’: Patterns, Applications, Thoughts and Urgently Needed Global Progress for Sustainability”. en. In: *Machine Learning for Ecology and Sustainable Natural Resource Management*. Ed. by Grant HUMPHRIES, Dawn R. MAGNESS, and Falk HUETTMANN. Cham: Springer International Publishing, pp. 315–333. ISBN: 978-3-319-96978-7. DOI: 10.1007/978-3-319-96978-7\_16.
- HUH, Minyoung *et al.* (Oct. 2021). “The Low-Rank Simplicity Bias in Deep Networks”. In: *arXiv:2103.10427*. arXiv: 2103.10427.
- HUI, Francis K. C. (2016). “boral – Bayesian Ordination and Regression Analysis of Multivariate Abundance Data in r”. en. In: *Methods in Ecology and Evolution* 7.6, pp. 744–750. ISSN: 2041-210X. DOI: 10.1111/2041-210X.12514.
- HUMPHREYS, John M. *et al.* (2019). “Seasonal occurrence and abundance of dabbling ducks across the continental United States: Joint spatio-temporal modelling for the Genus *Anas*”. en. In: *Diversity and Distributions* 25.9, pp. 1497–1508. ISSN: 1472-4642. DOI: 10.1111/ddi.12960.
- HUMPHRIES, Grant RW, Dawn R MAGNESS, and Falk HUETTMANN (2018). *Machine learning for ecology and sustainable natural resource management*. Springer.
- IDALINE, Laigle *et al.* (Feb. 2018). “Species traits as drivers of food web structure”. In: *Oikos* 127.2, pp. 316–326. ISSN: 0030-1299. DOI: 10.1111/oik.04712.
- INGRAM, Martin, Damjan VUKCEVIC, and Nick GOLDING (2020). “Multi-output Gaussian processes for species distribution modelling”. en. In: *Methods in Ecology and Evolution* 11.12, pp. 1587–1598. ISSN: 2041-210X. DOI: <https://doi.org/10.1111/2041-210X.13496>.
- JAHN, Najko (2021). *europemc: R Interface to the Europe PubMed Central RESTful Web Service*.
- JANZEN, Daniel H. (Dec. 1985). *On Ecological Fitting*. en.

- JANZING, Dominik, Lenon MINORICS, and Patrick BLÖBAUM (2020). “Feature relevance quantification in explainable AI: A causal problem”. In: *International Conference on artificial intelligence and statistics*. PMLR, pp. 2907–2916.
- JENSEN, Tobias *et al.* (Jan. 2020). “Employing Machine Learning for Detection of Invasive Species using Sentinel-2 and AVIRIS Data: The Case of Kudzu in the United States”. en. In: *Sustainability* 12.9. Number: 9 Publisher: Multidisciplinary Digital Publishing Institute, p. 3544. DOI: 10.3390/su12093544.
- JIMÉNEZ-ALFARO, Borja *et al.* (2016). “Seed germination traits can contribute better to plant community ecology”. In: *Journal of Vegetation Science* 27.3, pp. 637–645.
- JIN, Yi and Hong QIAN (2019). “V. PhyloMaker: an R package that can generate very large phylogenies for vascular plants”. In: *Ecography* 42.8, pp. 1353–1359.
- JOHNSON, Jerald B. and Kristian S. OMLAND (Feb. 2004). “Model selection in ecology and evolution”. In: *Trends in Ecology & Evolution* 19.2, pp. 101–108. ISSN: 0169-5347. DOI: 10.1016/j.tree.2003.10.013.
- JOHNSON, Paul C. D., Sarah J. E. BARRY, *et al.* (2015). “Power analysis for generalized linear mixed models in ecology and evolution”. en. In: *Methods in Ecology and Evolution* 6.2, pp. 133–142. ISSN: 2041-210X. DOI: <https://doi.org/10.1111/2041-210X.12306>.
- JORDAN, M. I. and T. M. MITCHELL (July 2015). “Machine learning: Trends, perspectives, and prospects”. en. In: *Science* 349.6245, pp. 255–260. ISSN: 0036-8075, 1095-9203. DOI: 10.1126/science.aaa8415.
- JORDANO, Pedro, Jordi BASCOMPTE, and Jens M. OLESEN (2003). “Invariant properties in coevolutionary networks of plant–animal interactions”. en. In: *Ecology Letters* 6.1, pp. 69–81. ISSN: 1461-0248. DOI: 10.1046/j.1461-0248.2003.00403.x.
- JOSEPH, Maxwell B. (2020a). “Neural hierarchical models of ecological populations”. en. In: *Ecology Letters* 23.4, pp. 734–747. ISSN: 1461-0248. DOI: 10.1111/ele.13462.
- (2020b). “Neural hierarchical models of ecological populations”. en. In: *Ecology Letters* n/a.n/a (). ISSN: 1461-0248. DOI: 10.1111/ele.13462.
- JUMPER, John *et al.* (Aug. 2021). “Highly accurate protein structure prediction with AlphaFold”. en. In: *Nature* 596.7873. Bandiera\_abtest: a Cc\_license\_type: cc\_by Cg\_type: Nature Research Journals Number: 7873 Primary\_atype: Research Publisher: Nature Publishing Group Subject\_term: Computational biophysics;Machine learning;Protein structure predictions;Structural biology Subject\_term\_id: computational-biophysics;machine-learning;protein-structure-predictions;structural-biology, pp. 583–589. ISSN: 1476-4687. DOI: 10.1038/s41586-021-03819-2.
- JUNG, Kiju *et al.* (2014). “Female hurricanes are deadlier than male hurricanes”. In: *Proceedings of the National Academy of Sciences* 111.24, pp. 8782–8787.
- KADANE, Joseph B (2020). *Principles of uncertainty*. Chapman and Hall/CRC.
- KALIN ARROYO, Mary T *et al.* (1999). “Persistent soil seed bank and standing vegetation at a high alpine site in the central Chilean Andes”. In: *Oecologia* 119.1, pp. 126–132.
- KARNIADAKIS, George Em *et al.* (May 2021). “Physics-informed machine learning”. en. In: *Nature Reviews Physics*. Publisher: Nature Publishing Group, pp. 1–19. ISSN: 2522-5820. DOI: 10.1038/s42254-021-00314-5.
- KE, Nan Rosemary *et al.* (2019). “Learning neural causal models from unknown interventions”. In: *arXiv preprint arXiv:1910.01075*.
- KIM, Byungju *et al.* (June 2019). “Learning Not to Learn: Training Deep Neural Networks With Biased Data”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- KISSLING, W. Daniel and Matthias SCHLEUNING (2015). “Multispecies interactions across trophic levels at macroscales: retrospective and future directions”. en. In: *Ecography* 38.4, pp. 346–357. ISSN: 1600-0587. DOI: 10.1111/ecog.00819.

- KISSLING, W. Daniel, Ramona WALLS, *et al.* (Oct. 2018). “Towards global data products of Essential Biodiversity Variables on species traits”. En. In: *Nature Ecology & Evolution* 2.10, p. 1531. ISSN: 2397-334X. DOI: 10.1038/s41559-018-0667-3.
- KLEYER, Michael *et al.* (2008). “The LEDA Traitbase: a database of life-history traits of the Northwest European flora”. In: *Journal of ecology* 96.6, pp. 1266–1274.
- KOH, Pang Wei *et al.* (2021). “Wilds: A benchmark of in-the-wild distribution shifts”. In: *International Conference on Machine Learning*. PMLR, pp. 5637–5664.
- KÖNIG, Christian *et al.* (2021). “Scale dependency of joint species distribution models challenges interpretation of biotic interactions”. en. In: *Journal of Biogeography* 48.7, pp. 1541–1551. ISSN: 1365-2699. DOI: 10.1111/jbi.14106.
- KÖRNER, Christian (2007). “The use of ‘altitude’ in ecological research”. In: *Trends in ecology & evolution* 22.11, pp. 569–574.
- KRAFT, Nathan J. B. *et al.* (2015). “Community assembly, coexistence and the environmental filtering metaphor”. en. In: *Functional Ecology* 29.5, pp. 592–599. ISSN: 1365-2435. DOI: <https://doi.org/10.1111/1365-2435.12345>.
- KRAPU, Christopher and Mark BORSUK (2020). “A spatial community regression approach to exploratory analysis of ecological data”. en. In: *Methods in Ecology and Evolution* n/a.n/a. ISSN: 2041-210X. DOI: 10.1111/2041-210X.13371.
- KRAWCZYK, Bartosz (Nov. 2016). “Learning from imbalanced data: open challenges and future directions”. In: *Progress in Artificial Intelligence* 5.4, pp. 221–232. ISSN: 2192-6360.
- KRISTENSEN, Kasper *et al.* (2016). “TMB: Automatic Differentiation and Laplace Approximation”. In: *Journal of Statistical Software* 70.5, pp. 1–21. DOI: 10.18637/jss.v070.i05.
- KRIZHEVSKY, Alex, Ilya SUTSKEVER, and Geoffrey E HINTON (2012). “ImageNet Classification with Deep Convolutional Neural Networks”. In: *Advances in Neural Information Processing Systems* 25. Ed. by F. PEREIRA *et al.* Curran Associates, Inc., pp. 1097–1105.
- KRUEGER, Charlene and Lili TIAN (Oct. 2004). “A Comparison of the General Linear Mixed Model and Repeated Measures ANOVA Using a Dataset with Multiple Missing Data Points”. en. In: *Biological Research For Nursing* 6.2. Publisher: SAGE Publications, pp. 151–157. ISSN: 1099-8004. DOI: 10.1177/1099800404267682.
- KUHN, Lorenz, Clare LYLE, *et al.* (Mar. 2021). “Robustness to Pruning Predicts Generalization in Deep Neural Networks”. In: *arXiv:2103.06002 [cs, stat]*. arXiv: 2103.06002.
- KUHN, Max and Daniel FALBEL (2022). *brulee: High-Level Modeling Functions with ‘torch’*. R package version 0.2.0.
- KÜNZEL, Sören R *et al.* (2019). “Metalearners for estimating heterogeneous treatment effects using machine learning”. In: *Proceedings of the national academy of sciences* 116.10, pp. 4156–4165.
- KUZNETSOVA, Alexandra, P. BROCKHOFF, and R. CHRISTENSEN (2017). “lmerTest Package: Tests in Linear Mixed Effects Models”. In: *Journal of Statistical Software, Articles* 82.13. DOI: 10.18637/JSS.V082.I13.
- KWOK, Roberta (Mar. 2019). “AI empowers conservation biology”. EN. In: *Nature* 567, p. 133. DOI: 10.1038/d41586-019-00746-1.
- LAARHOVEN, Twan van and Elena MARCHIORI (June 2013). “Predicting Drug-Target Interactions for New Drug Compounds Using a Weighted Nearest Neighbor Profile”. In: *PLOS ONE* 8.6, e66952. DOI: 10.1371/journal.pone.0066952.
- LAIRD, N. M. and J. H. WARE (Dec. 1982). “Random-effects models for longitudinal data”. eng. In: *Biometrics* 38.4, pp. 963–974. ISSN: 0006-341X.
- LAM, Remi *et al.* (2023). “Learning skillful medium-range global weather forecasting”. In: *Science* 0.0, eadi2336. DOI: 10.1126/science.adi2336. eprint: <https://www.science.org/doi/pdf/10.1126/science.adi2336>.

