Performance evaluation and hyperparameter tuning of statistical and machine-learning models using 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

Machine-learning algorithms have gained popularity in recent years in the field of ecological modeling due to their promising results in predictive performance of classification problems. While the application of such 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, optimization of algorithms using hyperparameter tuning and spatial autocorrelation. We address these issues in the comparison of several widely used machine-learning algorithms such as Boosted Regression trees (BRT), k-Nearest Neighbor (WKNN), Random Forest (RF) and Support Vector Machine (SVM) to traditional parametric algorithms such as logistic regression (GLM) and semi-parametric ones like Generalized Additive Models (GAM). Different nested cross-validation methods including hyperparameter tuning methods are used to evaluate model performances with the aim to receive bias-reduced performance estimates. As a case study the spatial distribution of forest disease (*Diplodia sapinea*) in the Basque Country in Spain is investigated using common environmental variables such as temperature, precipitation, soil

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

or lithology as predictors.

Results show that GAM and Random Forest (RF) (mean AUROC estimates 0.708 and 0.699) outperform all other methods in predictive accuracy. The effect of hyperparameter tuning saturates at around 50 iterations for this data set. The AUROC differences between the bias-reduced (spatial cross-validation) and overoptimistic (non-spatial cross-validation) performance estimates of the GAM and RF are 0.167 (24%) and 0.213 (30%), respectively. It is recommended to also use spatial partitioning for cross-validation hyperparameter tuning of spatial data. The models developed in this study enhance the detection of *Diplodia sapinea* in the Basque Country compared to previous studies.

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

---

1. Introduction

Statistical learning has become an important tool in the process of knowledge discovery from big data in fields as diverse as finance or geomarketing (Heaton et al., 2016; Schernthanner et al., 2017), medicine (Leung et al., 2016), public administration (Maenner et al., 2016) and the sciences (Garofalo et al., 2016). We can classify statistical learning broadly into supervised and unsupervised techniques (e.g., ordination, clustering) (James et al., 2013). Though both fields are important in the spatial modeling field, we will focus in this paper on supervised predictive modeling and the comparison of (semi-)parametric models and machine learning techniques. Spatial predictions are of great importance in a wide variety of fields including geomorphology (Brenning et al., 2015), remote sensing (Stelmaszczuk-Górska et al., 2017), hydrology (Naghibi et al., 2016), epidemiology (Adler et al., 2017), climatology (Voyant et al., 2017), the soil sciences (Hengl et al., 2017) and of course ecology. Ecological applications range from species distribution models (Halvorsen et al., 2016; Quillfeldt et al., 2017; Wieland et al., 2017), predicting floristic (Muenchow et al., 2013a) and

faunal composition to disentangling the relationships between species and their environment (Muenchow et al., 2013b). Additional applications include biomass estimation (Fassnacht et al., 2014) and disease mapping as for example caused by fungal infections (Iturritxa et al., 2014). The latter marks the research area of this work.

Fungal species such as *Diplodia sapinea* inflict severe damage upon Monterrey pine trees (*Pinus radiata*) which trees are 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 play an important role in this process.

Supervised 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 interactions between the response and its predictors and to improve the general understanding of the modeled relationship. Model interpretability should certainly be an important criterion for choosing models when the analysis of relationships between a response variable such as species richness or species presence/absence and the corresponding environment is of interest (Goetz et al., 2015). While the most commonly used statistical models such as generalized linear models (GLMs) are parametric, especially machine learning techniques offer a non-parametric approach to spatial modeling in ecology. These have gained popularity due to their ability to handle high-dimensional and highly correlated data and the lack of explicit model assumptions. Some model comparison studies in the spatial modeling field suggest that machine learning models might be the better choice when the primary aim is accurate prediction (Hong et al., 2015; Smoliński & Radtke, 2016; Youssef et al., 2015). However, other studies found

no major performance difference to parametric models (Bui et al., 2015; Goetz et al., 2015).

