Crucial but often neglected: The important role of spatial autocorrelation in hyperparameter tuning and predictive performance of machine-learning algorithms for spatial data.

Patrick Schratz[a], Jannes Muenchow[a], Eugenia Iturritxa[b], Jakob Richter[c], Alexander Brenning[a]

[a] *Department of Geography, GIScience group, Grietgasse 6, 07743, Jena, Germany*
[b] *NEIKER, Granja Modelo –Arkaute, Apdo. 46, 01080 Vitoria-Gasteiz, Arab, Spain*
[c] *Department of Statistics, TU Dortmund University, Germany*

Abstract

While the application of machine-learning algorithms has been highly simplified in the last years due to their well-documented integration in commonly used statistical programming languages such as R, there are several practical challenges in the field of ecological modeling related to unbiased performance estimation. One is the influence of spatial autocorrelation in both hyperparameter tuning and performance estimation. Grouped cross-validation strategies have been proposed in recent years in environmental as well as medical contexts to reduce bias in predictive performance (Brenning & Lausen, 2008; Meyer et al., 2018; Peña & Brenning, 2015; Pohjankukka et al., 2017; Roberts et al., 2017). In this study we showed the effects of spatial autocorrelation on hyperparameter tuning and performance estimation by comparing several widely used machine-learning algorithms such as Boosted Regression Trees (BRT), k-Nearest Neighbor (KNN), Random Forest (RF) and Support Vector Machine (SVM) with traditional parametric algorithms such as logistic regression (GLM) and semi-parametric ones like Generalized Additive Models (GAM) in terms of predictive performance. Spatial and non-spatial cross-validation methods were used to evaluate model

*Corresponding author
Email address: patrick.schratz@uni-jena.de (Patrick Schratz)

performances aiming to obtain bias-reduced performance estimates. A detailed analysis on the sensitivity of hyperparameter tuning when using different resampling methods (spatial/non-spatial) was performed. As a case study the spatial distribution of forest disease (*Diplodia sapinea*) in the Basque Country in Spain was investigated using common environmental variables such as temperature, precipitation, soil and lithology as predictors. Random Forest (mean Brier score estimate of 0.166) outperformed all other methods with regard to predictive accuracy. While algorithms tuned using spatial and non-spatial resampling performed equally well in this work, spatial hyperparameter tuning maintains consistency with spatial estimation of classifier performance and should be favored over non-spatial hyperparameter optimization. Though the sensitivity to hyperparameter tuning differed between the ML algorithms, there were in most cases no significant differences between spatial and non-spatial partitioning for hyperparameter tuning. High performance differences (up to 47%) between the bias-reduced (spatial cross-validation) and overoptimistic (non-spatial cross-validation) cross-validation settings showed the high need to account for the influence of spatial autocorrelation. Overoptimistic performance estimates may lead to false actions in ecological decision making based on biased model predictions.

Keywords: spatial modeling, machine-learning, spatial autocorrelation, hyperparameter tuning, spatial cross-validation

---

1. Introduction

Spatial predictions are of great importance in a wide variety of fields including hydrology (Naghibi et al., 2016), epidemiology (Adler et al., 2017), geomorphology (Brenning et al., 2015), remote sensing (Stelmaszczuk-Górska et al., 2017), climatology (Voyant et al., 2017), the soil sciences (Hengl et al., 2017) and of course ecology (Baasch et al., 2010; Muenchow et al., 2013; Murase et al., 2009; Vorpahl et al., 2012). Ecological applications range from species distribu-

tion models (Halvorsen et al., 2016; Quillfeldt et al., 2017; Wieland et al., 2017) to landslide prediction (Vorpahl et al., 2012) and resource selection (Baasch et al., 2010).

Fungal species such as *Diplodia sapinea* inflict severe damage on Monterrey pine trees (*Pinus radiata*) which are then subjected to environmental stress (Wingfield et al., 2008). Infected forest stands cause economic as well as ecological damages worldwide (Ganley et al., 2009). In Spain, where timber production is regionally an important economic factor, about 25% of the timber production stems from Monterrey pine (*Pinus radiata*) plantations in northern Spain, and here mostly from the Basque Country (Iturritxa et al., 2014). Consequently, the early detection and subsequent containment of fungal diseases is of great importance. Statistical and machine-learning models can help in this process by mapping the current infection state and exploring relations between the pathogens and environmental variables. These findings can then be used for spatially predicting the risk of future outbreaks.

1.1 The special role of spatial autocorrelation in predictive modeling

All of the previously mentioned fields have at least one thing in common: The observations inherit spatial information. One of the main challenges that comes with this information is to account for the influence of spatial autocorrelation in the data (Legendre, 1993). Cross-validation and bootstrapping are two widely used performance estimation techniques (Efron, 1983; Gordon et al., 1984; Kohavi & others, 1995). However, in the presence of spatial autocorrelation, estimates obtained using regular (non-spatial) random resampling may be biased and overoptimistic. This has led to the adoption of spatial resampling in cross-validation and bootstrapping for bias reduction. The mentioned bias inherits from the fact that training and test observations are located close to each other (in a geographical space) if a random sampling is used in Cross-Validation (CV) (Legendre, 1993). Random sampling in CV leads to the selection of test observations that are spatially close to training observations. However, according

to the first law of geography, close observations are frequently more similar to each other than observations further apart which violates the fundamental assumption of independence in cross-validation. Hence, algorithms fitted on the training data often achieve very good performance results, simply because the characteristics of the evaluation set are very similar to the training data.

One approach to solve this, which has been applied in various studies in the last decade, builds upon the idea to spatially disjoin training and test set in CV. The naming of this concept varies with the scientific field in which it is applied: Burman et al. (1994); Roberts et al. (2017); Shao (1993) label it "Block cross-validation", Brenning (2005) as "spatial cross-validation", Pohjankukka et al. (2017) "spatial k-fold cross-validation" and Meyer et al. (2018) "Leave-location-out cross-validation". Here, we use the term "spatial cross-validation" because it is the most generic wording to label this concept and hope that this naming convention will prevail. Although the importance of bias-reduced spatial resampling methods for performance estimation has been emphasized repeatedly in recent years (Geißet al., 2017; Meyer et al., 2018; Wenger & Olden, 2012), many studies have been published in recent years that did not account for this problem (Bui et al., 2015; Pourghasemi & Rahmati, 2018; Smoliński & Radtke, 2016; Wollan et al., 2008; Youssef et al., 2015).

1.2 Parametric vs. non-parametric algorithms

Supervised learning techniques can be broadly divided into parametric and non-parametric models. Parametric models can be written as mathematical equations involving model coefficients. This enables ecologists to interpret relationships between the response and its predictors. Choosing the best performing algorithm for a specific dataset is an essential step in ecological modeling to maximize predictive accuracy. In this context, model interpretability should certainly be an important criterion in the selection process when the aim is to make inference on the modeled relationship (Johnson & Omland, 2004). While the most commonly used statistical models such as generalized linear mixed

models (GLMMs) are parametric, especially machine-learning techniques offer a non-parametric approach to spatial modeling in ecology (De'ath, 2007). Although their ability to make inference is limited compared to parametric ones, these gained in popularity due to their ability to handle high-dimensional and highly correlated data and their less important model assumptions.

1.3 The importance of hyperparameter optimization

To reach good performance results with non-parametric models, their respective hyperparameters must be optimized. Default hyperparameter settings can not guarantee an optimal performance of machine-learning techniques and additional attention should be directed to this critical step. When performance estimation techniques such as cross-validation are used in this step, the adequacy of non-spatial techniques for spatial datasets can be questioned. Although spatial resampling methods have been used in studies that deal with spatial data for quite some time now (Geißet al., 2017; Iturritxa et al., 2014; Meyer et al., 2018), there is no analysis of the effect and meaningfulness of using spatial resampling techniques for hyperparameter tuning. This study aims to check if optimizing hyperparameters using a non-spatial sampling may potentially lead to non-optimal performance estimates.

1.4 Main objectives

Overall, the intention of this work is to emphasize the need for spatial CV and corresponding hyperparameter tuning in spatial modeling to receive biased-reduced performance estimates. The following objectives (and hypotheses) are addressed:

- Comparing predictive performance of spatial and non-spatial partitioning methods (We assume that non-spatial partitioning methods will yield over-optimistic results in the presence of spatial autocorrelation.)
- Exploring the effects of (spatial) hyperparameter tuning for commonly used algorithms in the field of ecological modeling (We propose that op-

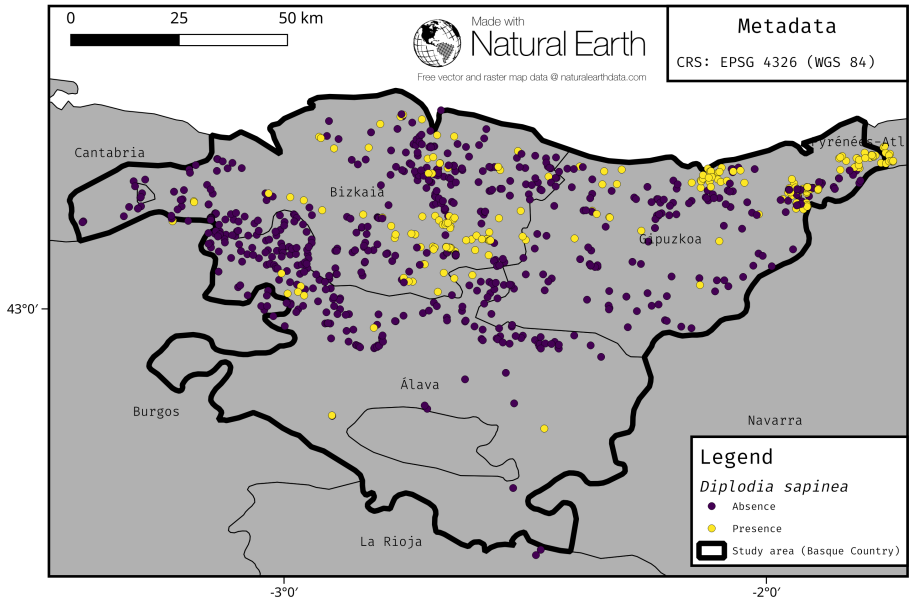

Figure 1: Spatial distribution of tree observations within the Basque Country, northern Spain, showing infection state by *Diplodia sapinea*.

timal hyperparameter tuning has an substantial effect on model performance.)

- Comparing the predictive performance of parametric (GLM, GAM) and non-parametric algorithms (BRT, RF, SVM, KNN) (We assume that the predictive performance of non-parametric algorithms is substantially higher)

2. Data and study area

2.1 Summary of the prediction task

This study uses parts of the dataset from Iturritxa et al. (2014). While Iturritxa et al. (2014) focused on the influence of environmental predictors on pathogen probability, the aim of this study is to compare different algorithms with the focus of exploring the influence of spatial autocorrelation on predictive accuracy and hyperparameter tuning. In addition to that, in the present study, we also

used new predictors (probability of hail damage at trees, soil type, lithology type, pH) to enhance the prediction of the response variable *Diplodia sapinea*.