- LANDOLT, Elias *et al.* (2010). *Flora indicativa: Okologische Zeigerwerte und biologische Kennzeichen zur Flora der Schweiz und der Alpen*. Haupt.
- LANG, Michel *et al.* (2019). “mlr3: A modern object-oriented machine learning framework in R”. In: *Journal of Open Source Software*. DOI: 10.21105/joss.01903.
- LASSECK, Mario (2018). “Audio-based bird species identification with deep convolutional neural networks”. In: *Working Notes of CLEF 2018*.
- LAUBACH, Zachary M *et al.* (2021). “A biologist’s guide to model selection and causal inference”. In: *Proceedings of the Royal Society B* 288.1943, p. 20202815.
- LE GUILLARME, Nicolas and Wilfried THUILLER (2022). “TaxoNERD: Deep neural models for the recognition of taxonomic entities in the ecological and evolutionary literature”. en. In: *Methods in Ecology and Evolution* 13.3, pp. 625–641. ISSN: 2041-210X. DOI: 10.1111/2041-210X.13778.
- LECUN, Y. *et al.* (Nov. 1998). “Gradient-based learning applied to document recognition”. In: *Proceedings of the IEEE* 86.11, pp. 2278–2324. ISSN: 0018-9219. DOI: 10.1109/5.726791.
- LECUN, Yann, Yoshua BENGIO, and Geoffrey HINTON (2015). “Deep learning”. In: *Nature* 521, p. 436.
- LEDERER, David J. *et al.* (Sept. 2018). “Control of Confounding and Reporting of Results in Causal Inference Studies. Guidance for Authors from Editors of Respiratory, Sleep, and Critical Care Journals”. In: *Annals of the American Thoracic Society* 16.1. Publisher: American Thoracic Society - AJRCCM, pp. 22–28. ISSN: 2329-6933. DOI: 10.1513/AnnalsATS.201808-564PS.
- LEIBOLD, M. A., M. HOLYOAK, *et al.* (2004). “The metacommunity concept: a framework for multi-scale community ecology”. en. In: *Ecology Letters* 7.7, pp. 601–613. ISSN: 1461-0248. DOI: 10.1111/j.1461-0248.2004.00608.x.
- LEIBOLD, Mathew A and Jonathan M CHASE (2017). *Metacommunity ecology*. Vol. 59. Princeton University Press.
- LEIBOLD, Mathew A and Gregory M MIKKELSON (2002). “Coherence, species turnover, and boundary clumping: elements of meta-community structure”. In: *Oikos* 97.2. Publisher: Wiley Online Library, pp. 237–250.
- LEIBOLD, Mathew A., F. Javiera RUDOLPH, *et al.* (2022). “The internal structure of metacommunities”. In: *Oikos* 2022.1. DOI: <https://doi.org/10.1111/oik.08618>. eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1111/oik.08618>.
- LENTH, Russell V. (2021). *emmeans: Estimated Marginal Means, aka Least-Squares Means*.
- LEVINE, Jonathan M. *et al.* (June 2017). “Beyond pairwise mechanisms of species coexistence in complex communities”. en. In: *Nature* 546.7656, pp. 56–64. ISSN: 1476-4687. DOI: 10.1038/nature22898.
- LI, Zhiyuan, Yuping LUO, and Kaifeng LYU (Apr. 2021). “Towards Resolving the Implicit Bias of Gradient Descent for Matrix Factorization: Greedy Low-Rank Learning”. In: *arXiv:2012.09839 [cs, stat]*. arXiv: 2012.09839.
- LIAKOS, Konstantinos G. *et al.* (Aug. 2018). “Machine Learning in Agriculture: A Review”. en. In: *Sensors* 18.8. Number: 8 Publisher: Multidisciplinary Digital Publishing Institute, p. 2674. ISSN: 1424-8220. DOI: 10.3390/s18082674.
- LINDSTROM, Mary J and Douglas M BATES (1988). “Newton-Raphson and EM Algorithms for Linear Mixed-Effects Models for Repeated-Measures Data”. en. In: *Journal of the American Statistical Association* 83.404, pp. 1014–1022.
- LITTELL, Ramon C. (Dec. 2002). “Analysis of unbalanced mixed model data: A case study comparison of ANOVA versus REML/GLS”. en. In: *Journal of Agricultural, Biological, and Environmental Statistics* 7.4, pp. 472–490. ISSN: 1085-7117, 1537-2693. DOI: 10.1198/108571102816.
- LIU, Shengyu, Buzhou TANG, *et al.* (2016). *Drug-Drug Interaction Extraction via Convolutional Neural Networks*. en. Research article. DOI: 10.1155/2016/6918381.
- LIU, Suyun and Luis Nunes VICENTE (Aug. 2021). *The Sharpe predictor for fairness in machine learning*. arXiv:2108.06415 [cs] version: 1. DOI: 10.48550/arXiv.2108.06415.



- LIU, Udayangani, Tiziana Antonella COSSU, and John B DICKIE (2019). “Royal Botanic Gardens, Kew’s Seed Information Database (SID): A compilation of taxon-based biological seed characteristics or traits.” In: *Biodiversity Information Science and Standards* 1.
- LIU, Yinhan, Myle OTT, *et al.* (July 2019). “RoBERTa: A Robustly Optimized BERT Pretraining Approach”. In: *arXiv:1907.11692 [cs]*. arXiv: 1907.11692.
- LIU, Ze, Yutong LIN, *et al.* (2021). “Swin transformer: Hierarchical vision transformer using shifted windows”. In: *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 10012–10022.
- LIU, Zhuang, Hanzi MAO, *et al.* (2022). “A convnet for the 2020s”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11976–11986.
- LONG, Rowena L *et al.* (2015). “The ecophysiology of seed persistence: a mechanistic view of the journey to germination or demise”. In: *Biological Reviews* 90.1, pp. 31–59.
- LOQUERCIO, Antonio, Mattia SEGU, and Davide SCARAMUZZA (2020). “A general framework for uncertainty estimation in deep learning”. In: *IEEE Robotics and Automation Letters* 5.2, pp. 3153–3160.
- LUCAS, Tim C. D. (2020). “A translucent box: interpretable machine learning in ecology”. en. In: *Ecological Monographs* 90.4, e01422. ISSN: 1557-7015. DOI: <https://doi.org/10.1002/ecm.1422>.
- LUNDBERG, Scott M, Gabriel G ERION, and Su-In LEE (2018). “Consistent individualized feature attribution for tree ensembles”. In: *arXiv preprint arXiv:1802.03888*.
- LUNDBERG, Scott M and Su-In LEE (2017). “A unified approach to interpreting model predictions”. In: *Advances in neural information processing systems* 30.
- LÜRIG, Moritz D *et al.* (2021). “Computer vision, machine learning, and the promise of phenomics in ecology and evolutionary biology”. In: *Frontiers in Ecology and Evolution* 9, p. 642774.
- MAAS, Cora J. M. and Joop J. HOX (Jan. 2005). “Sufficient Sample Sizes for Multilevel Modeling”. en. In: *Methodology* 1.3, pp. 86–92. ISSN: 1614-1881, 1614-2241. DOI: 10.1027/1614-2241.1.3.86.
- MAC AODHA, Oisín *et al.* (2018). “Bat detective—Deep learning tools for bat acoustic signal detection”. In: *PLOS Computational Biology* 14.3, e1005995. DOI: 10.1371/journal.pcbi.1005995.
- MÄDER, Patrick *et al.* (2021). “The Flora Incognita app – Interactive plant species identification”. en. In: *Methods in Ecology and Evolution* 12.7, pp. 1335–1342. ISSN: 2041-210X. DOI: 10.1111/2041-210X.13611.
- MAGLIANESI, María A. *et al.* (2015). “Functional structure and specialization in three tropical plant–hummingbird interaction networks across an elevational gradient in Costa Rica”. en. In: *Ecography* 38.11, pp. 1119–1128. ISSN: 1600-0587. DOI: 10.1111/ecog.01538.
- MAGLIANESI, María Alejandra *et al.* (2014). “Morphological traits determine specialization and resource use in plant–hummingbird networks in the neotropics”. en. In: *Ecology* 95.12, pp. 3325–3334. ISSN: 1939-9170. DOI: 10.1890/13-2261.1.
- MAINALI, Kumar P. *et al.* (2015). “Projecting future expansion of invasive species: comparing and improving methodologies for species distribution modeling”. en. In: *Global Change Biology* 21.12, pp. 4464–4480. ISSN: 1365-2486. DOI: 10.1111/gcb.13038.
- MARKE, T *et al.* (2013). “The Berchtesgaden National Park (Bavaria, Germany): a platform for interdisciplinary catchment research”. In: *Environmental earth sciences* 69.2, pp. 679–694.
- MARTIN, Julien G. A., Daniel H. NUSSEY, *et al.* (2011). “Measuring individual differences in reaction norms in field and experimental studies: a power analysis of random regression models”. In: *Methods in Ecology and Evolution* 2.4, pp. 362–374. DOI: <https://doi.org/10.1111/j.2041-210X.2010.00084.x>. eprint: <https://besjournals.onlinelibrary.wiley.com/doi/pdf/10.1111/j.2041-210X.2010.00084.x>.
- MARTIN, Laura J, Bernd BLOSSEY, and Erle ELLIS (2012). “Mapping where ecologists work: biases in the global distribution of terrestrial ecological observations”. en. In: *Frontiers in Ecology and the Environment* 10.4, pp. 195–201. ISSN: 1540-9309. DOI: 10.1890/110154.