The estimation of predictive performances and the tuning of model hyperparameters (where present) are two intertwined critical issues in ecological modeling and model comparisons, both of which are addressed in this study. Cross-validation and bootstrapping are two widely used performance estimation techniques (Brenning, 2005; Kohavi et al., 1995). However, in the presence of spatial autocorrelation, estimates obtained using regular (non-spatial) random resampling may be biased and overoptimistic, which has led to the adoption of spatial resampling in cross-validation and bootstrapping for bias reduction. Currently, different names are used in science for the same idea: Brenning (2005) named it "spatial cross-validation", Meyer et al. (2018) "Leave-location-out cross-validation" and Roberts et al. (2017) labels it "Block cross-validation". Although the importance of bias-reduced spatial resampling methods for performance estimation has been emphasized repeatedly in recent years (Geiß et al., 2017; Meyer et al., 2018; Wenger & Olden, 2012), such techniques have not been adopted in all cases (Bui et al., 2015; Pourghasemi & Rahmati, 2018; Smoliński & Radtke, 2016; Wollan et al., 2008; Youssef et al., 2015). Since default hyperparameter settings, which are used by some authors (Goetz et al., 2015; Ruß & Brenning, 2010; Ruß & Kruse, 2010; Vorpahl et al., 2012), can in no way guarantee an optimal performance of machine-learning techniques, additional attention should be directed to this potentially critical step. Again, performance estimation techniques such as cross-validation are used in this step, and the adequacy of non-spatial techniques for spatial data sets can be questioned. This work aims to be an exemplary model comparison study for spatial data using spatial cross-validation including spatial hyperparameter tuning to receive bias-reduced performance estimates. This approach is compared with cross-validation approaches that use other resampling strategies (i.e. random resampling) or conduct no hyperparameter tuning.

We provide the complete code (including a packrat file) in the supplementary material to make this work fully reproducible and to encourage a wider adoption

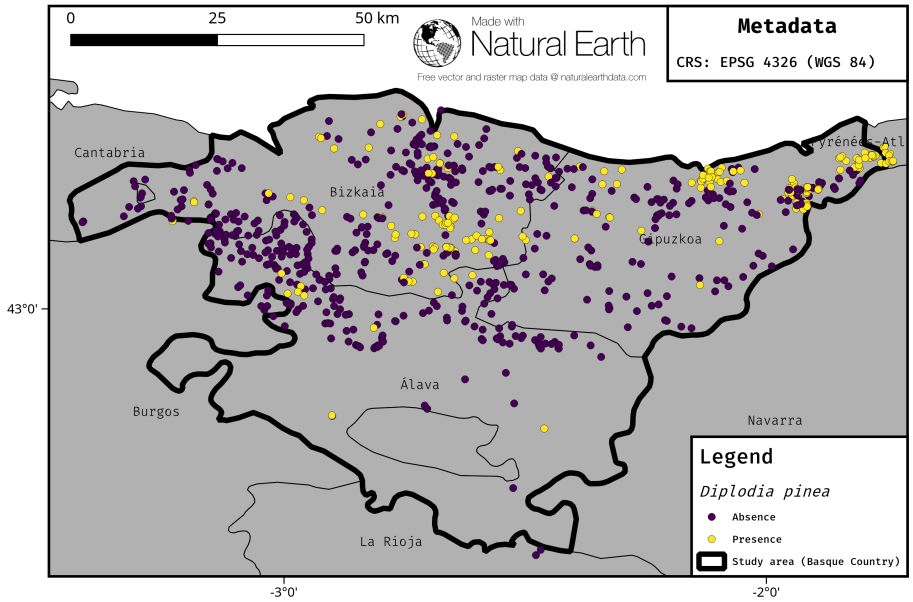

Figure 1: Spatial distribution of tree observations within the Basque Country, northern Spain, showing infection state by *Diplodia sapinea*.

of the proposed methodology. In our exemplary analysis we used a selection of six models (statistical and machine-learning) that are commonly used in the spatial modeling field: Boosted Regression Trees (BRT), Generalized Additive Model (GAM), Generalized Linear Model (GLM), Weighted k-nearest neighbor (WKNN), RF and Support Vector Machines (SVM).