This dataset was chosen because it incorporates attributes of common geospatial modeling tasks: An uneven distribution of the binary response variable (25/75), presence of spatial autocorrelation and predictor variables derived from various sources (previous modeling results, remote sensing data, surveyed information). It is representative for many other ecological datasets in terms of sample size (926), number (11) and predictor type (numeric as well as nominal).

2.2 Variables

The following (environmental) variables were used as predictors: Mean temperature (March - September), mean total precipitation (July - September), Potential Incoming Solar Radiation (PISR), elevation, slope (degrees), potential hail damage at trees, tree age, pH value of soil, soil type, lithology type, and the year when the tree was surveyed. Temperature, precipitation and PISR are long-term averages (1951 - 1999) of meteorological stations across the Iberian Peninsula (Ninyerola et al., 2005). Tree infection caused by the fungal pathogen *Diplodia sapinea* represents the response variable. The ratio of infected and non-infected trees in the sample is roughly 1:3 (223, 703).

Iturritxa et al. (2014) showed that variable "hail", a variable representing the presence or absence of hail damage observed on trees, explained best pathogen infections of trees in the Basque Country. Because almost every infected tree by *Diplodia sapinea* showed hail damage, it was assumed that the pathogen uses the open wounds caused by the hail damage as an entry point. To make the tree-based hail damage variable spatially available for the whole Basque country, we spatially predicted hail damage potential (in probabilities from 0 - 1) as a function of climatic variables using a Generalized Additive Model (GAM) (Schratz, 2016). The reasoning for extending the original dataset was that the pathogen might favor specific soil or lithology types, pH environments or younger/older trees. In the following we shortly describe the source and

modifications of the new variables. For the remaining ones, please see Iturritxa et al. (2014).

Predictor *soil* was predicted by Hengl et al. (2017) using ca. 150.000 soil profiles at a spatial resolution of 250 m. Predictor *age* was imputed and trimmed to a value of 40 to reduce the influence of outliers. Predictor *pH* was mapped by the European Commission (2010) using a regression-kriging approach based on 12.333 soil pH measurements from 11 different sources. GeoEuskadi provided the lithology types (GeoEuskadi, 1999). The rock class were aggregated by the respective top level class for magmatic types and sub-classes for sedimentary rocks (Grotzinger & Jordan, 2016) (Table A.3).

We removed three observations due to missing information in some variables leaving a total of 926 observations (Table A.2).

2.3 Study area

The Basque country in northern Spain represents the study area (Figure 1). It has a spatial extent of 7355 km^2. Precipitation decreases towards the south while the duration of summer drought increases. Between 1961 and 1990, mean annual precipitation ranged from 600 to 2000 mm with annual mean temperatures between 8 and 16°C (Ganuza & Almendros, 2003). The wooded area covers approximately 54% of the territory (3969.62 km^2), which is one of the highest ratios in the EU. Radiata pine is the most abundant species occupying 33.27% of the total area (Múgica et al., 2016).

3. Methods

In this study we provide an exemplary analysis combining both tuning of hyperparameters (see subsection 1.3) using nested CV (see subsubsection 3.2.1) and the use of spatial CV to assess bias-reduced model performances (see subsection 1.1). We compared predictive performances using four setups: Non-spatial CV for performance estimation combined with non-spatial hyperparameter tuning (*non-spatial/non-spatial*), spatial CV estimation with spatial hyperparam-

eter tuning (*spatial/spatial*), spatial CV estimation with non-spatial hyperparameter tuning (*spatial/non-spatial*), and spatial CV estimation without hyperparameter tuning (*spatial/no tuning*). We used the open-source statistical programming language R (R Core Team, 2017). The algorithm implementations of the following packages have been used: *gbm* (Ridgeway, 2017) (Boosted Regression Trees (BRT), Elith et al. (2008)), *mgcv* (Wood, 2017) (GAM), *kernlab* (Karatzoglou et al., 2004) (Support Vector Machine (SVM), Vapnik (1998)), *kknn* (Schliep & Hechenbichler, 2016) (Weighted k-nearest neighbor (KNN), Dudani (1976)), and *ranger* (Wright & Ziegler, 2017) (Random Forest (RF), Breiman (2001)). The spatial partitioning functions of the *sperrorest* package have been integrated into the *mlr* package as part of this work. *mlr* provides a standardized interface for a wide variety of statistical and machine-learning models in R simplifying essential modeling tasks such as hyperparameter tuning, model performance evaluation and parallelization (Bischl et al., 2016). We provide the complete code and required data as a Mendeley dataset to make this work fully reproducible (Schratz & Iturritxa, 2018).

3.1 Tuning

Determining the optimal (hyperparameter) settings for each model is crucial for the bias-reduced assessment of a model's predictive power. Hyperparameters of machine-learning algorithms need to be tuned to achieve optimal performances (Bergstra & Bengio, 2012; Duarte & Wainer, 2017; Hutter et al., 2011). Often enough, parametric models do not require tuning to achieve optimal performances. However, some (semi-)parametric algorithms (e.g. GAM, penalized regression methods) can be optimized to possibly increase their performance.

3.1.1 Parameter vs. hyperparameter

For parametric models the term "parameter" is often used to refer to the regression coefficients of each predictor in the fitted model. However, for machine-learning algorithms, the terms "parameter" and "hyperparameter" both refer

to "hyperparameter" as there are no regression coefficients for these models. In addition, the term "parameter" is often used in programming to refer to an argument of a function. Hyperparameters determine how exactly an algorithm work and they have an influence on the final outcome.

Usually it is not known in advance how to set them to achieve the best outcome for a specific problem. Therefore hyperparameter optimization is necessary to determine the best setting. Hyperparameters are determined by finding the optimal setting for an algorithm using optimization procedures such as *random search* or *Bayesian optimization* while parameters of parametric models are estimated when fitting them to the data (Kuhn & Johnson, 2013).

3.1.2 Tuning method

For hyperparameter tuning, we used Sequential Model-Based Optimization (SMBO) as implemented in the *mlrMBO* package (Bischl et al., 2017). At first, n hyperparameter settings are randomly chosen from a user-defined search space. Next,

Table 1: Hyperparameter ranges and types for each model. Notations of hyperparameters from the respective R packages were used. Note that parameter `sp` of the GAM is a vector with eight entries (one entry for each numeric predictor). `p` is the number of predictors.

Algorithm (package)	Hyperparameter	Type	Start	End	Default
BRT (gbm)	`n.tree`	integer	100	15000	100
	`shrinkage`	numeric	0	1.0	0.001
	`interaction.depth`	integer	1	20	1
KNN (kknn)	`k`	integer	1	250	7
	`distance`	integer	1	300	2
GAM (mgcv)	`sp`	numeric	0	10^6	-
RF (ranger)	`mtry`	integer	1	11	$\sqrt{p}$
	`min.node.size`	integer	1	10	1
	`sample.fraction`	numeric	0.2	0.9	1
SVM (kernlab)	`C`	numeric	2^{-15}	2^{15}	1
	σ	numeric	2^{-15}	2^{15}	1

they are evaluated on the chosen resampling strategy. Based on the previous evaluations a regression model is fitted. The regression model estimates the performance of the machine learning method for unknown hyperparameter settings. Using these estimates, a new promising hyperparameter setting is proposed to be evaluated next. This is continued until a termination criterion is reached (Hutter et al., 2011; Jones et al., 1998). In this work we used an initial design of 30 randomly composed hyperparameter settings and a termination criterion of 70 iterations, resulting in a total budget of 100 evaluated settings per fold. This tuning approach substantially reduces the tuning budget that is needed to find a setting that is close to the global minimum compared to methods that do not use information from previous runs such as *random search* or *grid search* (Bergstra & Bengio, 2012).

3.1.3 Hyperparameter search spaces

The boundaries of the hyperparameter search spaces were based on the suggestions of the *mlrHyperopt* package. In cases when the optimal setting of the folds of a model was close to the specified minimum or maximum of the tuning space, we extended the limits. We furthermore checked on the first five inner folds of each outer fold that the number of tuning iterations set in the SMBO tuning was sufficiently large (Figure 4). This requirement was met if no new local minimum was found in the last 10 % of the iterations of the selected fold.

In addition, all models were fitted using their respective default hyperparameter settings, i.e. no tuning was performed. For SVM we used $\sigma = 1$ and $C = 1$ to suppress the automatic tuning that is usually applied by the *kernlab* package. These are the default settings set by the package before the automatic tuning is applied. The GAM implementation used in this work performs by default an internal non-spatial Generalized Cross-Validation (GCV) to find the best smoothing parameter λ for each predictor (Wood, 2017). To make the optimization of models comparable, we tuned λ for each covariate using the tuning method that was also applied to the machine-learning algorithms. For

the "no tuning" setups, we set $\lambda = 0$ for all predictors. The basis dimension for all GAM setups was set to $k = 15$ for all variables. The search space for λ $(0 - 10^6)$ was determined by examining the results of a prior tuning using the internal tuning of the GAM.

3.1.4 Spatial vs. non-spatial hyperparameter tuning

Hyperparameters estimated from a non-spatial tuning lead to fitted models which are more adapted to the training data than models with hyperparameters estimated from a spatial tuning. In a non-spatial tuning setting, hyperparameters that lead to a close fit of the algorithm to the data will be favored in the tuning process due to the presence of spatial autocorrelation.

Models fitted with hyperparameters from a non-spatial tuning can potentially benefit from the remaining spatial autocorrelation in the train/test split (even if a spatial resampling was used) during performance estimation and achieve a better performance than models tuned using a spatial resampling. However, depending on the dataset structure and closeness of the model fit on the data, the reverse effect might occur and models fitted with a spatial tuning setting might yield better results. In the end it depends on whether the training/test difference is more similar to a spatial tuning setting (i.e. more heterogeneous train/test splits) or to a non-spatial tuning setting (i.e. more homogeneous train/test sets).

3.1.5 Practical implementation

Most packages offering CV solutions in R offer only random partitioning methods, assuming independence of the observations. Package *mlr*, which was used as the modeling framework in this work, was missing spatial partitioning functions but provides a unified framework for modeling and simplifies hyperparameter tuning. With this study we implemented the spatial partitioning methods of package *sperrorest* into *mlr*.