- MASAHIRO, Ryo and C. RILLIG MATTHIAS (Nov. 2017). “Statistically reinforced machine learning for nonlinear patterns and variable interactions”. In: *Ecosphere* 8.11, e01976. ISSN: 2150-8925. DOI: 10.1002/ecs2.1976.
- MATHIS, Alexander *et al.* (Sept. 2018). “DeepLabCut: markerless pose estimation of user-defined body parts with deep learning”. En. In: *Nature Neuroscience* 21.9, p. 1281. ISSN: 1546-1726. DOI: 10.1038/s41593-018-0209-y.
- MATUSCHEK, Hannes *et al.* (June 2017). “Balancing Type I error and power in linear mixed models”. en. In: *Journal of Memory and Language* 94, pp. 305–315. ISSN: 0749-596X. DOI: 10.1016/j.jml.2017.01.001.
- MAXIMILIANPI (Dec. 2022). *MaximilianPi/Pichler-and-Hartig-2022: Publication*. Version v1.0. DOI: 10.5281/zenodo.7433226.
- MAYFIELD, Margaret M. and Daniel B. STOUFFER (Feb. 2017). “Higher-order interactions capture unexplained complexity in diverse communities”. In: *Nature Ecology & Evolution* 1, p. 0062.
- McCULLOCH, Warren S. and Walter PITTS (Dec. 1943). “A logical calculus of the ideas immanent in nervous activity”. en. In: *The bulletin of mathematical biophysics* 5.4, pp. 115–133. ISSN: 1522-9602. DOI: 10.1007/BF02478259.
- McINTIRE, Eliot J. B. *et al.* (2022). “PERFICT: A Re-imagined foundation for predictive ecology”. en. In: *Ecology Letters*. ISSN: 1461-0248. DOI: 10.1111/ele.13994.
- McLEAN, Robert A., William L. SANDERS, and Walter W. STROUP (Feb. 1991). “A Unified Approach to Mixed Linear Models”. In: *The American Statistician* 45.1. Publisher: Taylor & Francis \_eprint: <https://www.tandfonline.com/doi/pdf/10.1080/00031305.1991.10475767>, pp. 54–64. ISSN: 0003-1305. DOI: 10.1080/00031305.1991.10475767.
- McMAHON, Sean M. and Jeffrey M. DIEZ (2007). “Scales of association: hierarchical linear models and the measurement of ecological systems”. en. In: *Ecology Letters* 10.6, pp. 437–452. ISSN: 1461-0248. DOI: 10.1111/j.1461-0248.2007.01036.x.
- McNEISH, Daniel (Sept. 2017). “Small Sample Methods for Multilevel Modeling: A Colloquial Elucidation of REML and the Kenward-Roger Correction”. In: *Multivariate Behavioral Research* 52.5, pp. 661–670. ISSN: 0027-3171. DOI: 10.1080/00273171.2017.1344538.
- McNEISH, Daniel M. and Laura M. STAPLETON (June 2016). “The Effect of Small Sample Size on Two-Level Model Estimates: A Review and Illustration”. en. In: *Educational Psychology Review* 28.2, pp. 295–314. ISSN: 1040-726X, 1573-336X. DOI: 10.1007/s10648-014-9287-x.
- MELGANI, F. and L. BRUZZONE (Aug. 2004). “Classification of hyperspectral remote sensing images with support vector machines”. In: *IEEE Transactions on Geoscience and Remote Sensing* 42.8. Conference Name: IEEE Transactions on Geoscience and Remote Sensing, pp. 1778–1790. ISSN: 1558-0644. DOI: 10.1109/TGRS.2004.831865.
- MENDEN, Michael P *et al.* (2013). “Machine learning prediction of cancer cell sensitivity to drugs based on genomic and chemical properties”. In: *PLoS one* 8.4, e61318. DOI: <https://doi.org/10.1371/journal.pone.0061318>.
- METEYARD, Lotte and Robert A. I. DAVIES (June 2020). “Best practice guidance for linear mixed-effects models in psychological science”. en. In: *Journal of Memory and Language* 112, p. 104092. ISSN: 0749-596X. DOI: 10.1016/j.jml.2020.104092.
- METROPOLIS, Nicholas *et al.* (1953). “Equation of state calculations by fast computing machines”. In: *The journal of chemical physics* 21.6, pp. 1087–1092.
- MEYER, Carsten, Patrick WEIGELT, and Holger KREFT (2016). “Multidimensional biases, gaps and uncertainties in global plant occurrence information”. en. In: *Ecology Letters* 19.8, pp. 992–1006. ISSN: 1461-0248. DOI: 10.1111/ele.12624.
- MEYER, Hanna and Edzer PEBESMA (2021). “Predicting into unknown space? Estimating the area of applicability of spatial prediction models”. In: *Methods in Ecology and Evolution* 12.9, pp. 1620–1633.

- MIGNAN, Arnaud and Marco BROCCARDO (Oct. 2019). “One neuron versus deep learning in aftershock prediction”. en. In: *Nature* 574.7776, E1–E3. ISSN: 0028-0836, 1476-4687. DOI: 10.1038/s41586-019-1582-8.
- MILLAR, Russell B. and Marti J. ANDERSON (Dec. 2004). “Remedies for pseudoreplication”. en. In: *Fisheries Research* 70.2-3, pp. 397–407. ISSN: 01657836. DOI: 10.1016/j.fishres.2004.08.016.
- MITTELBAACH, Gary G. and Douglas W. SCHEMSKE (May 2015). “Ecological and evolutionary perspectives on community assembly”. en. In: *Trends in Ecology & Evolution* 30.5, pp. 241–247. ISSN: 01695347. DOI: 10.1016/j.tree.2015.02.008.
- MOLNAR, Christoph (2020). *Interpretable machine learning*. Lulu. com.
- MOLNAR, Christoph *et al.* (2020). “General pitfalls of model-agnostic interpretation methods for machine learning models”. In: *International Workshop on Extending Explainable AI Beyond Deep Models and Classifiers*. Springer, pp. 39–68.
- MOMAL, Raphaëlle, Stéphane ROBIN, and Christophe AMBROISE (Feb. 2020). “Tree-based Inference of Species Interaction Networks from Abundance Data”. en. In: *Methods in Ecology and Evolution*, pp. 2041–210X.13380. ISSN: 2041-210X, 2041-210X. DOI: 10.1111/2041-210X.13380.
- MORA, Bernat Bramon *et al.* (July 2018). “Identifying a common backbone of interactions underlying food webs from different ecosystems”. En. In: *Nature Communications* 9.1, p. 2603. ISSN: 2041-1723. DOI: 10.1038/s41467-018-05056-0.
- MORADI, Sohrab *et al.* (Feb. 2019). “Identifying high-priority conservation areas for avian biodiversity using species distribution modeling”. en. In: *Ecological Indicators* 97, pp. 159–164. ISSN: 1470-160X. DOI: 10.1016/j.ecolind.2018.10.003.
- MORALES-CASTILLA, Ignacio *et al.* (June 2015). “Inferring biotic interactions from proxies”. In: *Trends in Ecology & Evolution* 30.6, pp. 347–356. ISSN: 0169-5347. DOI: 10.1016/j.tree.2015.03.014.
- MOUNTRAKIS, Giorgos, Jungho IM, and Caesar OGOLE (May 2011). “Support vector machines in remote sensing: A review”. en. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 66.3, pp. 247–259. ISSN: 0924-2716. DOI: 10.1016/j.isprsjprs.2010.11.001.
- MUFF, Stefanie *et al.* (Mar. 2022). “Rewriting results sections in the language of evidence”. en. In: *Trends in Ecology & Evolution* 37.3, pp. 203–210. ISSN: 0169-5347. DOI: 10.1016/j.tree.2021.10.009.
- MURDOCH, W James *et al.* (2019). “Definitions, methods, and applications in interpretable machine learning”. In: *Proceedings of the National Academy of Sciences* 116.44, pp. 22071–22080.
- MURÚA, M. and A. ESPÍNDOLA (2015). “Pollination syndromes in a specialised plant-pollinator interaction: does floral morphology predict pollinators in *Calceolaria*?” en. In: *Plant Biology* 17.2, pp. 551–557. ISSN: 1438-8677. DOI: 10.1111/plb.12225.
- MUSILA, Simon *et al.* (2024). “Occurrence records of mammal species in Tana River Basin, Kenya”. eng. In: (). DOI: 10.15468/0msz3d.
- NAKAGAWA, Shinichi and Holger SCHIELZETH (2013). “A general and simple method for obtaining R<sup>2</sup> from generalized linear mixed-effects models”. en. In: *Methods in Ecology and Evolution* 4.2, pp. 133–142. ISSN: 2041-210X. DOI: 10.1111/j.2041-210x.2012.00261.x.
- NAKKIRAN, Preetum *et al.* (Dec. 2019). “Deep Double Descent: Where Bigger Models and More Data Hurt”. In: *arXiv:1912.02292 [cs, stat]*. arXiv: 1912.02292.
- NAVARRO, Rene (2024). “Kenya Virtual Museum Records”. eng. In: (). DOI: 10.15468/jolylt.
- NIKU, Jenni, Wesley BROOKS, *et al.* (2020). *gllvm: Generalized linear latent variable models*.
- NIKU, Jenni, Francis K. C. HUI, *et al.* (2019). “gllvm: Fast analysis of multivariate abundance data with generalized linear latent variable models in r”. en. In: *Methods in Ecology and Evolution* 10.12, pp. 2173–2182. ISSN: 2041-210X. DOI: 10.1111/2041-210X.13303.
- NORBERG, Anna *et al.* (2019). “A comprehensive evaluation of predictive performance of 33 species distribution models at species and community levels”. en. In: *Ecological Monographs* 89.3, e01370. ISSN: 1557-7015. DOI: 10.1002/ecm.1370.