2. Data and study area

2.1. Data

This study uses the data set from Iturritxa et al. (2014) to illustrate procedures and challenges that are common to many geospatial analyses problems: An uneven distribution of the binary response variable, influence of spatial autocorrelation and predictor variables derived from various sources (other modeling results, remote sensing data, surveyed information). It is representative

for many other ecological data sets in terms of sample size (926) and the number (11) and types of predictors (numeric as well as nominal). 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. Tree infection caused by fungal pathogens (here *Diplodia sapinea*) represents the response variable. The ratio of infected and non-infected trees in the sample is roughly 1:3 (223, 703). Compared to the original data set from Iturritxa et al. (2014), we added soil type (aggregated from 12 to 7 classes in accordance with the world reference base (Working Group WRB, 2015)) (Hengl et al., 2017), lithology type (condensed from 17 to 5 classes) (GeoEuskadi, 1999) and pH value of the soil (European Commission, 2010) to the already available predictors.

Iturritxa et al. (2014) showed that hail damage explained best pathogen infections in trees in the Basque Country. In this study hail damage was a binary predictor available as in-situ observations. To make it available as a predictor for the Basque country, we spatially predicted the hail damage potential as a function of climatic variables using a GAM (Schratz, 2016).

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 European Commission (2010) using a regression-kriging approach based on 12,333 soil pH measurements from 11 different sources. Spatial predictions utilized 54 auxiliary variables in the form of raster maps at a 1 km $\times$ 1 km resolution and were aggregated to a spatial resolution of 5 km $\times$ 5 km. Information about lithology types were extracted from a classification provided by GeoEuskadi that is based on the year 1999 (GeoEuskadi, 1999). Rock type condensing was done using the respective top level class for magmatic types and sub-classes for sedimentary rocks (Grotzinger & Jordan, 2016) (Table B.4).

We removed three observations due to missing information in some variables

leaving a total of 926 observations (Table B.3). The methodology we present in this work, i.e. a binary classification problem, can be easily adapted to multiclass problems as well as to quantitative response variables.

2.2. Study area

The Basque country in northern Spain represents our 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 (396.962 hectars), 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 using nested cross-validation (CV) and the use of spatial CV to assess bias-reduced model performances. 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 hyperparameter 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 a selection of commonly used machine learning algorithms (RF, SVM, WKNN, BRT) and the statistical methods GLM and GAM.

3.1. Cross-validation estimation of predictive performance

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 data set into training and test sets using a user-defined number of partitions (Figure 2). First, the data set 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.

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 (Brenning, 2005; Telford & Birks, 2005). Model performance estimates should be 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 (Brenning, 2012). Therefore, cross-validation approaches that adapt to this problem should be used in any kind of performance evaluation when spatial data is involved (Brenning, 2012; 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 2).

Five-fold partitioning repeated 100 times was chosen for performance estimation (Figure 2). 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 random search with a varying number of iterations (0, 10, 50, 100, 200) was applied to each fold of the tuning level. Model performances of every hyperparameter setting were computed at the tuning level and averaged across folds. The hyperparameter setting with the highest mean Area Under the Receiver Operating Characteristics Curve (AUROC) result 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 (100 repetitions with five folds each and varying random search iterations) to reduce the variance introduced by partitioning.

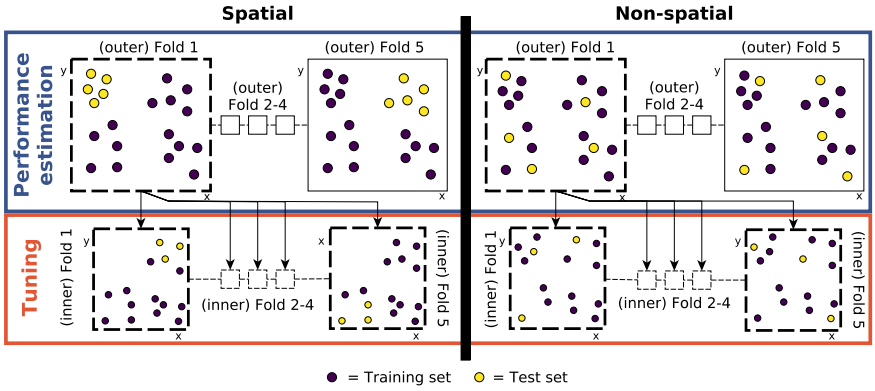

Figure 2: Theoretical concept of spatial and non-spatial nested cross-validation using five folds for hyperparameter tuning and performance estimation. Yellow/purple dots represent the training and test set for performance estimation, respectively. The tuning sample is based on the respective performance 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.