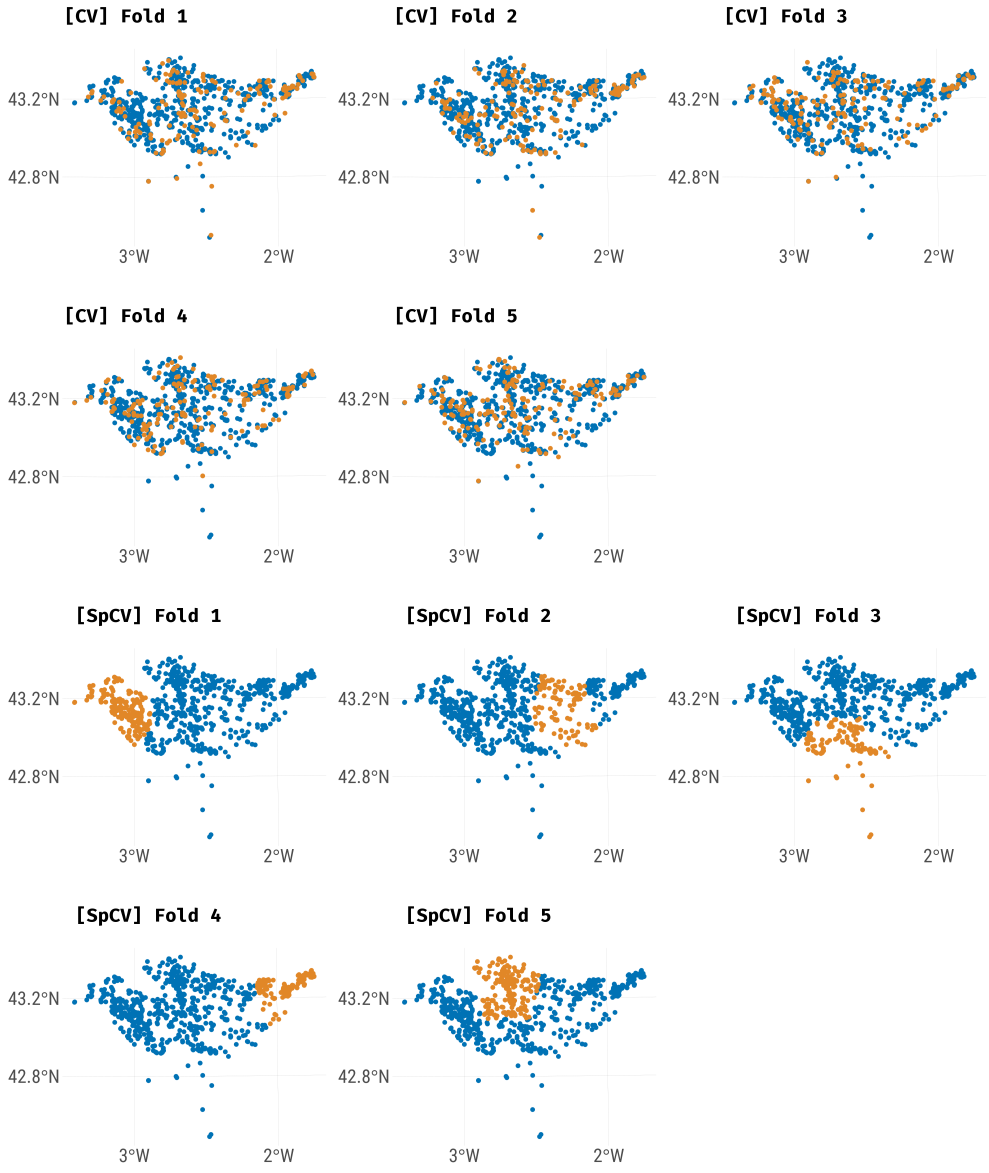

Figure 2: Comparison of spatial and non-spatial partitioning of the first five folds in spatial and non-spatial cross-validation performance estimation. Blue dots represent the training samples and orange dots the testing sample. "SpCV" stands for spatial cross-validation (spatial sampling of observations) and "CV" for cross-validation (random sampling of observations).

3.2 Estimation of predictive performance

3.2.1 Nested cross-validation

Cross-validation is a resampling-based technique for the estimation of a model's predictive performance (James et al., 2013). The basic idea behind CV is to split an existing dataset into training and test sets using a user-defined number of partitions (Figure 3). First, the dataset is divided into k partitions or folds. The training set consists of $k-1$ partitions and the test set of the remaining partition. The model is trained on the training set and evaluated on the test partition. A repetition consists of k iterations for which every time a model is trained on the training set and evaluated on the test set. Each partition serves as a test set once.

3.2.2 Influence of spatial autocorrelation in cross-validation

In ecology, observations are often spatially dependent (Dormann et al., 2007; Legendre & Fortin, 1989). Subsequently, they are affected by underlying spatial autocorrelation by a varying magnitude (Legendre, 1993; Cliff & Ord, 1970; Telford & Birks, 2005). Model performance estimates are expected to be overoptimistic due to the similarity of training and test data in a non-spatial partitioning setup when using any kind of cross-validation for tuning or validation (Burman et al., 1994; Cliff & Ord, 1970; Racine, 2000). Therefore, cross-validation approaches that adapt to this problem should be used in any kind of performance evaluation when spatial data is involved (Meyer et al., 2018; Telford & Birks, 2009). In this work we use the spatial cross-validation approach after Brenning (2012) which uses k-means clustering to reduce the influence of spatial autocorrelation. In contrast to non-spatial CV, spatial CV reduces the influence of spatial autocorrelation by partitioning the data into spatially disjoint subsets (Figure 3). These are determined by k-means clustering (Brenning, 2012).

Five-fold partitioning repeated 100 times was chosen for performance estimation (Figure 3). For the hyperparameter tuning, again five folds were used to split the training set of each fold. Hyperparameter tuning only applied to the

machine-learning algorithms. A sequential model-based optimization approach was used to tune the hyperparameters (see subsection 3.1). Model performances of every hyperparameter setting were computed at the tuning level and averaged across folds. The hyperparameter setting with the lowest mean Brier score across all tuning folds was used to train a model on the training set of the respective performance estimation level. This model was then evaluated on the test set of the respective fold (performance estimation level). The procedure was repeated 500 times to reduce random variability introduced by partitioning.

3.2.3 Cross-Validation setups

To underline the crucial need for spatial CV when assessing a model's performance, and to identify overoptimistic outcomes when neglecting to do so, we used the following CV setups: Nested non-spatial CV which uses random partitioning and non-spatial hyperparameter tuning (*non-spatial/non-spatial*), nested spatial CV which uses k-means clustering for partitioning (Brenning, 2005) and results in a spatial grouping of the observations and performs non-spatial hyperparameter tuning (*spatial/non-spatial*), nested spatial CV including spatial hyperparameter tuning (*spatial/spatial*) and spatial CV without hyperparameter tuning (*spatial/no tuning*). Setup (*non-spatial/non-spatial*) was only used to show the overoptimistic results when using non-spatial CV with spatial data and setups *spatial/non-spatial*, *spatial/spatial* to reveal the differences between spatial and non-spatial hyperparameter tuning. Setup (*spatial/spatial*) should be used when conducting spatial modeling with machine learning algorithms that require hyperparameter tuning.

3.2.4 Performance measure

The Brier score was selected as a scoring rule to to compare the predictive performances of all algorithms (Brier, 1950). In contrast to other commonly used measures for binary classification (e.g. the Area Under the Receiver Operating Characteristics Curve (AUROC)), the Brier score classifies as a proper scoring

rule (Byrne, 2016; Gneiting & Raftery, 2007). It is defined as the mean quadratic loss between the predicted and observed probabilities. It ranges from 0 to 1 with low values indicating a good prediction (Brier, 1950).

3.2.5 A note on spatial autocorrelation structures in parametric models

In this work we assume that, on average, the predictive accuracy of parametric models with and without spatial autocorrelation structures is the same. However, there is little research on this specific topic (Dormann, 2007; Mets et al., 2017) and a detailed analysis goes beyond the scope of this work. In our view, a possible analysis would need to estimate the spatial autocorrelation structure of a model for every fold of a cross-validation using a data-driven approach (i.e. automatically estimate the spatial autocorrelation structure from each training set in the respective CV fold) and compare the results to the same model fitted without a spatial autocorrelation structure. Since we only focused on predictive accuracy in this work, we did not use spatial autocorrelation structures during model fitting for Generalized Linear Model (GLM) and GAM to reduce runtime.

4. Results

4.1 Tuning

4.1.1 Optimization paths

To proof the effectiveness of the tuning, the optimization paths of the first five folds of RF for settings spatial/spatial and *spatial/non-spatial* are visualized (Figure 4). Using 100 SMBO iterations, all shown folds show decreases in Brier score along the optimization path (Figure 4). Apart from fold 5 of setting *spatial/non-spatial*, all folds show a saturation of at least 15 or more iterations in which no new local minimum was found.

4.1.2 Best hyperparameter settings

There were notable differences in the distribution of the estimated optimal hyperparameters between the spatial (*spatial/spatial*) and non-spatial (*spatial/non-*

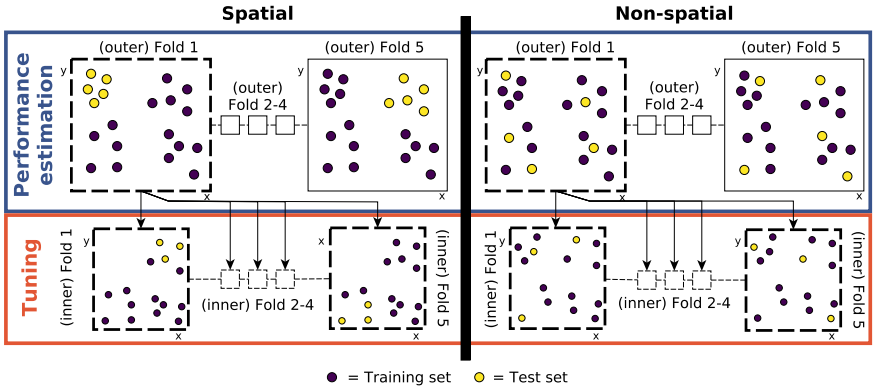

Figure 3: Theoretical concept of spatial and non-spatial nested cross-validation using five folds for hyperparameter tuning and performance estimation. Yellow/purple dots represent the training and test set for performance estimation, respectively. The tuning sample is based on the respective performance estimation fold sample and consists again of training (orange) and test set (blue). Although the tuning folds of only one fold are shown here, the tuning is performed for every fold of the performance estimation level.

spatial, *non-spatial/non-spatial*) tuning settings (Figure 5): In the spatial tuning setting, all models besides BRT show a wide range of optimal hyperparameters across folds. By contrast, the range of optimal settings in the non-spatial tuning case is much smaller and often clusters around a few specific settings (e.g. compare the spatial and non-spatial results of the SVM) (Figure 5).