- NOROUZZADEH, Mohammad Sadegh, Dan MORRIS, *et al.* (2021). “A deep active learning system for species identification and counting in camera trap images”. In: *Methods in ecology and evolution* 12.1, pp. 150–161.
- NOROUZZADEH, Mohammad Sadegh, Anh NGUYEN, *et al.* (June 2018). “Automatically identifying, counting, and describing wild animals in camera-trap images with deep learning”. en. In: *Proceedings of the National Academy of Sciences* 115.25, E5716–E5725. ISSN: 0027-8424, 1091-6490. DOI: 10.1073/pnas.1719367115.
- NOVAK, Roman *et al.* (Dec. 2019). “Neural Tangents: Fast and Easy Infinite Neural Networks in Python”. In: *arXiv:1912.02803 [cs, stat]*. arXiv: 1912.02803.
- NUGENT, Joshua R. and Ken P. KLEINMAN (Apr. 2021). “Type I error control for cluster randomized trials under varying small sample structures”. eng. In: *BMC medical research methodology* 21.1, p. 65. ISSN: 1471-2288. DOI: 10.1186/s12874-021-01236-7.
- OBERDORFER, Erich (1949). *Pflanzensoziologische Exkursionsflora für Südwestdeutschland und die angrenzenden Gebiete*. deu. Tech. rep. E. Ulmer.
- OLHEDE, S. C. and P. J. WOLFE (Sept. 2018). “The growing ubiquity of algorithms in society: implications, impacts and innovations”. In: *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 376.2128, p. 20170364. DOI: 10.1098/rsta.2017.0364.
- OLITO, Colin and Jeremy W. FOX (2015). “Species traits and abundances predict metrics of plant–pollinator network structure, but not pairwise interactions”. en. In: *Oikos* 124.4, pp. 428–436. ISSN: 1600-0706. DOI: 10.1111/oik.01439.
- OLLERTON, Jeff *et al.* (June 2009). “A global test of the pollination syndrome hypothesis”. In: *Annals of Botany* 103.9, pp. 1471–1480. ISSN: 0305-7364. DOI: 10.1093/aob/mcp031.
- OOI, Hong (2021). *glmnetUtils: Utilities for 'Glmnet'*. R package version 1.1.8.
- OOI, Mark KJ (2012). “Seed bank persistence and climate change”. In: *Seed Science Research* 22.S1, S53–S60.
- ORTEGA, M, C LEVASSOR, and B PECO (1997). “Seasonal dynamics of Mediterranean pasture seed banks along environmental gradients”. In: *Journal of Biogeography* 24.2, pp. 177–195.
- OTT, Tankred and Ulrich LAUTENSCHLAGER (2022). “GinJinn2: Object detection and segmentation for ecology and evolution”. en. In: *Methods in Ecology and Evolution* 13.3, pp. 603–610. ISSN: 2041-210X. DOI: 10.1111/2041-210X.13787.
- OVASKAINEN, Otso, Nerea ABREGO, *et al.* (2016). “Using latent variable models to identify large networks of species-to-species associations at different spatial scales”. en. In: *Methods in Ecology and Evolution* 7.5, pp. 549–555. ISSN: 2041-210X. DOI: 10.1111/2041-210X.12501.
- OVASKAINEN, Otso, Jenni HOTTOLA, and Juha SIITONEN (2010). “Modeling species co-occurrence by multivariate logistic regression generates new hypotheses on fungal interactions”. en. In: *Ecology* 91.9, pp. 2514–2521. ISSN: 1939-9170. DOI: 10.1890/10-0173.1.
- OVASKAINEN, Otso, Gleb TIKHONOV, *et al.* (May 2017). “How are species interactions structured in species-rich communities? A new method for analysing time-series data”. In: *Proceedings of the Royal Society B: Biological Sciences* 284.1855. Publisher: Royal Society, p. 20170768. DOI: 10.1098/rspb.2017.0768.
- PASZKE, Adam, Sam GROSS, Soumith CHINTALA, *et al.* (2017). “Automatic differentiation in PyTorch”. In: *NIPS autodiff workshop*.
- PASZKE, Adam, Sam GROSS, Francisco MASSA, *et al.* (2019). “PyTorch: An Imperative Style, High-Performance Deep Learning Library”. In: *Advances in Neural Information Processing Systems* 32. Ed. by H. WALLACH *et al.* Curran Associates, Inc., pp. 8024–8035.
- PEARL, Judea (2009). *Causality*. Cambridge university press.
- (Feb. 2019). “The seven tools of causal inference, with reflections on machine learning”. en. In: *Communications of the ACM* 62.3, pp. 54–60. ISSN: 0001-0782, 1557-7317. DOI: 10.1145/3241036.

- (Jan. 2021). “Radical empiricism and machine learning research”. en. In: *Journal of Causal Inference* 9.1. Publisher: De Gruyter, pp. 78–82. ISSN: 2193-3685. DOI: 10.1515/jci-2021-0006.
- PEARL, Judea and Dana MACKENZIE (2018). *The book of why: the new science of cause and effect*. Basic books.
- PEARSE, Ian S. and Florian ALTERMATT (2013). “Predicting novel trophic interactions in a non-native world”. In: *Ecol Lett* 16.8, pp. 1088–1094. ISSN: 1461-0248.
- PENONE, Caterina *et al.* (2014). “Imputation of missing data in life-history trait datasets: which approach performs the best?” en. In: *Methods in Ecology and Evolution* 5.9, pp. 961–970. ISSN: 2041-210X. DOI: 10.1111/2041-210X.12232.
- PEREIRA, Talmo D. *et al.* (Jan. 2019). “Fast animal pose estimation using deep neural networks”. En. In: *Nature Methods* 16.1, p. 117. ISSN: 1548-7105. DOI: 10.1038/s41592-018-0234-5.
- PERES-NETO, Pedro R. and Pierre LEGENDRE (2010). “Estimating and controlling for spatial structure in the study of ecological communities”. en. In: *Global Ecology and Biogeography* 19.2, pp. 174–184. ISSN: 1466-8238. DOI: 10.1111/j.1466-8238.2009.00506.x.
- PHARTYAL, Shyam S *et al.* (2020). “Ready for change: Seed traits contribute to the high adaptability of mudflat species to their unpredictable habitat”. In: *Journal of Vegetation Science* 31.2, pp. 331–342.
- PICCIULIN, Marta *et al.* (2019). “Listening to the unseen: Passive acoustic monitoring reveals the presence of a cryptic fish species”. en. In: *Aquatic Conservation: Marine and Freshwater Ecosystems* 29.2, pp. 202–210. ISSN: 1099-0755. DOI: 10.1002/aqc.2973.
- PICHLER, Maximilian, Virginie BOREUX, *et al.* (2020). “Machine learning algorithms to infer trait-matching and predict species interactions in ecological networks”. en. In: *Methods in Ecology and Evolution* 11.2, pp. 281–293. ISSN: 2041-210X. DOI: <https://doi.org/10.1111/2041-210X.13329>.
- PICHLER, Maximilian and Florian HARTIG (2021a). “A new joint species distribution model for faster and more accurate inference of species associations from big community data”. en. In: *Methods in Ecology and Evolution* 12.11, pp. 2159–2173. ISSN: 2041-210X. DOI: 10.1111/2041-210X.13687.
- (July 2021b). *Pichler & Hartig 2021 - A new joint species distribution model for faster and more accurate inference of species associations from big community data*. DOI: 10.5281/zenodo.5131594.
- (2023a). “Can predictive models be used for causal inference?” In: *arXiv preprint arXiv:2306.10551*.
- (2023b). “Machine learning and deep learning—A review for ecologists”. In: *Methods in Ecology and Evolution* 14.4, pp. 994–1016.
- PIMM, Stuart L. *et al.* (Nov. 2015). “Emerging Technologies to Conserve Biodiversity”. In: *Trends in Ecology & Evolution* 30.11, pp. 685–696. ISSN: 0169-5347. DOI: 10.1016/j.tree.2015.08.008.
- PINHEIRO, José C. and Douglas M. BATES (Mar. 1995). “Approximations to the Log-Likelihood Function in the Nonlinear Mixed-Effects Model”. en. In: *Journal of Computational and Graphical Statistics* 4.1, pp. 12–35. ISSN: 1061-8600, 1537-2715. DOI: 10.1080/10618600.1995.10474663.
- PLUE, Jan and Sara AO COUSINS (2018). “Seed dispersal in both space and time is necessary for plant diversity maintenance in fragmented landscapes”. In: *Oikos* 127.6, pp. 780–791.
- PLUE, Jan, Hans VAN CALSTER, *et al.* (2021). “Buffering effects of soil seed banks on plant community composition in response to land use and climate”. In: *Global Ecology and Biogeography* 30.1, pp. 128–139.
- POGGIATO, Giovanni *et al.* (Feb. 2021). “On the Interpretations of Joint Modeling in Community Ecology”. en. In: *Trends in Ecology & Evolution*. ISSN: 0169-5347. DOI: 10.1016/j.tree.2021.01.002.
- POISOT, Timothée, Daniel B. STOUFFER, and Dominique GRAVEL (2015). “Beyond species: why ecological interaction networks vary through space and time”. en. In: *Oikos* 124.3, pp. 243–251. ISSN: 1600-0706. DOI: 10.1111/oik.01719.
- POL, Martijn van de (2012). “Quantifying individual variation in reaction norms: how study design affects the accuracy, precision and power of random regression models”. en. In: *Methods in*