The AUROC was selected as a goodness of fit measure due to the binary response variable. The present methodology can also be applied with other measures than AUROC which are suited for binary classification. This measure combines both True Positive Rate (TPR) and False Positive Rate (FPR) of the classification and is also independent of a specific decision threshold (Candy & Breitfeller, 2013). A resulting AUROC value of close to 0.5 indicates no separation power of the model while a value of 1.0 would mean that all cases were correctly classified.

Hyperparameter tuning was performed for RF, SVM, BRT and WKNN. For GLM, no tuning is needed because the model has no hyperparameters and assumes a logit relationship between response and predictors. For GAM, see subsubsection 3.4.5.

3.2. Tuning of hyperparameters

Determining the optimal (hyperparameter) settings for each model is crucial for the bias-reduced assessment of a model's predictive power. While (semi-)parametric algorithms cannot be tuned in the same way as machine-learning algorithms (although some perform an internal optimization, e.g. the implementation of the GAM in the *mgcv* package from Wood (2006)), hyperparameters of machine-learning algorithms need to be tuned to achieve optimal performances (Bergstra & Bengio, 2012; Duarte & Wainer, 2017; Hutter et al., 2011). Note that for parametric models the term "parameter" is often used to refer to the regression coefficients of each predictor in the fitted model. 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. These different usages often lead to confusion and hence both terms should be used with caution. Hyperparameters are determined by finding the optimal value for a model across multiple unknown data sets by using a

Table 1: Hyperparameter limits and types for each model. Notations of hyperparameters from the respective R packages were used.

Algorithm (package)	Hyperparameter	Type	Value	Start	End
BRT (gbm)	`n.tree`	integer	-	100	10000
	`shrinkage`	numeric	-	0	1.5
	`interaction.depth`	integer	-	1	40
RF (ranger)	`mtry`	integer	-	1	11
	`num.trees`	integer	-	10	10000
SVM (kernlab)	`C`	numeric	-	2^{-12}	2^{15}
	σ	numeric	-	2^{-15}	2^{6}
WKNN (kknn)	`k`	integer	-	10	400
	`distance`	integer	-	1	100
	`kernel`	nominal	*		

* triangular, Epanechnikov, biweight, triweight, cos, inv, Gaussian, optimal

optimization procedure such as CV or Bayesian optimization while parameters of parametric models are estimated when fitting them to the data (Kuhn & Johnson, 2013).

We used a random search with a varying number of iterations (10, 50, 100, 200, 300, 400) for all machine learning models in this study to analyze the difference of varying tuning iterations. A random search has desirable properties in high dimensional and no disadvantages in low dimensional situations compared to a grid search (Bergstra & Bengio, 2012). This is due to the fact that often high dimensional situations have a "low effective dimension", i.e. only a subset of the hyperparameters is actually relevant. Another practical advantage is that one does not have to set the step size for the grid but only the parameter limits. We did not perform stepwise variable selection or similar for the parametric models (GLM, GAM) as we required all models to have the same predictor set. An exploratory analysis was done on using different starting basis dimensions for the optimal smoothing estimation of each predictor of the GAM. The *mgcv* package does an internal optimization of the smoothing degree value using the supplied basis dimensions as the starting point. The reported GAM model was initiated with $k = 10$ as the basis dimension which ensured full flexibility of the smoothing terms for each predictor. Please note that although we attributed the GAM to settings *non-spatial/no tuning* and *spatial/no tuning* as we did not perform a tuning ourselves, the GAM actually does a non-spatial optimization of the smoothing degrees for each predictor. We are aware that this attribution is somewhat contrary to the attribution of all other algorithms in this study. Strictly, we would also need to implement a spatial optimization procedure of the smoothing degrees for the GAM to follow our philosophy of spatial hyperparameter tuning in this work. However, such an implementation exceeds the scope of this work. Belonging to the parametric algorithm group, we decided to attribute the GAM to the "no tuning" class and leave all the tuning settings to the machine-learning models.