For the spatial tuning case of RF, the estimated m_{try} values mainly ranged between 1 and 3 and m_{try} of 1 was most often the optimal value. This stand in strong contrast to the non-spatial tuning setting in which m_{try} mainly ranged between 3 and 5 and m_{try} of 3 was most often the optimal choice. Generally, we observed the tendency that spatially tuned hyperparameters are often close to the limits of the search space especially when compared to their non-spatial counterparts. The GAM results are not included in Figure 5 as the estimated hyperparameter (smoothing parameter λ) is a vector of length eight (eight being the number of non-linear variables in the model formula) that cannot be

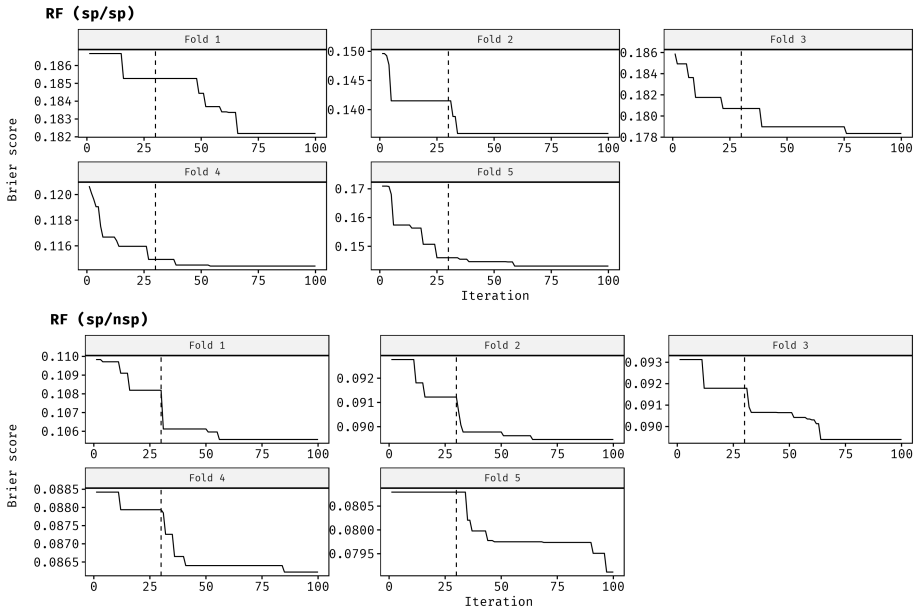

Figure 4: SMBO optimization paths of the first five folds of the *spatial/spatial* and *spatial/non-spatial* CV setting for RF. The dashed line marks the border between the initial design (30 randomly composed hyperparameter settings) and the sequential optimization part in which each setting was proposed using information from the prior evaluated settings. Optimization paths of the remaining models can be found in the appendix.

visualized in a 2D space.

4.2 Predictive performance

4.2.1 Which models showed the best performance?

For the spatial settings (*spatial/spatial* and *spatial/no tuning*), RF showed the best predictive performance followed by BRT, KNN and GLM (Figure 6). The absolute difference between the best (RF) and worst (GAM) performing model in our setup is 0.039 (mean Brier score (*spatial/spatial*)). The GAM showed a high variance for all spatial settings compared to all other algorithms.

4.2.2 Effect of hyperparameter tuning on predictive performance

The tuning of hyperparameters resulted in a clear increase of predictive performance for BRT (0.183 (*spatial/spatial*) vs. 0.201 (*spatial/no tuning*) mean Brier score), GAM (0.206 (*spatial/spatial*) vs. 0.251 (*spatial/no tuning*) and KNN (0.181 (*spatial/spatial*) vs 0.210 (*spatial/no tuning*) mean Brier score) (Figure 6). No effect of hyperparameter tuning on predictive performance was visible for RF and SVM. The use of spatial partitioning in hyperparameter tuning (setting (*spatial/spatial*) had an substantial positive impact for BRT and a negative one for GAM and KNN (Figure 6).

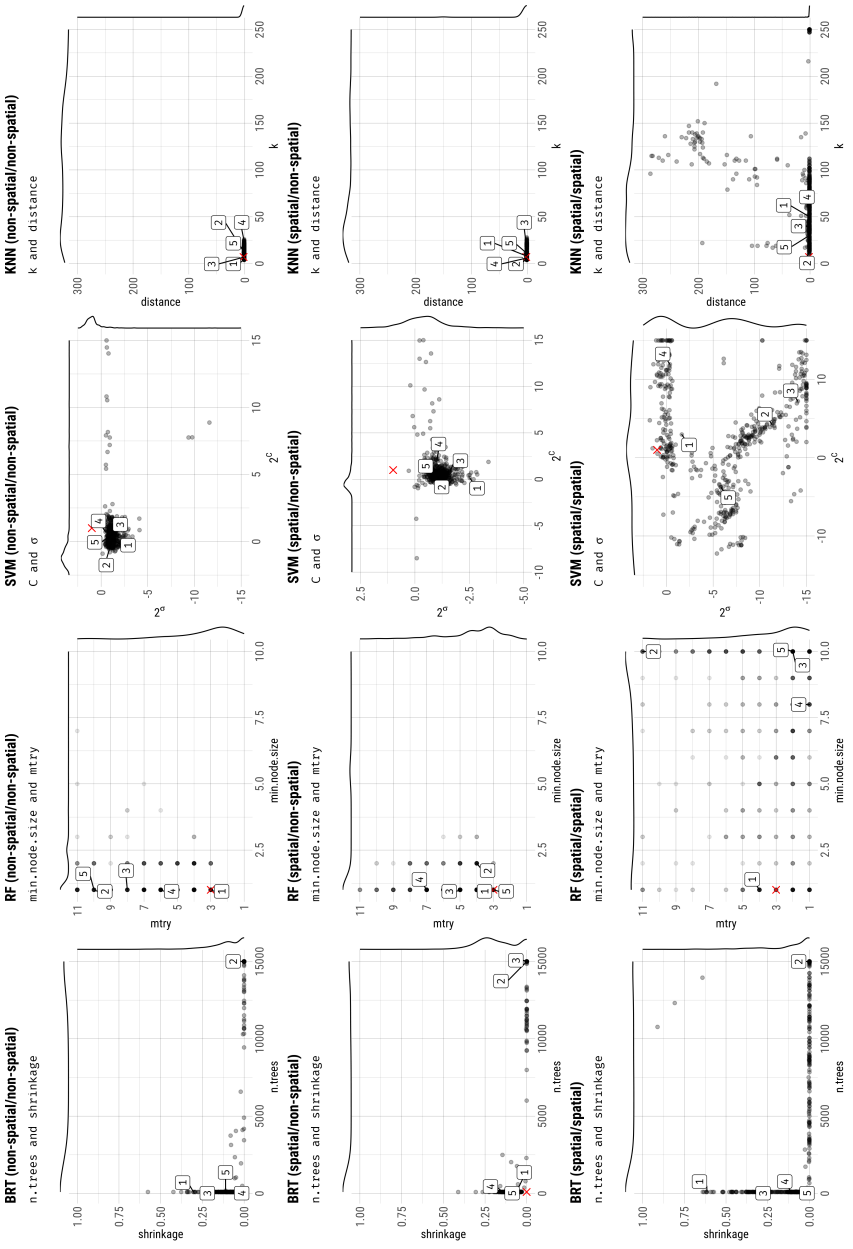

Figure 5: Best hyperparameter settings by fold (500 total) each estimated from 100 (30/70) SMBO tuning iterations per fold using five-fold cross-validation. Split by spatial and non-spatial partitioning setup and model type. Red crosses indicate the default hyperparameters of the respective model. Black dots represent the winning hyperparameter setting of each fold. The labels ranging from one to five show the winning hyperparameter settings of the first five folds. Density lines on both axis show the density distribution of the respective variable.

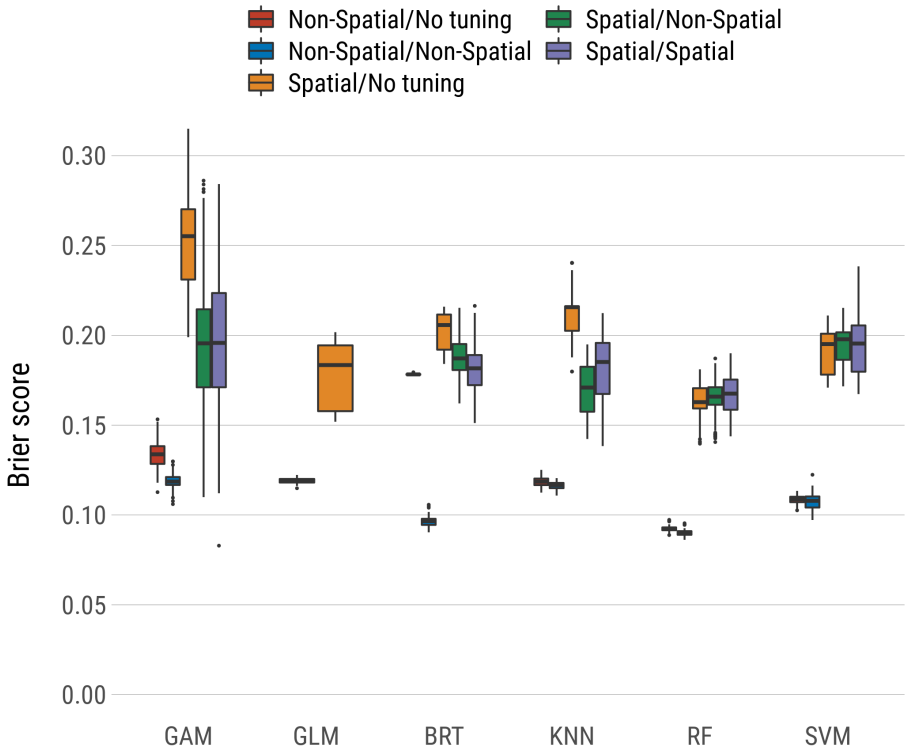

Figure 6: (Nested) CV estimates of model performance at the repetition level using 100 SMBO iterations for hyperparameter tuning. CV setting refers to performance estimation/hyperparameter tuning of the respective (nested) CV, e.g. "Spatial/Non-Spatial" means that spatial partitioning was used for performance estimation and non-spatial partitioning for hyperparameter tuning.

4.2.3 Comparison of spatial vs non-spatial tuning

Predictive performance estimates based on non-spatial partitioning (*non-spatial/non-spatial* or *non-spatial/no tuning*) are around 33 - 47% higher, i.e. overoptimistic, compared to their spatial equivalents (*spatial/spatial*, *spatial/no tuning*). BRT and RF show the highest differences between these two settings (47% and 46%, respectively) while GLM was the least affected (33%).

5. Discussion

5.1 Tuning

5.1.1 Tuning methods

The question on the most efficient approach of hyperparameter tuning has been discussed for decades (Bengio, 2000; Probst et al., 2018a; Yang et al., 2017). The goal is to use as few computational resources as possible to find a nearly optimal hyperparameter setting of an algorithm for a specific dataset. In this respect, methods like *random search* are particularly promising in multidimensional hyperparameter spaces with possibly redundant or insensitive hyperparameters (low effective dimensionality; (Bergstra & Bengio, 2012)). Adaptive search algorithms offer computationally efficient solutions to these difficult global optimization problems in which little prior knowledge on optimal subspaces is available. Approaches like Bayesian Optimization and F-racing are widely used for optimization of black-box models (Birattari et al., 2002; Bischl et al., 2017; Brochu et al., 2010; Malkomes et al., 2016). In this study, we used a sequential model-based optimization (Bayesian optimization) method. Other tuning methods would have yielded almost identical results but at the cost of increased computational efficiency and less robustness in terms of finding the local minimum.

5.1.2 Algorithm sensitivity to tuning

Some models (e.g. RF) are known to be relatively insensitive to hyperparameter tuning (Probst et al., 2018b). However, as the effect of hyperparameter tuning also depends on the dataset, hyperparameters should always be tuned. If no tuning is conducted, it cannot be ensured that the respective model showed its best possible predictive performance on the dataset.