- Ecology and Evolution* 3.2, pp. 268–280. ISSN: 2041-210X. DOI: <https://doi.org/10.1111/j.2041-210X.2011.00160.x>.
- POLLOCK, Laura J. *et al.* (2014). “Understanding co-occurrence by modelling species simultaneously with a Joint Species Distribution Model (JSDM)”. en. In: *Methods in Ecology and Evolution* 5.5, pp. 397–406. ISSN: 2041-210X. DOI: 10.1111/2041-210X.12180.
- POMERANZ, Justin P. F. *et al.* (2019). “Inferring predator–prey interactions in food webs”. en. In: *Methods in Ecology and Evolution* 10.3, pp. 356–367. ISSN: 2041-210X. DOI: 10.1111/2041-210X.13125.
- PONTARP, Mikael *et al.* (Mar. 2019). “The Latitudinal Diversity Gradient: Novel Understanding through Mechanistic Eco-evolutionary Models”. en. In: *Trends in Ecology & Evolution* 34.3, pp. 211–223. ISSN: 0169-5347. DOI: 10.1016/j.tree.2018.11.009.
- POPOVIC, Gordana C *et al.* (2019). “Untangling direct species associations from indirect mediator species effects with graphical models”. In: *Methods in Ecology and Evolution* 10.9. Publisher: Wiley Online Library, pp. 1571–1583.
- POSCHLOD, Peter *et al.* (2013). “Seed ecology and assembly rules in plant communities”. In: *Vegetation ecology* 2, pp. 164–202.
- POTTS, Simon G. *et al.* (Dec. 2016). “Safeguarding pollinators and their values to human well-being”. en. In: *Nature* 540.7632, pp. 220–229. ISSN: 1476-4687. DOI: 10.1038/nature20588.
- PROBERT, Robin J, Matthew I DAWES, and Fiona R HAY (2009). “Ecological correlates of ex situ seed longevity: a comparative study on 195 species”. In: *Annals of botany* 104.1, pp. 57–69.
- PYŠEK, Petr *et al.* (May 2008). “Geographical and taxonomic biases in invasion ecology”. en. In: *Trends in Ecology & Evolution* 23.5, pp. 237–244. ISSN: 0169-5347. DOI: 10.1016/j.tree.2008.02.002.
- QI, Di and Andrew J MAJDA (2020). “Using machine learning to predict extreme events in complex systems”. In: *Proceedings of the National Academy of Sciences* 117.1, pp. 52–59.
- QUESTAGAME (2023). “Earth Guardians Weekly Feed”. eng. In: DOI: 10.15468/slqqt8.
- R CORE TEAM (2019). *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing.
- (2021). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria.
  - (2022). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria.
- RACKAUCKAS, Christopher *et al.* (Nov. 2021). “Universal Differential Equations for Scientific Machine Learning”. In: *arXiv:2001.04385 [cs, math, q-bio, stat]*. arXiv: 2001.04385.
- RADFORD, Alec *et al.* (2019). “Language models are unsupervised multitask learners”. In: *OpenAI blog* 1.8, p. 9.
- RADHAKRISHNAN, Adityanarayanan, Mikhail BELKIN, and Caroline UHLER (2023). “Wide and deep neural networks achieve consistency for classification”. In: *Proceedings of the National Academy of Sciences* 120.14, e2208779120.
- RAMMER, Werner and Rupert SEIDL (2019). “A scalable model of vegetation transitions using deep neural networks”. en. In: *Methods in Ecology and Evolution* 0.0. ISSN: 2041-210X. DOI: 10.1111/2041-210X.13171.
- RAVURI, Suman *et al.* (Sept. 2021). “Skilful precipitation nowcasting using deep generative models of radar”. en. In: *Nature* 597.7878. Number: 7878 Publisher: Nature Publishing Group, pp. 672–677. ISSN: 1476-4687. DOI: 10.1038/s41586-021-03854-z.
- RAYHAN, Farshid *et al.* (Dec. 2017). “iDTI-ESBoost: Identification of Drug Target Interaction Using Evolutionary and Structural Features with Boosting”. En. In: *Scientific Reports* 7.1, p. 17731. ISSN: 2045-2322. DOI: 10.1038/s41598-017-18025-2.

- REICHSTEIN, Markus *et al.* (Feb. 2019). “Deep learning and process understanding for data-driven Earth system science”. en. In: *Nature* 566.7743, pp. 195–204. ISSN: 1476-4687. DOI: 10.1038/s41586-019-0912-1.
- RENNE, Ian J and Benjamin F TRACY (2007). “Disturbance persistence in managed grasslands: shifts in aboveground community structure and the weed seed bank”. In: *Plant Ecology* 190.1, pp. 71–80.
- REW, Jehyeok *et al.* (Jan. 2019). “Animal Movement Prediction Based on Predictive Recurrent Neural Network”. en. In: *Sensors* 19.20. Number: 20 Publisher: Multidisciplinary Digital Publishing Institute, p. 4411. ISSN: 1424-8220. DOI: 10.3390/s19204411.
- RIBEIRO, Marco Tulio, 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. event-place: San Francisco, California, USA. New York, NY, USA: ACM, pp. 1135–1144. ISBN: 978-1-4503-4232-2. DOI: 10.1145/2939672.2939778.
- ROBERTS, David R. *et al.* (2017). “Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure”. en. In: *Ecography* 40.8, pp. 913–929. ISSN: 1600-0587. DOI: 10.1111/ecog.02881.
- ROBINSON, Peter M (1988). “Root-N-consistent semiparametric regression”. In: *Econometrica: Journal of the Econometric Society*, pp. 931–954.
- RODGERS, Amie D. *et al.* (Apr. 2010). “Modeling Liver-Related Adverse Effects of Drugs Using kNearest Neighbor Quantitative Structure-Activity Relationship Method”. In: *Chemical Research in Toxicology* 23.4, pp. 724–732. ISSN: 0893-228X. DOI: 10.1021/tx900451r.
- ROMERO, M. Pilar *et al.* (Mar. 2021). “A comparison of the value of two machine learning predictive models to support bovine tuberculosis disease control in England”. en. In: *Preventive Veterinary Medicine* 188, p. 105264. ISSN: 0167-5877. DOI: 10.1016/j.prevetmed.2021.105264.
- ROSAS-GUERRERO, Víctor *et al.* (2014a). “A quantitative review of pollination syndromes: do floral traits predict effective pollinators?” In: *Ecol Lett* 17.3, pp. 388–400. ISSN: 1461-0248.
- (2014b). “A quantitative review of pollination syndromes: do floral traits predict effective pollinators?” en. In: *Ecology Letters* 17.3, pp. 388–400. ISSN: 1461-0248. DOI: 10.1111/ele.12224.
- ROSBAKH, Sergey and Peter POSCHLOD (2021). “Plant community persistence strategy is elevation-specific”. In: *Journal of Vegetation Science* 32.3, e13028.
- ROSENBLATT, F. (1958). “The Perceptron: A Probabilistic Model for Information Storage and Organization in The Brain”. In: *Psychological Review*, pp. 65–386.
- ROY, Amédée, Ronan FABLET, and Sophie Lanco BERTRAND (2022). “Using generative adversarial networks (GAN) to simulate central-place foraging trajectories”. en. In: *Methods in Ecology and Evolution* 13.6, pp. 1275–1287. ISSN: 2041-210X. DOI: 10.1111/2041-210X.13853.
- RUDIN, Cynthia (2019). “Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead”. In: *Nature machine intelligence* 1.5, pp. 206–215.
- RYO, Masahiro *et al.* (2021). “Explainable artificial intelligence enhances the ecological interpretability of black-box species distribution models”. en. In: *Ecography* 44.2, pp. 199–205. ISSN: 1600-0587. DOI: 10.1111/ecog.05360.
- SAATKAMP, A *et al.* (2014). *Seeds: the ecology of regeneration in plant communities*.
- SAGI, Omer and Lior ROKACH (2018). “Ensemble learning: A survey”. en. In: *WIREs Data Mining and Knowledge Discovery* 8.4, e1249. ISSN: 1942-4795. DOI: 10.1002/widm.1249.
- SANG, Huiyan, Mikyoung JUN, and Jianhua Z. HUANG (Dec. 2011). “Covariance approximation for large multivariate spatial data sets with an application to multiple climate model errors”. EN. In: *The Annals of Applied Statistics* 5.4. Publisher: Institute of Mathematical Statistics, pp. 2519–2548. ISSN: 1932-6157, 1941-7330. DOI: 10.1214/11-AOAS478.