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 of the *kernlab* package. The ranges of the tuning spaces were set by iteratively checking the tuning results and adjusting the search space to make sure that the resulting optimal hyperparameter settings of each fold are not possibly limited by the defined search space. However, in practice this is sometimes impossible (see the problems we faced for WKNN and BRT in subsection 3.4) because models start to fail if hyperparameter values outside of computationally valid ranges are tested.

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*.

3.3. 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 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 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.

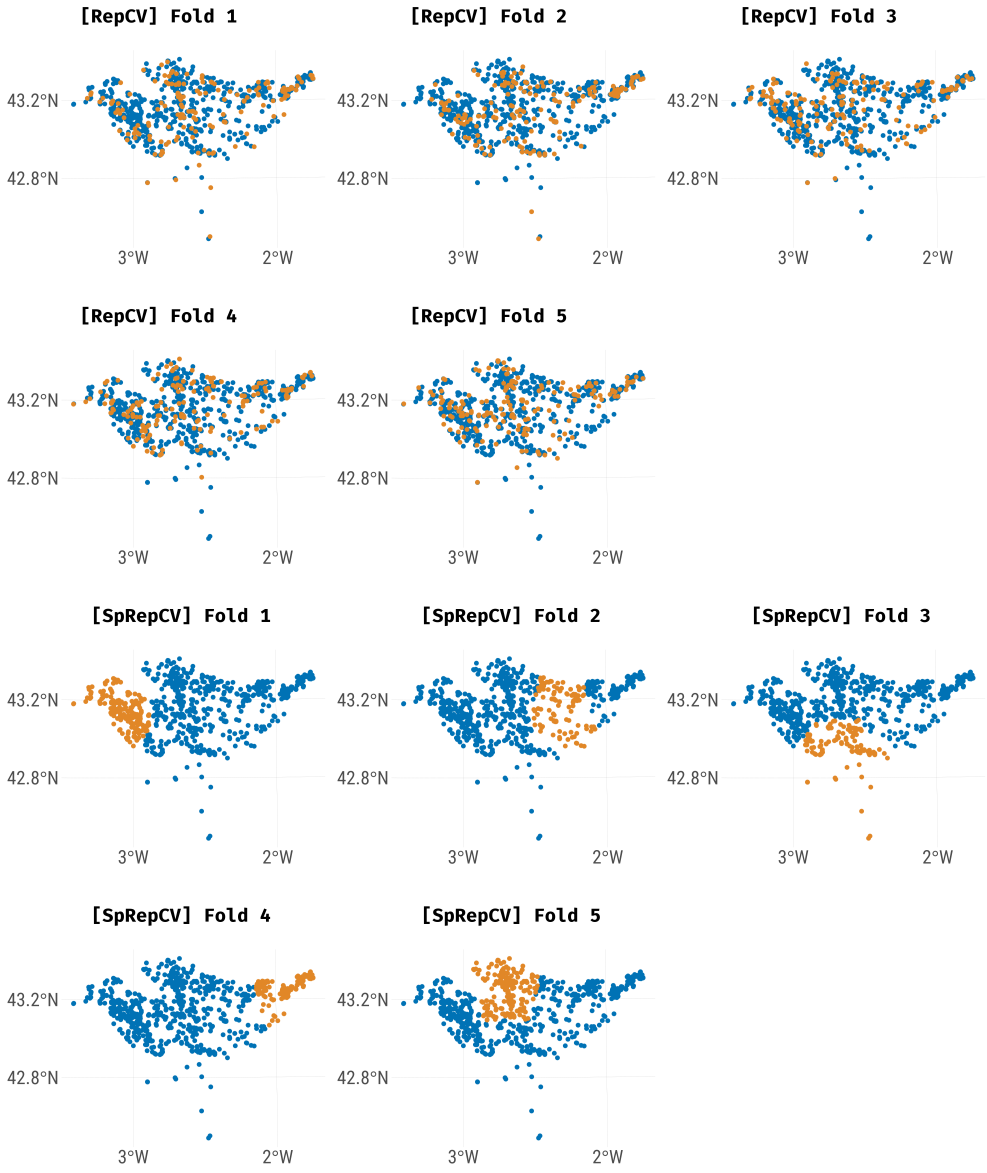

Figure 3: Comparison of spatial and non-spatial partitioning of the first five folds in spatial and non-spatial cross-validation performance estimation. Yellow/purple dots represent the training and test set, respectively.

3.4. Model characteristics and hyperparameters

An exemplary selection of widely-used statistical and machine-learning techniques was compared in this study. While the following sections describe the used models and their settings, a justification of the choice of specific implementations in the statistical software R is included in Appendix A. We used the open-source statistical programming language R (R Core Team, 2017) for all analyses and the packages *gbm* (Ridgeway, 2017) (BRT), *mgcv* (Wood, 2006) (GAM), *kernlab* (Karatzoglou et al., 2004) (SVM), *kknn* (Schliep & Hechenbichler, 2016) (WKNN), and *ranger* (Wright & Ziegler, 2017) (RF). We have integrated the spatial partitioning functions of the *sperrorest* package 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.

3.4.1. Random Forest

Classification trees are a non-linear technique that uses binary decision rules to predict a class based on the given predictors (Gordon et al., 1984). RF aggregates many classification trees by counting the votes of all individual trees. The class with the most votes wins and will be the predicted class. Fitting a high number of trees is then referred to as fitting a 'forest' in a metaphorical way. Using many trees stabilizes the model (Breiman, 2001). However, RF saturates at a specific number of trees, meaning that adding more trees will not increase its performance anymore but only increases computing time. Randomness is introduced in two ways: First a bootstrap sample ob observations is drawn for each tree. Second, for each node only a random subset of m_{try}) variables is considered for generating the decision rule (Breiman, 2001).