5.1.3 Hyperparameter search spaces

Computational expense, especially when using tuning methods like random search, should focus on plausible parameter settings for each model. It should be ensured by visual inspection that the majority of the obtained optimal hyperparameter settings is not close to the ranges of the tuning space. If the optimal hyperparameter settings are clustered at the borders of the parameter search space, this implies that optimal hyperparameters may actually lie outside the given range. However, extending the tuning space is not always possible nor practical as (1) numerical problems within the algorithm may occur that may prohibit further extension of the tuning space; (2) some algorithms tend to mainly use the limits of the given search space although no significant increase is achieved (e.g. KNN in the *spatial/spatial* setting).

We encountered exactly these problems in the *spatial/spatial* setting for BRT, KNN and SVM. For example, in the *spatial/spatial* setting, we should have further increased the search space for the mentioned models based on the distribution of the optimal hyperparameters of each fold (Figure 5). However, using the extended setting, the algorithms did not converge anymore for some folds and at the same time runtime increased without a significant increase in predictive performance.

All these points show the need for a thorough specification of parameter search spaces. As the optimal hyperparameter ranges also depend on the dataset characteristics, it is not possible to define a universal search space that works best on every dataset. Nevertheless, the chosen hyperparameter limits of this work can serve as a starting point for future analyses in the spatial modeling field. Within the framework of the *mlr* project a database exists which stores hyperparameter settings of various models from users that can serve as a reference point (Richter, 2017).

5.1.4 Comparison of spatial vs non-spatial tuning

No major differences in model performances were found when using spatial versus non-spatial hyperparameter tuning procedures (e.g. 0.019 for BRT (0.182 vs. 0.201 mean Brier score).

The winning algorithm RF is used to discuss the optimal estimated hyperparameters per fold of the spatial and non-spatial tuning setting in more detail. Although the tuning of RF had no substantial effect on predictive performance (Figure 6), the estimated optimal hyperparameters of RF differ for the non-spatial and spatial tuning setting (Figure 5). We split the following discussion into two points: (1) The nature of the algorithm and (2) the implications which method should be chosen. (1) In a non-spatial tuning setting, RF will prioritize spatially autocorrelated predictors because these will yield the best performance results in the internal variable selection process that uses the *Gini impurity measure* (Biau & Scornet, 2016; Gordon et al., 1984). We define "spatially autocorrelated predictors" as variables that show highly similar patterns in its relationship to the response in both training and test set. By selecting these, the algorithm is able to achieve good performances because the trained patterns appear almost identical in the test set. The resulting performances are then overoptimistic as they benefit highly from the non-spatial sampling scheme.

In this pre-selection, `mtry` values around 3 - 5 are favored because they provide a fair chance of having one of the autocorrelated predictors included in the selection. At the same time, `mtry` is low enough to prevent overfitting on the training data because the autocorrelated predictors are not always available to the algorithm.

In the spatial tuning setting, mainly `mtry = 1` is chosen. This specific value essentially removes the internal variable selection process by `mtry` as RF is forced to use the predictor that was randomly assigned. Subsequently, on average, each predictor will be chosen equally often and the higher weighting of spatially autocorrelated predictors in the final model (by choosing them more often in

the trees) does not apply. This leads to a more general model that apparently performs better on heterogeneous datasets (e.g. if training and test data are less affected by spatial autocorrelation) as it is the case in a spatial sampling.

(2) Even though the estimated hyperparameters from a spatial and non-spatial sampling differ, they roughly achieve the same performance when being evaluated at the performance estimation level of the CV. This outcome is not generalizable and highly depends on the dataset of this study. It needs to be verified by using other spatial datasets. Performance differences might be more substantial when using either `mtry = 1` or `mtry = 3` on datasets with different characteristics. This applies also to all other algorithms used in this study. In addition, if a model is going to be evaluated on a specific sampling scheme (here spatial sampling), we see no valid argument why its hyperparameters should be trained on a different sampling scheme if the predictive performances do not differ significantly.

5.2 Predictive Performance

5.2.1 Non-spatial vs. spatial CV

Our findings agree with previous studies in that non-spatial performance estimates appear to be substantially "better" than spatial performance estimates (Meyer et al., 2018; Micheletti et al., 2013; Roberts et al., 2017). However, this difference can be attributed to an overoptimistic bias in non-spatial performance estimates in the presence of spatial autocorrelation (Goetz et al., 2015; Meyer et al., 2018; Rußß& Brenning, 2010; Steger et al., 2016). Spatial cross-validation is therefore required for performance estimation in spatial predictive modeling, and similar grouped cross-validation strategies have been proposed elsewhere in environmental as well as medical contexts to reduce bias in predictive performance (Brenning & Lausen, 2008; Meyer et al., 2018; Peña & Brenning, 2015; Pohjankukka et al., 2017; Roberts et al., 2017).

5.2.2 The effect of hyperparameter tuning on predictive accuracy

Although hyperparameter tuning certainly increases the predictive performance for some models (e.g. BRT, GAM and KNN) in our case, the magnitude always depends on the meaningful/arbitrary defaults of the respective algorithm and the characteristics of the dataset. Naturally, the tuning effect is higher for models without meaningful defaults (such as BRT and KNN) than for models with meaningful defaults such as RF. To underline this statement, there was basically no tuning effect for SVM in this study (Figure 6) although the SVM usually shows substantial increases when being tuned (Rojas-Dominguez et al., 2018).

5.2.3 Predictive performance across algorithms

Other studies also found that RF showed the best predictive performance (referring to setting *spatial/spatial*) (Bahn & McGill, 2012; Jarnevich et al., 2017; Smoliński & Radtke, 2016; Vorpahl et al., 2012). The fact that the GLM is showing a better performance than the GAM shows the heterogeneous train/test split introduced by spatial partitioning: The GAM is not able to generalize enough (i.e. it overfits on the training set) if the test set is substantially different to the training set. The high variance of the GAM performances in the spatial setting verify this: If the training set is somewhat similar to the test set, the GAM is able to achieve Brier score results around 0.19. In cases where training and test set are more heterogeneous, the predictive performance shows Brier score estimates up to 0.30. Overall, the linear approach of the GLM is able to generalize better in this study and subsequently results in a better performance.

It maybe surprising at first that the GLM is showing a showing a performance which is similar to that of BRT, KNN and SVM. This is most likely due to the ability of the algorithm to generalize. Highly flexible algorithms have a tendency to overfit when the test set differs substantially from the training set. For instance, a test set located close to the sea might be hard to predict for models trained on data that was almost exclusively located in mountainous regions.

In such a setting, a low degree of flexibility will result in better predictions. This example also shows the importance of traditional parametric approaches in ecological modeling: Often enough ecological datasets show a high degree of diversity and machine-learning models might suffer from overfitting. In this case, the interpretability, speed and generalization attributes of a GLM make this algorithm a valid choice, especially if the differences in predictive accuracy compared to black-box models is small.

5.2.4 The influence of the performance measure

The choice of the scoring rule for the evaluation of binary classifications is an important decision (Gneiting & Raftery, 2007). Measures that are not classified as "proper" can lead to undetected deviations during scoring that can end up in biased results (Byrne, 2016). One of the most used performance measures in the field of binary classification, the AUROC, is affected by this. In a previous version of this work we used AUROC to rank the algorithms. In this, the GAM showed a similar performance as RF. So by only changing the measure, the GAM went from the best (AUROC) to the worst (Brier score) algorithm in this work. This example highlights the importance of selecting a proper scoring rule for model comparison studies. The huge effect of the measure on the GAM should be evaluated in a separate study.

5.2.5 A note on spatial autocorrelation structures in parametric models

In this work we assume that, on average, the predictive accuracy of parametric models with and without spatial autocorrelation structures is the same. However, there is little research on this specific topic (Dormann, 2007; Mets et al., 2017) and a detailed analysis goes beyond the scope of this work. In our view, a possible analysis would need to estimate the spatial autocorrelation structure of a model for every fold of a cross-validation using a data-driven approach (i.e. automatically estimate the spatial autocorrelation structure from each training set in the respective CV fold) and compare the results to the same model fitted

without a spatial autocorrelation structure. Since we only focused on predictive accuracy in this work, we did not use spatial autocorrelation structures during model fitting for GLM and GAM to reduce runtime.

5.3 The effect of overoptimistic performance estimates on ecological decision making

Unbiased model outcomes are the foundation of informed ecological decision-making, biodiversity conservation as well as renaturation strategies (Muenchow et al., 2018). In particular, reliable outcomes are indispensable in species distribution (Loehle, 2018), invasive species dispersal (Srivastava et al., 2018), and ecosystem service modeling (Watanabe & Ortega, 2014). Global change makes model predictions uncertain enough (IPCC, 2013). Therefore, it is unnecessary to introduce an additional autocorrelation bias, especially since one can relatively easy account for it. By contrast, the damage done in terms of monetary and ecological mismanagement due to biased advice is disproportionately harder to reverse.

6. Conclusion

A total of six statistical and machine-learning models have been compared in this study focusing on predictive performance. We proofed that non-spatial partitioning yields overoptimistic performance results if spatial autocorrelation is present.

Hyperparameter tuning did not always have an substantial effect on predictive performance of algorithms. The effect of hyperparameter tuning of machine-learning algorithms depends on the algorithm and dataset. The effect of hyperparameter tuning on predictive performance in this work was smaller than the differences between the algorithms. No substantial differences between spatial and non-spatial hyperparameter tuning were found. The magnitude of performance increase when performing hyperparameter tuning depends on the algorithm. However, hyperparameter tuning should always be performed us-

ing a sampling scheme that is consistent with the one used for performance estimation.

The assumption of higher predictive performances of machine-learning models was true for all algorithms besides SVM. Subsequently this statement only holds true partly.

Spatial CV should be used instead of non-spatial CV when working with spatial data to obtain bias-reduced predictive performance results for both hyperparameter tuning and performance estimation. Spatial autocorrelation led to substantial overoptimistic performance results for all algorithms if non-spatial CV was used. As modeling studies with an ecological context always deal with spatial data, the findings of the present work are important for any study that aims to report optimal and unbiased performance estimates. These are very important in helping taking correct actions in ecological decision making.

The findings of this study should be verified on additional datasets. In this regard it would be desirable to establish a database of spatial benchmark datasets.

Furthermore, we recommend to clearly identify the main goal of an analysis beforehand: If the goal is to understand environmental processes with the help of statistical inference, (semi-)parametric models should be favored even if they do not provide the best predictive accuracy. On the other hand, if the intention is to make highly accurate spatial predictions, spatially tuned machine-learning models should be considered for the task. We hope that this work motivates and helps scientists to report more bias-reduced performance estimates in the future.

7. Acknowledgments

This work was funded by the EU LIFE Healthy Forest project: LIFE14 ENV/ES/000179 and funding from the German Scholars Organization/Carl Zeiss Foundation awarded to A. Brenning.