- SCHAPIRE, Robert E (1990). “The strength of weak learnability”. In: *Machine learning* 5.2. Publisher: Springer, pp. 197–227.
- SCHERER, Daniel and Christian KÖRNER (2011). “Topographically controlled thermal-habitat differentiation buffers alpine plant diversity against climate warming”. In: *Journal of biogeography* 38.2, pp. 406–416.
- SCHIELZETH, Holger (2010). “Simple means to improve the interpretability of regression coefficients”. en. In: *Methods in Ecology and Evolution* 1.2, pp. 103–113. ISSN: 2041-210X. DOI: <https://doi.org/10.1111/j.2041-210X.2010.00012.x>.
- SCHIELZETH, Holger, Niels J. DINGEMANSE, et al. (2020). “Robustness of linear mixed-effects models to violations of distributional assumptions”. en. In: *Methods in Ecology and Evolution* 11.9, pp. 1141–1152. ISSN: 2041-210X. DOI: 10.1111/2041-210X.13434.
- SCHIELZETH, Holger and Wolfgang FORSTMEIER (Mar. 2009). “Conclusions beyond support: overconfident estimates in mixed models”. en. In: *Behavioral Ecology* 20.2. Publisher: Oxford Academic, pp. 416–420. ISSN: 1045-2249. DOI: 10.1093/beheco/arn145.
- SCHLEUNING, Matthias, Jochen FRÜND, and Daniel GARCÍA (2015). “Predicting ecosystem functions from biodiversity and mutualistic networks: an extension of trait-based concepts to plant-animal interactions”. In: *Ecography* 38.4, pp. 380–392. ISSN: 1600-0587.
- SCHOLBECK, Christian A et al. (2022). “Marginal effects for non-linear prediction functions”. In: *arXiv preprint arXiv:2201.08837*.
- SCHÖLKOPF, Bernhard (Nov. 2019). “Causality for Machine Learning”. In: *arXiv:1911.10500 [cs, stat]*. arXiv: 1911.10500.
- SCHWARTZ, Roy et al. (Nov. 2020). “Green AI”. In: *Communications of the ACM* 63.12, pp. 54–63. ISSN: 0001-0782. DOI: 10.1145/3381831.
- SCHWARZ, Benjamin et al. (2021). “Within-day dynamics of plant–pollinator networks are dominated by early flower closure: An experimental test of network plasticity”. In: *Oecologia* 196.3, pp. 781–794.
- SEJNOWSKI, Terrence J. (Dec. 2020). “The unreasonable effectiveness of deep learning in artificial intelligence”. en. In: *Proceedings of the National Academy of Sciences* 117.48. Publisher: National Academy of Sciences Section: Colloquium Paper, pp. 30033–30038. ISSN: 0027-8424, 1091-6490. DOI: 10.1073/pnas.1907373117.
- SHAH, Harshay et al. (2020). “The pitfalls of simplicity bias in neural networks”. In: *arXiv preprint arXiv:2006.07710*.
- SHAVER, J. Myles (Mar. 2019). “Interpreting Interactions in Linear Fixed-Effect Regression Models: When Fixed-Effect Estimates Are No Longer Within-Effects”. In: *Strategy Science* 4.1. Publisher: INFORMS, pp. 25–40. ISSN: 2333-2050. DOI: 10.1287/stsc.2018.0065.
- SHMUELI, Galit (Aug. 2010). “To Explain or to Predict?” EN. In: *Statistical Science* 25.3, pp. 289–310. ISSN: 0883-4237, 2168-8745. DOI: 10.1214/10-STS330.
- SHWARTZ-ZIV, Ravid and Alexander A. ALEMI (Feb. 2020). “Information in Infinite Ensembles of Infinitely-Wide Neural Networks”. en. In: *Proceedings of The 2nd Symposium on Advances in Approximate Bayesian Inference*. ISSN: 2640-3498. PMLR, pp. 1–17.
- SILVER, David et al. (Oct. 2017). “Mastering the game of Go without human knowledge”. en. In: *Nature* 550.7676, pp. 354–359. ISSN: 1476-4687. DOI: 10.1038/nature24270.
- SIMMONS, Joseph P., Leif D. NELSON, and Uri SIMONSOHN (Nov. 2011). “False-Positive Psychology: Undisclosed Flexibility in Data Collection and Analysis Allows Presenting Anything as Significant”. en. In: *Psychological Science* 22.11. Publisher: SAGE Publications Inc, pp. 1359–1366. ISSN: 0956-7976. DOI: 10.1177/0956797611417632.
- SIMPSON, R. et al. (1992). “Biological pattern recognition by neural networks”. In: *Marine Ecology Progress Series* 79.3. Publisher: Inter-Research Science Center, pp. 303–308. ISSN: 0171-8630.



- SKRONDAL, Anders and Sophia RABE-HESKETH (2004). *Generalized latent variable modeling: Multi-level, longitudinal, and structural equation models*. Chapman and Hall/CRC.
- SOCOLAR, Jacob B. *et al.* (Jan. 2016). “How Should Beta-Diversity Inform Biodiversity Conservation?” In: *Trends in Ecology & Evolution* 31.1, pp. 67–80. ISSN: 0169-5347. DOI: 10.1016/j.tree.2015.11.005.
- SOFAER, Helen R. *et al.* (July 2019). “Development and Delivery of Species Distribution Models to Inform Decision-Making”. en. In: *BioScience* 69.7. Publisher: Oxford Academic, pp. 544–557. ISSN: 0006-3568. DOI: 10.1093/biosci/biz045.
- SONNEWALD, Maike *et al.* (2020). “Elucidating ecological complexity: Unsupervised learning determines global marine eco-provinces”. In: *Science advances* 6.22, eaay4740.
- SRIVASTAVA, Nitish *et al.* (2014). “Dropout: A simple way to prevent neural networks from overfitting”. In: *The Journal of Machine Learning Research* 15.1, pp. 1929–1958.
- STANG, Martina, Peter G. L. KLINKHAMER, and Eddy van der MEIJDEN (Mar. 2007). “Asymmetric specialization and extinction risk in plant–flower visitor webs: a matter of morphology or abundance?” en. In: *Oecologia* 151.3, pp. 442–453. ISSN: 1432-1939. DOI: 10.1007/s00442-006-0585-y.
- STEIN, Michael L. (June 2007). “Spatial variation of total column ozone on a global scale”. EN. In: *The Annals of Applied Statistics* 1.1. Publisher: Institute of Mathematical Statistics, pp. 191–210. ISSN: 1932-6157, 1941-7330. DOI: 10.1214/07-AOAS106.
- (May 2014). “Limitations on low rank approximations for covariance matrices of spatial data”. en. In: *Spatial Statistics*. Spatial Statistics Miami 8, pp. 1–19. ISSN: 2211-6753. DOI: 10.1016/j.spasta.2013.06.003.
- STEKHOVEN, Daniel J. and Peter BÜHLMANN (Jan. 2012). “MissForest—non-parametric missing value imputation for mixed-type data”. en. In: *Bioinformatics* 28.1, pp. 112–118. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btr597.
- STÖCKLIN, Jürg and Markus FISCHER (1999). “Plants with longer-lived seeds have lower local extinction rates in grassland remnants 1950–1985”. In: *Oecologia* 120.4, pp. 539–543.
- STONE, M. (1974). “Cross-Validatory Choice and Assessment of Statistical Predictions”. en. In: *Journal of the Royal Statistical Society: Series B (Methodological)* 36.2, pp. 111–133. ISSN: 2517-6161. DOI: 10.1111/j.2517-6161.1974.tb00994.x.
- STOWELL, Dan *et al.* (2018). “Automatic acoustic detection of birds through deep learning: the first Bird Audio Detection challenge”. In: *Methods Ecol Evol* 0.ja. ISSN: 2041-210X. DOI: 10.1111/2041-210x.13103.
- STRAM, Daniel O. and Jae Won LEE (1994). “Variance Components Testing in the Longitudinal Mixed Effects Model”. In: *Biometrics* 50.4. Publisher: [Wiley, International Biometric Society], pp. 1171–1177. ISSN: 0006-341X. DOI: 10.2307/2533455.
- STRYDOM, Tanya *et al.* (Nov. 2021). “A roadmap towards predicting species interaction networks (across space and time)”. In: *Philosophical Transactions of the Royal Society B: Biological Sciences* 376.1837. Publisher: Royal Society, p. 20210063. DOI: 10.1098/rstb.2021.0063.
- STUPARIU, Mihai-Sorin *et al.* (May 2022). “Machine learning in landscape ecological analysis: a review of recent approaches”. en. In: *Landscape Ecology* 37.5, pp. 1227–1250. ISSN: 1572-9761. DOI: 10.1007/s10980-021-01366-9.
- SUNDARARAJAN, Mukund and Amir NAJMI (Aug. 2019). “The many Shapley values for model explanation”. In: *arXiv:1908.08474 [cs, econ]*. arXiv: 1908.08474.
- SWALLOW, William H. and John F. MONAHAN (1984). “Monte Carlo Comparison of ANOVA, MIVQUE, REML, and ML Estimators of Variance Components”. en. In: *Technometrics* 26.1, pp. 47–57.
- SZEGEDY, Christian *et al.* (Dec. 2015). “Rethinking the Inception Architecture for Computer Vision”. In: *arXiv:1512.00567 [cs]*. arXiv: 1512.00567.