8. Appendix

Appendix A. Descriptive summary of numerical and nominal predictor variables

Variable	**n**	**Min**	$\mathbf{q_1}$	$\widetilde{\mathbf{x}}$	$\bar{\mathbf{x}}$	$\mathbf{q_3}$	**Max**	**IQR**	**#NA**
temp	926	12.6	14.6	15.2	15.1	15.7	16.8	1.0	0
p_sum	926	124.4	181.8	224.6	234.2	252.3	496.6	70.5	0
r_sum	926	-0.1	0.0	0.0	0.0	0.0	0.1	0.1	0
elevation	926	0.6	197.2	327.2	338.7	455.9	885.9	258.8	0
slope_degrees	926	0.2	12.5	19.5	19.8	27.1	55.1	14.6	0
hail_prob	926	0.0	0.2	0.6	0.5	0.7	1.0	0.5	0
age	926	2.0	13.0	20.0	18.9	24.0	40.0	11.0	0
ph	926	4.0	4.4	4.6	4.6	4.8	6.0	0.4	0

Table A.2: Summary of numerical predictor variables. Precipitation (p_sum) in mm/m^2, temperature (temp) in °C, solar radiation (r_sum) in kW/m^2, tree age (age) in years. Statistics show sample size (**n**), minimum (**Min**), 25% percentile ($\mathbf{q_1}$), median ($\widetilde{\mathbf{x}}$), mean ($\bar{\mathbf{x}}$), 75% percentile ($\mathbf{q_3}$), maximum (**Max**), inner-quartile range (**IQR**) and NA Count (**#NA**).

Variable	**Levels**	**n**	%
diplo01	0	703	75.9
	1	223	24.1
	all	926	100.0
lithology	surface deposits	32	3.5
	clastic sedimentary rock	602	65.0
	biological sedimentary rock	136	14.7
	chemical sedimentary rock	143	15.4
	magmatic rock	13	1.4
	all	926	100.0
soil	soils with little or no profile differentiation (Cambisols, Fluvisols)	672	72.6
	pronounced accumulation of organic matter in the mineral topsoil (Chernozems, Kastanozems)	22	2.4
	soils with limitations to root growth (Cryosols, Leptosols)	19	2.0
	accumulation of moderately soluble salts or non-saline substances (Durisols, Gypsisols)	13	1.4
	soils distinguished by Fe/Al chemistry (Ferralsols, Gleysols)	35	3.8
	organic soil (Histosols)	14	1.5
	soils with clay-enriched subsoil (Lixisols, Luvisols)	151	16.3
	all	926	100.0
year	2009	401	43.3
	2010	261	28.2
	2011	102	11.0
	2012	162	17.5
	all	926	100.0

Table A.3: Summary of nominal predictor variables

Appendix B. Additional hyperparameter tuning results

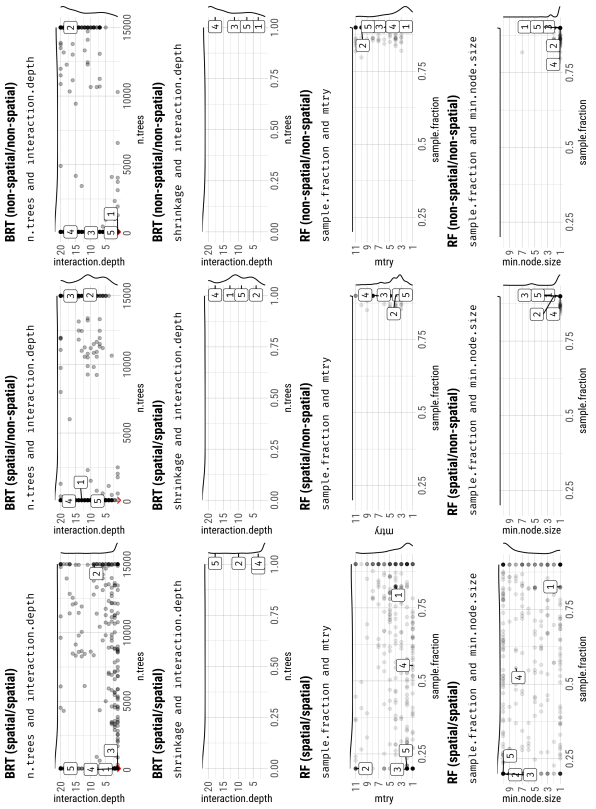

Figure B.7: Best hyperparameter settings by fold (500 total) each estimated from 100 (30/70) SMBO tuning iterations per fold using five-fold cross-validation. Split by spatial and non-spatial partitioning setup and model type. Red crosses indicate the default hyperparameters of the respective model. Black dots represent the winning hyperparameter setting of each fold. The labels ranging from one to five show the winning hyperparameter settings of the first five folds

Appendix C. SMBO optimization paths for all models

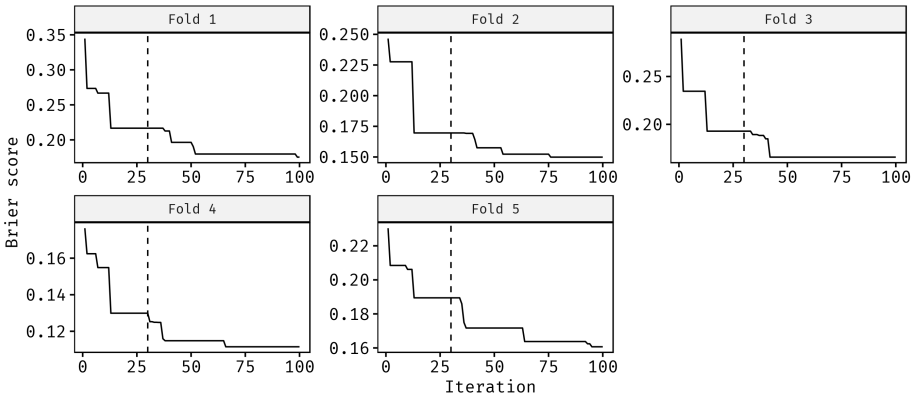

Figure C.8: SMBO optimization paths of the first five folds of the *spatial/spatial* for BRT. The dashed line marks the border between the initial design (30 randomly composed hyperparameter settings) and the sequential optimization part in which each setting was proposed using information from the prior evaluated settings.

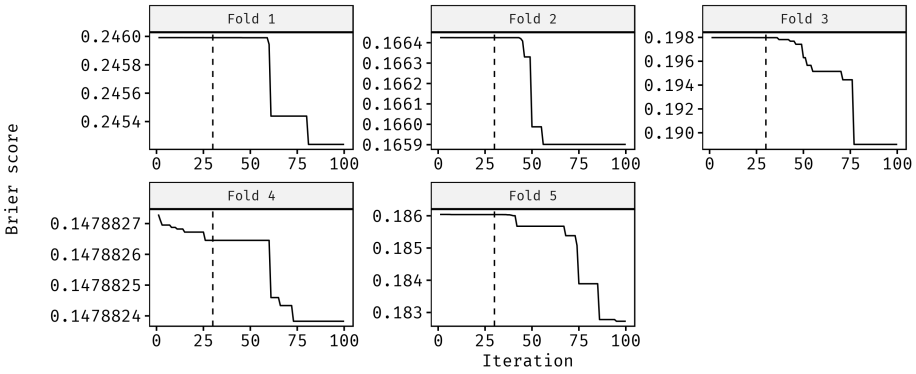

Figure C.9: SMBO optimization paths of the first five folds of the *spatial/spatial* for GAM. The dashed line marks the border between the initial design (30 randomly composed hyperparameter settings) and the sequential optimization part in which each setting was proposed using information from the prior evaluated settings.

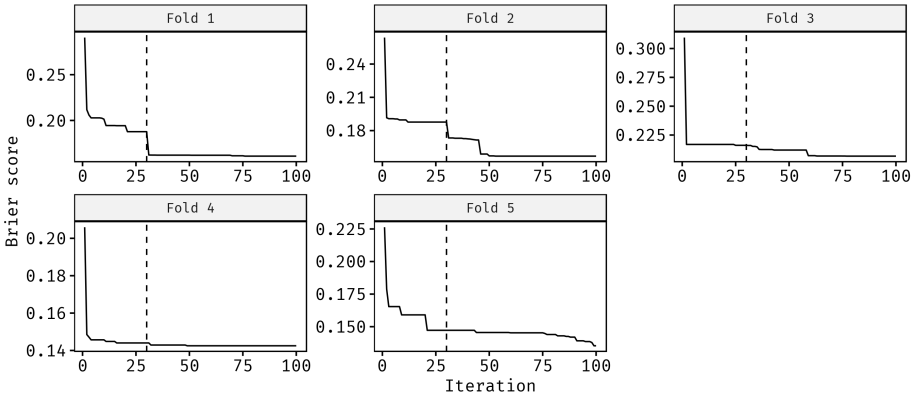

Figure C.10: SMBO optimization paths of the first five folds of the *spatial/spatial* for KNN. The dashed line marks the border between the initial design (30 randomly composed hyperparameter settings) and the sequential optimization part in which each setting was proposed using information from the prior evaluated settings.

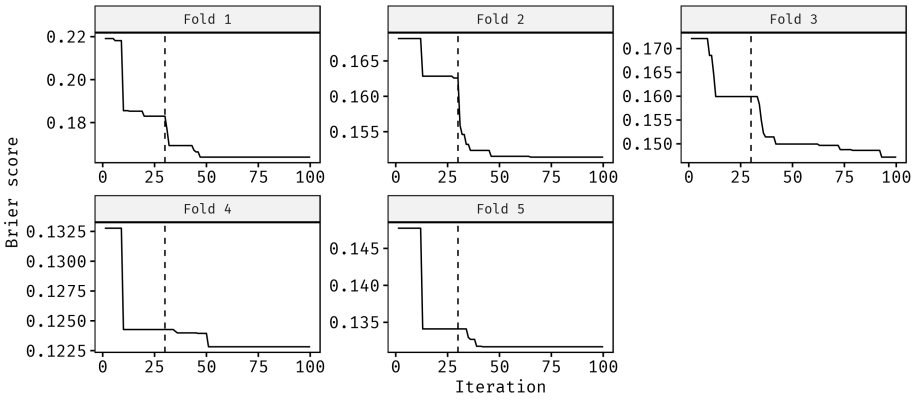

Figure C.11: SMBO optimization paths of the first five folds of the *spatial/spatial* for SVM. The dashed line marks the border between the initial design (30 randomly composed hyperparameter settings) and the sequential optimization part in which each setting was proposed using information from the prior evaluated settings.

References

Adler, W., Gefeller, O., & Uter, W. (2017). Positive reactions to pairs of allergens associated with polysensitization: Analysis of IVDK data with machine-learning techniques. *Contact Dermatitis*, *76*, 247–251. doi:10/gdq9ms.

Baasch, D. M., Tyre, A. J., Millspaugh, J. J., Hygnstrom, S. E., & Vercauteren, K. C. (2010). An evaluation of three statistical methods used to model resource selection. *Ecological Modelling*, *221*, 565–574. doi:10/bxkrb6.

Bahn, V., & McGill, B. J. (2012). Testing the predictive performance of distribution models. *Oikos*, *122*, 321–331. doi:10/f4qs6h.