- TABAK, Michael A. *et al.* (2019). “Machine learning to classify animal species in camera trap images: Applications in ecology”. en. In: *Methods in Ecology and Evolution* 10.4, pp. 585–590. ISSN: 2041-210X. DOI: 10.1111/2041-210X.13120.
- TANK, Alex *et al.* (2021). “Neural Granger Causality”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence*. arXiv: 1802.05842, pp. 1–1. ISSN: 0162-8828, 2160-9292, 1939-3539. DOI: 10.1109/TPAMI.2021.3065601.
- TARI, Luis *et al.* (Sept. 2010). “Discovering drug-drug interactions: a text-mining and reasoning approach based on properties of drug metabolism”. In: *Bioinformatics* 26.18, pp. i547–i553. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btq382.
- TAUSCH, Simone *et al.* (2019). “Dormancy and endosperm presence influence the ex situ conservation potential in central European calcareous grassland plants”. In: *AoB Plants* 11.4, plz035.
- TAYLOR-RODRÍGUEZ, Daniel *et al.* (Dec. 2017). “Joint Species Distribution Modeling: Dimension Reduction Using Dirichlet Processes”. EN. In: *Bayesian Analysis* 12.4. Publisher: International Society for Bayesian Analysis, pp. 939–967. ISSN: 1936-0975, 1931-6690. DOI: 10.1214/16-BA1031.
- TEMELES, Ethan J. *et al.* (2009). “Effect of flower shape and size on foraging performance and trade-offs in a tropical hummingbird”. en. In: *Ecology* 90.5, pp. 1147–1161. ISSN: 1939-9170. DOI: 10.1890/08-0695.1.
- TER BRAAK, Cajo JF and Jasper A VRUGT (2008). “Differential evolution Markov chain with snooker updater and fewer chains”. In: *Statistics and Computing* 18, pp. 435–446.
- THOMPSON, Ken *et al.* (2003). “Are seed dormancy and persistence in soil related?” In: *Seed Science Research* 13.2, pp. 97–100.
- THUILLER, Wilfried *et al.* (2006). “Using niche-based modelling to assess the impact of climate change on tree functional diversity in Europe”. In: *Diversity and Distributions*, pp. 49–60. ISSN: 1366-9516. DOI: 10.1111/j.1366-9516.2006.00216.x@10.1111/(ISSN)1472-4642.species-distribution-models-in-conservation-biogeography.
- TIBSHIRANI, Robert (1996). “Regression Shrinkage and Selection Via the Lasso”. en. In: *Journal of the Royal Statistical Society: Series B (Methodological)* 58.1, pp. 267–288. ISSN: 2517-6161. DOI: 10.1111/j.2517-6161.1996.tb02080.x.
- TIKHONOV, Gleb, Nerea ABREGO, *et al.* (2017). “Using joint species distribution models for evaluating how species-to-species associations depend on the environmental context”. en. In: *Methods in Ecology and Evolution* 8.4, pp. 443–452. ISSN: 2041-210X. DOI: 10.1111/2041-210X.12723.
- TIKHONOV, Gleb, Li DUAN, *et al.* (2019). “Computationally efficient joint species distribution modeling of big spatial data”. en. In: *Ecology* n/a.n/a, e02929. ISSN: 1939-9170. DOI: 10.1002/ecy.2929.
- TIKHONOV, Gleb, Øystein H. OPEDAL, *et al.* (2020). “Joint species distribution modelling with the r-package Hmsc”. en. In: *Methods in Ecology and Evolution* 11.3, pp. 442–447. ISSN: 2041-210X. DOI: 10.1111/2041-210X.13345.
- TIKHONOV, Gleb, Otso OVASKAINEN, *et al.* (2019). *Hmsc: Hierarchical model of species communities*.
- TOBLER, Mathias W. *et al.* (2019). “Joint species distribution models with species correlations and imperfect detection”. en. In: *Ecology* 100.8, e02754. ISSN: 1939-9170. DOI: 10.1002/ecy.2754.
- TORNEY, Colin J. *et al.* (2019). “A comparison of deep learning and citizen science techniques for counting wildlife in aerial survey images”. en. In: *Methods in Ecology and Evolution* 10.6, pp. 779–787. ISSN: 2041-210X. DOI: <https://doi.org/10.1111/2041-210X.13165>.
- TOWNSEND, Zac *et al.* (2013). “The Choice between Fixed and Random Effects”. en. In: *The SAGE Handbook of Multilevel Modeling*. 1 Oliver’s Yard, 55 City Road, London EC1Y 1SP United Kingdom: SAGE Publications Ltd, pp. 73–88. DOI: 10.4135/9781446247600.n5.
- TREDENNICK, Andrew T. *et al.* (2021). “A practical guide to selecting models for exploration, inference, and prediction in ecology”. en. In: *Ecology* 102.6, e03336. ISSN: 1939-9170. DOI: 10.1002/ecy.3336.

- TRIMBLE, Morgan J. and Rudi J. van AARDE (2012). “Geographical and taxonomic biases in research on biodiversity in human-modified landscapes”. en. In: *Ecosphere* 3.12, art119. ISSN: 2150-8925. DOI: 10.1890/ES12-00299.1.
- TSENG, Gabriel, Hannah KERNER, and David ROLNICK (2022). “Timl: Task-informed meta-learning for agriculture”. In: *arXiv preprint arXiv:2202.02124*.
- TUIA, Devis *et al.* (Feb. 2022). “Perspectives in machine learning for wildlife conservation”. en. In: *Nature Communications* 13.1. Number: 1 Publisher: Nature Publishing Group, p. 792. ISSN: 2041-1723. DOI: 10.1038/s41467-022-27980-y.
- TYLER, Torbjörn *et al.* (2021). “Ecological indicator and traits values for Swedish vascular plants”. In: *Ecological Indicators* 120, p. 106923.
- URBAN, M. C. *et al.* (Sept. 2016). “Improving the forecast for biodiversity under climate change”. en. In: *Science* 353.6304. ISSN: 0036-8075, 1095-9203. DOI: 10.1126/science.aad8466.
- USHEY, Kevin, JJ ALLAIRE, and Yuan TANG (2019). *Reticulate: interface to 'Python'*.
- VALAVI, Roozbeh *et al.* (2019). “blockCV: An R package for generating spatially or environmentally separated folds for k-fold cross-validation of species distribution models”. In: *Methods in Ecology and Evolution* 10.2, pp. 225–232.
- VALDOVINOS, Fernanda S. (2019). “Mutualistic networks: moving closer to a predictive theory”. en. In: *Ecology Letters* 22.9, pp. 1517–1534. ISSN: 1461-0248. DOI: 10.1111/ele.13279.
- VALLE-PÉREZ, Guillermo, Chico Q. CAMARGO, and Ard A. LOUIS (Apr. 2019). “Deep learning generalizes because the parameter-function map is biased towards simple functions”. In: *arXiv:1805.08522 [cs, stat]*. arXiv: 1805.08522.
- VAN DER PUTTEN, Wim H., Mirka MACEL, and Marcel E. VISSER (July 2010). “Predicting species distribution and abundance responses to climate change: why it is essential to include biotic interactions across trophic levels”. In: *Philosophical Transactions of the Royal Society B: Biological Sciences* 365.1549, pp. 2025–2034. DOI: 10.1098/rstb.2010.0037.
- VAN EE, Justin J, Jacob S IVAN, and Mevin B HOOTEN (2022). “Community confounding in joint species distribution models”. In: *Scientific Reports* 12.1, p. 12235.
- VAN HORN, Grant *et al.* (2018). “The inaturalist species classification and detection dataset”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8769–8778.
- VANDVIK, Vigdis *et al.* (2016). “Seed banks are biodiversity reservoirs: species–area relationships above versus below ground”. In: *Oikos* 125.2, pp. 218–228.
- VAPNIK, V. and A. LERNER (1963). “Pattern recognition using generalized portrait method”. In: *Automation and Remote Control* 24, pp. 774–780.
- VAYENA, Effy, Alessandro BLASIMME, and I. Glenn COHEN (Nov. 2018). “Machine learning in medicine: Addressing ethical challenges”. en. In: *PLOS Medicine* 15.11. Publisher: Public Library of Science, e1002689. ISSN: 1549-1676. DOI: 10.1371/journal.pmed.1002689.
- VEIT, Andreas, Michael J WILBER, and Serge BELONGIE (2016). “Residual Networks Behave Like Ensembles of Relatively Shallow Networks”. In: *Advances in Neural Information Processing Systems*. Vol. 29. Curran Associates, Inc.
- VELLEND, Mark (June 2010). “Conceptual Synthesis in Community Ecology”. en. In: *The Quarterly Review of Biology* 85.2, pp. 183–206. ISSN: 0033-5770, 1539-7718. DOI: 10.1086/652373.
- VENABLE, D Lawrence and Joel S BROWN (1988). “The selective interactions of dispersal, dormancy, and seed size as adaptations for reducing risk in variable environments”. In: *The American Naturalist* 131.3, pp. 360–384.
- VENABLES, W. N. and B. D. RIPLEY (2002). *Modern Applied Statistics with S*. Fourth. ISBN 0-387-95457-0. New York: Springer.
- VIEILLEDENT, Ghislain and Jeanne CLÉMENT (2019). *jSDM: Joint species distribution models*. manual.
- VINYALS, Oriol *et al.* (2019). “Grandmaster level in StarCraft II using multi-agent reinforcement learning”. In: *Nature* 575.7782, pp. 350–354.

- VIZENTIN-BUGONI, J., P. K. MARUYAMA, and M. SAZIMA (Feb. 2014). “Processes entangling interactions in communities: forbidden links are more important than abundance in a hummingbird-plant network”. en. In: *Proceedings of the Royal Society B: Biological Sciences* 281.1780, pp. 20132397–20132397. ISSN: 0962-8452, 1471-2954. DOI: 10.1098/rspb.2013.2397.
- VOZNICA, J. *et al.* (Mar. 2021). *Deep learning from phylogenies to uncover the transmission dynamics of epidemics*. en. Company: Cold Spring Harbor Laboratory Distributor: Cold Spring Harbor Laboratory Label: Cold Spring Harbor Laboratory Section: New Results Type: article. DOI: 10.1101/2021.03.11.435006.
- WAGER, Stefan and Susan ATHEY (2018). “Estimation and inference of heterogeneous treatment effects using random forests”. In: *Journal of the American Statistical Association* 113.523, pp. 1228–1242.
- WAGER, Stefan, Sida WANG, and Percy S LIANG (2013). “Dropout training as adaptive regularization”. In: *Advances in neural information processing systems* 26.
- WAINWRIGHT, Patricia E., Scott T. LEATHERDALE, and Joel A. DUBIN (Nov. 2007). “Advantages of mixed effects models over traditional ANOVA models in developmental studies: A worked example in a mouse model of fetal alcohol syndrome”. en. In: *Developmental Psychobiology* 49.7, pp. 664–674. ISSN: 00121630, 10982302. DOI: 10.1002/dev.20245.
- WALCK, Jeffrey L *et al.* (2011). “Climate change and plant regeneration from seed”. In: *Global Change Biology* 17.6, pp. 2145–2161.
- WÄLDCHEN, Jana and Patrick MÄDER (2018). “Machine learning for image based species identification”. en. In: *Methods in Ecology and Evolution* 9.11, pp. 2216–2225. ISSN: 2041-210X. DOI: 10.1111/2041-210X.13075.
- WANG, Dongfang and Jin GU (2018). “VASC: dimension reduction and visualization of single-cell RNA-seq data by deep variational autoencoder”. In: *Genomics, proteomics & bioinformatics* 16.5, pp. 320–331.
- WANG, Shangying, Kai FAN, *et al.* (Sept. 2019). “Massive computational acceleration by using neural networks to emulate mechanism-based biological models”. en. In: *Nature Communications* 10.1, pp. 1–9. ISSN: 2041-1723. DOI: 10.1038/s41467-019-12342-y.
- WANG, Yaqing, Quanming YAO, *et al.* (2020). “Generalizing from a few examples: A survey on few-shot learning”. In: *ACM computing surveys (csur)* 53.3. Publisher: ACM New York, NY, USA, pp. 1–34.
- WARDEH, Maya, Matthew BAYLIS, and Marcus S. C. BLAGROVE (Feb. 2021). “Predicting mammalian hosts in which novel coronaviruses can be generated”. en. In: *Nature Communications* 12.1. Number: 1 Publisher: Nature Publishing Group, p. 780. ISSN: 2041-1723. DOI: 10.1038/s41467-021-21034-5.
- WARR, Susan J, Martin KENT, and Ken THOMPSON (1994). “Seed bank composition and variability in five woodlands in south-west England”. In: *Journal of Biogeography*, pp. 151–168.
- WARTON, David I. *et al.* (Dec. 2015). “So Many Variables: Joint Modeling in Community Ecology”. In: *Trends in Ecology & Evolution* 30.12, pp. 766–779. ISSN: 0169-5347. DOI: 10.1016/j.tree.2015.09.007.
- WEARN, Oliver R., Robin FREEMAN, and David M. P. JACOBY (Feb. 2019). “Responsible AI for conservation”. en. In: *Nature Machine Intelligence* 1.2. Number: 2 Publisher: Nature Publishing Group, pp. 72–73. ISSN: 2522-5839. DOI: 10.1038/s42256-019-0022-7.
- WEIN, S. *et al.* (Apr. 2021). “A graph neural network framework for causal inference in brain networks”. en. In: *Scientific Reports* 11.1. Bandiera\_abtest: a Cc\_license\_type: cc\_by Cg\_type: Nature Research Journals Number: 1 Primary\_atype: Research Publisher: Nature Publishing Group Subject\_term: Computational science;Dynamical systems;Network models Subject\_term\_id: computational-science;dynamical-systems;network-models, p. 8061. ISSN: 2045-2322. DOI: 10.1038/s41598-021-87411-8.