Bengio, Y. (2000). Gradient-Based Optimization of Hyperparameters. *Neural Computation*, *12*, 1889–1900. doi:10/d42j94.

Bergstra, J., & Bengio, Y. (2012). Random Search for Hyper-parameter Optimization. *J. Mach. Learn. Res.*, *13*, 281–305.

Biau, G., & Scornet, E. (2016). A random forest guided tour. *TEST*, *25*, 197–227. doi:10/gdqdv3.

Birattari, M., Stützle, T., Paquete, L., & Varrentrapp, K. (2002). A racing algorithm for configuring metaheuristics. In *Proceedings of the 4th Annual Conference on Genetic and Evolutionary Computation* (pp. 11–18). Morgan Kaufmann Publishers Inc.

Bischl, B., Lang, M., Kotthoff, L., Schiffner, J., Richter, J., Studerus, E., Casalicchio, G., & Jones, Z. M. (2016). mlr: Machine learning in R. *Journal of Machine Learning Research*, *17*, 1–5.

Bischl, B., Richter, J., Bossek, J., Horn, D., Thomas, J., & Lang, M. (2017). mlrMBO: A Modular Framework for Model-Based Optimization of Expensive Black-Box Functions. *ArXiv e-prints*, . arXiv:1703.03373.

Breiman, L. (2001). Random Forests. *Machine Learning*, *45*, 5–32. doi:10/d8zjwq.

Brenning, A. (2005). Spatial prediction models for landslide hazards: Review, comparison and evaluation. *Natural Hazards and Earth System Science*, *5*, 853–862. doi:10/cjqtg8.

Brenning, A. (2012). Spatial cross-validation and bootstrap for the assessment of prediction rules in remote sensing: The R package sperrorest. In *2012 IEEE International Geoscience and Remote Sensing Symposium*. IEEE. doi:10.1109/igarss.2012.6352393 R package version 2.1.0.

Brenning, A., & Lausen, B. (2008). Estimating error rates in the classification of paired organs. *Statistics in Medicine*, *27*, 4515–4531. doi:10/dq5s7q. 00017.

Brenning, A., Schwinn, M., Ruiz-Páez, A. P., & Muenchow, J. (2015). Landslide susceptibility near highways is increased by 1 order of magnitude in the Andes of southern Ecuador, Loja province. *Natural Hazards and Earth System Sciences*, *15*, 45–57. doi:10/f6zrvm. 00023.

Brier, G. W. (1950). Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, *78*, 1–3. doi:10/fp62r6.

Brochu, E., Cora, V. M., & de Freitas, N. (2010). A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *CoRR*, *abs/1012.2599*.

Bui, D. T., Tuan, T. A., Klempe, H., Pradhan, B., & Revhaug, I. (2015). Spatial prediction models for shallow landslide hazards: A comparative assessment of the efficacy of support vector machines, artificial neural networks, kernel logistic regression, and logistic model tree. *Landslides*, *13*, 361–378. doi:10/f8nfwf.

Burman, P., Chow, E., & Nolan, D. (1994). A cross-validatory method for dependent data. *Biometrika*, *81*, 351–358. doi:10/fbfnmd.

Byrne, S. (2016). A note on the use of empirical AUC for evaluating probabilistic forecasts. *Electronic Journal of Statistics*, *10*, 380–393. doi:10/gdq9mw.

Cliff, A. D., & Ord, K. (1970). Spatial autocorrelation: A Review of existing and new measures with applications. *Economic Geography*, *46*, 269. doi:10/d93r2k.

De'ath, G. (2007). Boosted Trees for Ecological Modeling and Prediction. *Ecology*, *88*, 243–251. doi:10/c46943. 00657.

Dormann, C. F. (2007). Effects of incorporating spatial autocorrelation into the analysis of species distribution data. *Global Ecology and Biogeography*, *16*, 129–138. doi:10/czthw3.

Dormann, C. F., McPherson, J. M., Araújo, M. B., Bivand, R., Bolliger, J., Carl, G., Davies, R. G., Hirzel, A., Jetz, W., Kissling, W. D., Kühn, I., Ohlemüller, R., Peres-Neto, P. R., Reineking, B., Schröder, B., Schurr, F. M., & Wilson, R. (2007). Methods to account for spatial autocorrelation in the analysis of species distributional data: A review. *Ecography*, *30*, 609–628. doi:10/bnfhck.

Duarte, E., & Wainer, J. (2017). Empirical comparison of cross-validation and internal metrics for tuning SVM hyperparameters. *Pattern Recognition Letters*, *88*, 6–11. doi:10/f9xpcm.

Dudani, S. A. (1976). The distance-weighted k-Nearest-Neighbor rule. *IEEE Transactions on Systems, Man, and Cybernetics*, *SMC-6*, 325–327. doi:10/bjz668.

Efron, B. (1983). Estimating the error rate of a prediction rule: Improvement on cross-validation. *Journal of the American Statistical Association*, *78*, 316. doi:10/dsdfkt.

Elith, J., Leathwick, J. R., & Hastie, T. (2008). A working guide to boosted regression trees. *Journal of Animal Ecology*, *77*, 802–813. doi:10/fn6m6v.

European Commission, J. R. C. (2010). *'Map of Soil pH in Europe', Land Resources Management Unit, Institute for Environment & Sustainability*. 00000.

Ganley, R. J., Watt, M. S., Manning, L., & Iturritxa, E. (2009). A global climatic risk assessment of pitch canker disease. *Canadian Journal of Forest Research*, *39*, 2246–2256. doi:10/bmj3nk.

Ganuza, A., & Almendros, G. (2003). Organic carbon storage in soils of the Basque Country (Spain): The effect of climate, vegetation type and edaphic variables. *Biol. Fertil. Soils*, *37*, 154–162. doi:10/dqjnk3.

Geiß, C., Pelizari, P. A., Schrade, H., Brenning, A., & Taubenböck, H. (2017). On the effect of spatially non-disjoint training and test samples on estimated model generalization capabilities in supervised classification with spatial features. *IEEE Geoscience and Remote Sensing Letters*, *14*, 2008–2012. doi:10/gdq9m2.

GeoEuskadi (1999). *Litologia y Permeabilidad*.

Gneiting, T., & Raftery, A. E. (2007). Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, *102*, 359–378. doi:10/c6758w.

Goetz, J. N., Cabrera, R., Brenning, A., Heiss, G., & Leopold, P. (2015). Modelling landslide susceptibility for a large geographical area using weights of evidence in lower Austria, Austria. In *Engineering Geology for Society and Territory - Volume 2* (pp. 927–930). Springer International Publishing. doi:10.1007/978-3-319-09057-3_160.

Gordon, A. D., Breiman, L., Friedman, J. H., Olshen, R. A., & Stone, C. J. (1984). Classification and Regression Trees. *Biometrics*, *40*, 874. doi:10/b6z2qx.

Grotzinger, J., & Jordan, T. (2016). Sedimente und Sedimentgesteine. In *Press/Siever Allgemeine Geologie* (pp. 113–144). Springer Berlin Heidelberg. doi:10.1007/978-3-662-48342-8_5 00001.

Halvorsen, R., Mazzoni, S., Dirksen, J. W., Næsset, E., Gobakken, T., & Ohlson, M. (2016). How important are choice of model selection method and spatial autocorrelation of presence data for distribution modelling by MaxEnt? *Ecological Modelling*, *328*, 108–118. doi:10/gcz75b.

Hengl, T., de Jesus, J. M., Heuvelink, G. B. M., Gonzalez, M. R., Kilibarda, M., Blagotić, A., Shangguan, W., Wright, M. N., Geng, X., Bauer-Marschallinger, B., Guevara, M. A., Vargas, R., MacMillan, R. A., Batjes, N. H., Leenaars, J. G. B., Ribeiro, E., Wheeler, I., Mantel, S., & Kempen, B. (2017). SoilGrids250m: Global gridded soil information based on machine learning. *PLOS ONE*, *12*, e0169748. doi:10/f9qc5p.

Hutter, F., Hoos, H. H., & Leyton-Brown, K. (2011). Sequential model-based optimization for general algorithm configuration. In *Lecture Notes in Computer Science* (pp. 507–523). Springer Berlin Heidelberg. doi:10.1007/978-3-642-25566-3_40 00678.

IPCC (2013). Summary for Policymakers. In T. Stocker, D. Qin, G.-K. Plattner, M. Tignor, S. Allen, J. Boschung, A. Nauels, Y. Xia, V. Bex, & P. Midgley (Eds.), *Climate Change 2013: The Physical Science Basis. Contribution of Working Group I to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change* book section SPM. (pp. 1–30). Cambridge, United Kingdom and New York, NY, USA: Cambridge University Press. doi:10.1017/CBO9781107415324.004 00014.

Iturritxa, E., Mesanza, N., & Brenning, A. (2014). Spatial analysis of the risk of major forest diseases in Monterey pine plantations. *Plant Pathology*, *64*, 880–889. doi:10/gdq9pb.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An Introduction to Statistical Learning*. Springer New York. doi:10.1007/978-1-4614-7138-7 02966.

Jarnevich, C. S., Talbert, M., Morisette, J., Aldridge, C., Brown, C. S., Kumar, S., Manier, D., Talbert, C., & Holcombe, T. (2017). Minimizing effects of

methodological decisions on interpretation and prediction in species distribution studies: An example with background selection. *Ecological Modelling*, *363*, 48–56. doi:10/gcg2ff.

Johnson, J. B., & Omland, K. S. (2004). Model selection in ecology and evolution. *Trends in Ecology & Evolution*, *19*, 101–108. doi:10/cbzhrm. 02884.

Jones, D. R., Schonlau, M., & Welch, W. J. (1998). Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, *13*, 455–492. doi:10/fg68nc.

Karatzoglou, A., Smola, A., Hornik, K., & Zeileis, A. (2004). Kernlab – An S4 Package for Kernel Methods in R. *Journal of Statistical Software*, *11*, 1–20. doi:10/gdq9pc. R package version 0.9-25.

Kohavi, R., & others (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Ijcai* (pp. 1137–1145). Stanford, CA volume 14.

Kuhn, M., & Johnson, K. (2013). *Applied Predictive Modeling*. Springer New York. doi:10.1007/978-1-4614-6849-3 01181.

Legendre, P. (1993). Spatial autocorrelation: Trouble or new paradigm? *Ecology*, *74*, 1659–1673. doi:10/fsm4n5.

Legendre, P., & Fortin, M. J. (1989). Spatial pattern and ecological analysis. *Vegetatio*, *80*, 107–138. doi:10/ccpkqj.

Loehle, C. (2018). Disequilibrium and relaxation times for species responses to climate change. *Ecological Modelling*, *384*, 23–29. doi:10/gdvmpx. 00000.

Malkomes, G., Schaff, C., & Garnett, R. (2016). Bayesian optimization for automated model selection. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems 29* (pp. 2900–2908). Curran Associates, Inc.

Mets, K. D., Armenteras, D., & Dávalos, L. M. (2017). Spatial autocorrelation reduces model precision and predictive power in deforestation analyses. *Ecosphere*, *8*, e01824. doi:10.1002/ecs2.1824. 00002.