- WEINSTEIN, Ben G. and Catherine H. GRAHAM (2017). “Persistent bill and corolla matching despite shifting temporal resources in tropical hummingbird-plant interactions”. In: *Ecol Lett* 20.3, pp. 326–335. ISSN: 1461-0248.
- WEISS, Karl, Taghi M. KHOSHGOFTAAR, and DingDing WANG (2016). “A survey of transfer learning”. In: *Journal of Big Data* 3.1, p. 9. ISSN: 2196-1115.
- WELLING, Pirjo, Anne TOLVANEN, and Kari LAINE (2004). “The alpine soil seed bank in relation to field seedlings and standing vegetation in subarctic Finland”. In: *Arctic, Antarctic, and Alpine Research* 36.2, pp. 229–238.
- WEN, Ming *et al.* (Apr. 2017). “Deep-Learning-Based Drug–Target Interaction Prediction”. In: *Journal of Proteome Research* 16.4, pp. 1401–1409. ISSN: 1535-3893. DOI: 10.1021/acs.jproteome.6b00618.
- WESSELKAMP, Marieke *et al.* (2022). “Process-guidance improves predictive performance of neural networks for carbon turnover in ecosystems”. In: *arXiv preprint arXiv:2209.14229*.
- WIENCIERZ, Andrea, Sonja GREVEN, and Helmut KÜCHENHOFF (Jan. 2011). “Restricted likelihood ratio testing in linear mixed models with general error covariance structure”. In: *Electronic Journal of Statistics* 5.none. Publisher: Institute of Mathematical Statistics and Bernoulli Society, pp. 1718–1734. ISSN: 1935-7524, 1935-7524. DOI: 10.1214/11-EJS654.
- WILKINSON, David P. *et al.* (2019). “A comparison of joint species distribution models for presence–absence data”. en. In: *Methods in Ecology and Evolution* 10.2, pp. 198–211. ISSN: 2041-210X. DOI: 10.1111/2041-210X.13106.
- (2021). “Defining and evaluating predictions of joint species distribution models”. en. In: *Methods in Ecology and Evolution* 12.3, pp. 394–404. ISSN: 2041-210X. DOI: <https://doi.org/10.1111/2041-210X.13518>.
- WILLEMS, JH and LPM BIK (1998). “Restoration of high species density in calcareous grassland: the role of seed rain and soil seed bank”. In: *Applied Vegetation Science* 1.1, pp. 91–100.
- WILLI, Marco *et al.* (2019). “Identifying animal species in camera trap images using deep learning and citizen science”. en. In: *Methods in Ecology and Evolution* 10.1, pp. 80–91. ISSN: 2041-210X. DOI: <https://doi.org/10.1111/2041-210X.13099>.
- WISZ, Mary Susanne *et al.* (2013). “The role of biotic interactions in shaping distributions and realised assemblages of species: implications for species distribution modelling”. en. In: *Biological Reviews* 88.1, pp. 15–30. ISSN: 1469-185X. DOI: 10.1111/j.1469-185X.2012.00235.x.
- WOLF, Thomas *et al.* (2019). “Huggingface’s transformers: State-of-the-art natural language processing”. In: *arXiv preprint arXiv:1910.03771*.
- WONG, Felix *et al.* (2023). “Discovery of a structural class of antibiotics with explainable deep learning”. In: *Nature*, pp. 1–9.
- WOOD, Connor M., Ralph J. GUTIÉRREZ, and M. Zachariah PEERY (2019). “Acoustic monitoring reveals a diverse forest owl community, illustrating its potential for basic and applied ecology”. en. In: *Ecology* 100.9, e02764. ISSN: 1939-9170. DOI: 10.1002/ecy.2764.
- WOOD, Simon N (2017). *Generalized additive models: an introduction with R*. CRC press.
- WREGE, Peter H. *et al.* (2017). “Acoustic monitoring for conservation in tropical forests: examples from forest elephants”. en. In: *Methods in Ecology and Evolution* 8.10, pp. 1292–1301. ISSN: 2041-210X. DOI: 10.1111/2041-210X.12730.
- WRIGHT, Marvin N. and Andreas ZIEGLER (2017). “ranger: A Fast Implementation of Random Forests for High Dimensional Data in C++ and R”. In: *Journal of Statistical Software* 77.1, pp. 1–17. DOI: 10.18637/jss.v077.i01.
- XU, Bing *et al.* (Nov. 2015). “Empirical Evaluation of Rectified Activations in Convolutional Network”. In: *arXiv:1505.00853 [cs, stat]*. arXiv: 1505.00853.

- YAMANISHI, Yoshihiro *et al.* (July 2008). “Prediction of drug-target interaction networks from the integration of chemical and genomic spaces”. In: *Bioinformatics* 24.13, pp. i232–i240. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btn162.
- YANG, Zitong *et al.* (2020). “Rethinking bias-variance trade-off for generalization of neural networks”. In: *International Conference on Machine Learning*. PMLR, pp. 10767–10777.
- YU, Douglas W., Yinqiu Ji, Brent C. EMERSON, *et al.* (2012). “Biodiversity soup: metabarcoding of arthropods for rapid biodiversity assessment and biomonitoring”. en. In: *Methods in Ecology and Evolution* 3.4, pp. 613–623. ISSN: 2041-210X. DOI: 10.1111/j.2041-210X.2012.00198.x.
- YU, Qiuyan, Wenjie Ji, Lara PRIHODKO, *et al.* (2021). “Study becomes insight: Ecological learning from machine learning”. en. In: *Methods in Ecology and Evolution* 12.11, pp. 2117–2128. ISSN: 2041-210X. DOI: 10.1111/2041-210X.13686.
- ZEČEVIĆ, Matej *et al.* (Oct. 2021). “Relating Graph Neural Networks to Structural Causal Models”. In: *arXiv:2109.04173 [cs, stat]*. arXiv: 2109.04173.
- ZERROUKI, Yacine *et al.* (2020). “Desertification detection using an improved variational autoencoder-based approach through ETM-landsat satellite data”. In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 14, pp. 202–213.
- ZHANG, Fei, Minghui WANG, Jianing XI, *et al.* (Feb. 2018). “A novel heterogeneous network-based method for drug response prediction in cancer cell lines”. In: *Scientific Reports* 8.1, p. 3355. ISSN: 2045-2322.
- ZHANG, Jingfa, Yang XU, *et al.* (2021). “Anomaly detection in hyperspectral image using 3D-convolutional variational autoencoder”. In: *2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS*. IEEE, pp. 2512–2515.
- ZHANG, Shuai, Meng WANG, Sijia LIU, *et al.* (2021). “Why Lottery Ticket Wins? A Theoretical Perspective of Sample Complexity on Pruned Neural Networks”. In: *arXiv preprint arXiv:2110.05667*.
- ZHAO, Gangming, Zhaoxiang ZHANG, *et al.* (2017). “Training Better CNNs Requires to Rethink ReLU”. In: *CoRR* abs/1709.06247.
- ZHAO, Qingyuan and Trevor HASTIE (2021). “Causal interpretations of black-box models”. In: *Journal of Business & Economic Statistics* 39.1, pp. 272–281.
- ZHOU, Daquan *et al.* (2021). “Deepvit: Towards deeper vision transformer”. In: *arXiv preprint arXiv:2103.11886*.
- ZHUANG, Fuzhen *et al.* (Jan. 2021). “A Comprehensive Survey on Transfer Learning”. In: *Proceedings of the IEEE* 109.1. Conference Name: Proceedings of the IEEE, pp. 43–76. ISSN: 1558-2256. DOI: 10.1109/JPROC.2020.3004555.
- ZOU, Hui and Trevor HASTIE (2005). “Regularization and variable selection via the elastic net”. In: *Journal of the royal statistical society: series B (statistical methodology)* 67.2. tex.publisher: Wiley Online Library, pp. 301–320.
- ZOU, James and Londa SCHIEBINGER (July 2018). “AI can be sexist and racist — it’s time to make it fair”. en. In: *Nature* 559.7714. Number: 7714 Publisher: Nature Publishing Group, pp. 324–326. DOI: 10.1038/d41586-018-05707-8.
- ZURELL, Damaris, Laura J. POLLOCK, and Wilfried THUILLER (2018). “Do joint species distribution models reliably detect interspecific interactions from co-occurrence data in homogenous environments?” en. In: *Ecography* 41.11, pp. 1812–1819. ISSN: 1600-0587. DOI: 10.1111/ecog.03315.
- ZUUR, Alain F *et al.* (2009). *Mixed effects models and extensions in ecology with R*. Vol. 574. Springer.