Meyer, H., Reudenbach, C., Hengl, T., Katurji, M., & Nauss, T. (2018). Improving performance of spatio-temporal machine learning models using forward feature selection and target-oriented validation. *Environmental Modelling & Software*, *101*, 1–9. doi:10.1016/j.envsoft.2017.12.001. 00004.

Micheletti, N., Foresti, L., Robert, S., Leuenberger, M., Pedrazzini, A., Jaboyedoff, M., & Kanevski, M. (2013). Machine learning feature selection methods for landslide susceptibility mapping. *Mathematical Geosciences*, *46*, 33–57. doi:10/gdq9pf.

Muenchow, J., Dieker, P., Kluge, J., Kessler, M., & von Wehrden, H. (2018). A review of ecological gradient research in the Tropics: Identifying research gaps, future directions, and conservation priorities. *Biodiversity and Conservation*, *27*, 273–285. doi:10/gcthf9. 00001.

Muenchow, J., Feilhauer, H., Bräuning, A., Rodríguez, E. F., Bayer, F., Rodríguez, R. A., & Wehrden, H. (2013). Coupling ordination techniques and GAM to spatially predict vegetation assemblages along a climatic gradient in an ENSO-affected region of extremely high climate variability. *Journal of vegetation science*, *24*, 1154–1166. 00015.

Múgica, J. R. M., Murillo, J. A., Ikazuriaga, I. A., Peña, B. n. E., Rodríguez, A. F., & Díaz, J. M. (2016). *Libro Blanco Del Sector de La Madera: Actividad Forestal e Industria de Transformación de La Madera. Evolución Reciente y Perspectivas En Euskadi*. Eusko Jaurlaritzaren Argitalpen Zerbitzu Nagusia, Servicio Central de Publicaciones del Gobierno VAsco, C/ Donostia-San Sebastián 1, 01010 Vitoria-Gasteiz. 00000.

Murase, H., Nagashima, H., Yonezaki, S., Matsukura, R., & Kitakado, T. (2009). Application of a generalized additive model (GAM) to reveal relationships between environmental factors and distributions of pelagic fish and

krill: A case study in Sendai Bay, Japan. *ICES Journal of Marine Science*, *66*, 1417–1424. doi:10/bvgptw. 00065.

Naghibi, S. A., Pourghasemi, H. R., & Dixon, B. (2016). GIS-based groundwater potential mapping using boosted regression tree, classification and regression tree, and random forest machine learning models in Iran. *Environmental monitoring and assessment*, *188*, 44.

Ninyerola, M., Pons, X., & Roure, J. (2005). *Atlas Climático Digital de Lapenínsula Ibérica. Metodología y Aplicaciones En Bioclimatología y Geobotánica.*. Universidad Autónoma de Barcelona, Bellaterra. 00000.

Peña, M., & Brenning, A. (2015). Assessing fruit-tree crop classification from Landsat-8 time series for the Maipo Valley, Chile. *Remote Sensing of Environment*, *171*, 234–244. doi:10/f745cg.

Pohjankukka, J., Pahikkala, T., Nevalainen, P., & Heikkonen, J. (2017). Estimating the prediction performance of spatial models via spatial k-fold cross validation. *International Journal of Geographical Information Science*, *31*, 2001–2019. doi:10/gdq9pg. 00001.

Pourghasemi, H. R., & Rahmati, O. (2018). Prediction of the landslide susceptibility: Which algorithm, which precision? *CATENA*, *162*, 177–192. doi:10/gcwqtx.

Probst, P., Bischl, B., & Boulesteix, A.-L. (2018a). Tunability: Importance of Hyperparameters of Machine Learning Algorithms. *ArXiv e-prints*, . arXiv:1802.09596. 00001.

Probst, P., Wright, M., & Boulesteix, A.-L. (2018b). Hyperparameters and Tuning Strategies for Random Forest. *ArXiv e-prints*, . arXiv:1804.03515. 00000.

Quillfeldt, P., Engler, J. O., Silk, J. R., & Phillips, R. A. (2017). Influence of device accuracy and choice of algorithm for species distribution modelling

of seabirds: A case study using black-browed albatrosses. *Journal of Avian Biology*, . doi:10/gct5qg.

R Core Team (2017). *R: A Language and Environment for Statistical Computing*. Vienna, Austria. 88058 R version 3.4.4.

Racine, J. (2000). Consistent cross-validatory model-selection for dependent data: Hv-block cross-validation. *Journal of Econometrics*, *99*, 39–61. doi:10/d45q6z.

Richter, J. (2017). mlrHyperopt: Easy hyperparameter optimization with mlr and mlrMBO, . R package version 0.1.1.

Ridgeway, G. (2017). *gbm: Generalized Boosted Regression Models*. R package version 2.1.3.

Roberts, D. R., Bahn, V., Ciuti, S., Boyce, M. S., Elith, J., Guillera-Arroita, G., Hauenstein, S., Lahoz-Monfort, J. J., Schröder, B., Thuiller, W., Warton, D. I., Wintle, B. A., Hartig, F., & Dormann, C. F. (2017). Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure. *Ecography*, *40*, 913–929. doi:10/gc4h8p.

Rojas-Dominguez, A., Padierna, L. C., Valadez, J. M. C., Puga-Soberanes, H. J., & Fraire, H. J. (2018). Optimal hyper-parameter tuning of SVM classifiers with application to medical diagnosis. *IEEE Access*, *6*, 7164–7176. doi:10/gdq9pm.

Ruß, G., & Brenning, A. (2010). Spatial variable importance assessment for yield prediction in precision agriculture. In *Advances in Intelligent Data Analysis IX* Lecture Notes in Computer Science (pp. 184–195). Springer, Berlin, Heidelberg. doi:10.1007/978-3-642-13062-5_18 00010.

Schliep, K., & Hechenbichler, K. (2016). *kknn: Weighted k-Nearest Neighbors*. R package version 1.3.1.

Schratz, P. (2016). *Modeling the Spatial Distribution of Hail Damage in Pine Plantations of Northern Spain as a Major Risk Factor for Forest Disease*. Ph.D. thesis Friedrich-Schiller-University Jena. doi:https://doi.org/10.5281/zenodo.814262 00000 (unpublished).

Schratz, P., & Iturritxa, E. (2018). Supplementary data for ECOMOD-18-226. *Mendeley datasets*, . doi:10/gdvmxg. 00000.

Shao, J. (1993). Linear model selection by cross-validation. *Journal of the American Statistical Association*, *88*, 486. doi:10/d47xdw.

Smoliński, S., & Radtke, K. (2016). Spatial prediction of demersal fish diversity in the Baltic Sea: Comparison of machine learning and regression-based techniques. *ICES Journal of Marine Science: Journal du Conseil*, (p. fsw136). doi:10/gdq9pp.

Srivastava, V., Griess, V. C., & Padalia, H. (2018). Mapping invasion potential using ensemble modelling. A case study on Yushania maling in the Darjeeling Himalayas. *Ecological Modelling*, *385*, 35–44. doi:10/gdvmrg. 00000.

Steger, S., Brenning, A., Bell, R., Petschko, H., & Glade, T. (2016). Exploring discrepancies between quantitative validation results and the geomorphic plausibility of statistical landslide susceptibility maps. *Geomorphology*, *262*, 8–23. doi:10/f8p6vn.

Stelmaszczuk-Górska, M., Thiel, C., & Schmullius, C. (2017). Remote sensing for aboveground biomass estimation in boreal forests. In *Earth Observation for Land and Emergency Monitoring* (pp. 33–55). John Wiley & Sons, Ltd. doi:10.1002/9781118793787.ch3.

Telford, R., & Birks, H. (2005). The secret assumption of transfer functions: Problems with spatial autocorrelation in evaluating model performance. *Quaternary Science Reviews*, *24*, 2173–2179. doi:10.1016/j.quascirev.2005.05.001. 00196.

Telford, R., & Birks, H. (2009). Evaluation of transfer functions in spatially structured environments. *Quaternary Science Reviews*, *28*, 1309–1316. doi:10/b87tzq.

Vapnik, V. (1998). The support vector method of function estimation. In *Nonlinear Modeling* (pp. 55–85). Springer US. doi:10.1007/978-1-4615-5703-6_3.

Vorpahl, P., Elsenbeer, H., Märker, M., & Schröder, B. (2012). How can statistical models help to determine driving factors of landslides? *Ecological Modelling*, *239*, 27–39. doi:10/fxvs2d.

Voyant, C., Notton, G., Kalogirou, S., Nivet, M.-L., Paoli, C., Motte, F., & Fouilloy, A. (2017). Machine learning methods for solar radiation forecasting: A review. *Renewable Energy*, *105*, 569–582. doi:10/gdq9px.

Watanabe, M. D. B., & Ortega, E. (2014). Dynamic emergy accounting of water and carbon ecosystem services: A model to simulate the impacts of land-use change. *Ecological Modelling*, *271*, 113–131. doi:10/f5kfvw. 00057.

Wenger, S. J., & Olden, J. D. (2012). Assessing transferability of ecological models: An underappreciated aspect of statistical validation. *Methods in Ecology and Evolution*, *3*, 260–267. doi:10/fzm72c.

Wieland, R., Kerkow, A., Früh, L., Kampen, H., & Walther, D. (2017). Automated feature selection for a machine learning approach toward modeling a mosquito distribution. *Ecological Modelling*, *352*, 108–112. doi:10/f96529.

Wingfield, M. J., Hammerbacher, A., Ganley, R. J., Steenkamp, E. T., Gordon, T. R., Wingfield, B. D., & Coutinho, T. A. (2008). Pitch canker caused by Fusarium circinatum – a growing threat to pine plantations and forests worldwide. *Australasian Plant Pathology*, *37*, 319. doi:10/b4dz77.

Wollan, A. K., Bakkestuen, V., Kauserud, H., Gulden, G., & Halvorsen, R. (2008). Modelling and predicting fungal distribution patterns using herbarium data. *Journal of Biogeography*, *35*, 2298–2310. doi:10/d9vqb5.

Wood, S. (2017). *Generalized Additive Models: An Introduction with R*. Chapman and Hall/CRC. 07117.

Wright, M. N., & Ziegler, A. (2017). ranger: A fast implementation of random forests for high dimensional data in c++ and r. *Journal of Statistical Software*, *77*, 1–17. doi:10/b8q3.

Yang, E.-S., Kim, J. D., Park, C.-Y., Song, H.-J., & Kim, Y.-S. (2017). Hyperparameter tuning for hidden unit conditional random fields. *Engineering Computations*, *34*, 2054–2062. doi:10/gbtm2n.

Youssef, A. M., Pourghasemi, H. R., Pourtaghi, Z. S., & Al-Katheeri, M. M. (2015). Erratum to: Landslide susceptibility mapping using random forest, boosted regression tree, classification and regression tree, and general linear models and comparison of their performance at Wadi Tayyah Basin, Asir Region, Saudi Arabia. *Landslides*, *13*, 1315–1318. doi:10/gdq9p2.