3.4.2. Support Vector Machines

SVMs transform the data in a high-dimensional feature space by performing non-linear transformations of the predictor variables (Vapnik, 1998). In this

high-dimensional setting, classes are linearly separated using decision hyperplanes. The tuning of SVMs is important and not trivial due to the sensitivity of the hyperparameters across a wide search space (Duan et al., 2003).

We decided to use the Radial Basis Function (RBF) kernel (also known as Gaussian kernel) which is the default in most implementations and most commonly used in the literature (Meyer et al., 2017; Guo et al., 2005; Pradhan, 2013). For this kernel, the regularization parameter C and bandwith σ, which control the degree of non-linearity, are the hyperparameters which have to be optimized. An exploratory analysis of the Laplace and Bessel kernels was done, which confirmed the expected insensitivity to the choice of the kernel. All these kernels (including the RBF kernel) are classified as "general purpose kernels" (Karatzoglou et al., 2004).

3.4.3. Boosted Regression Trees

BRT are different from RF in that trees are fitted on top of previous trees instead of being fitted parallel to each other without a relation to adjacent trees. In this iterative process, each tree learns from the previous fitted trees by a magnitude specified by the *shrinkage* parameter (Elith et al., 2008). This process is also called 'stage-wise fitting' (not step-wise) because the previous fitted trees remain unchanged while additional trees are added. BRT have a tendency towards overfitting the more trees are added. Therefore, a combination of a small learning rate with a high number of trees is preferable. BRT acts similar as a GLM as it can be applied to several response types (binomial, Poisson, Gaussian, etc.) using a respective link function. Also, the final model can be seen as a large regression model with every tree being a single term (Elith et al., 2008). Hyperparameter tuning was performed on the learning rate *shrinkage*, the number of trees *n.tree* and the interaction depth between the variables *interaction.depth*.

3.4.4. Weighted k-Nearest Neighbor

WKNN identifies the K-nearest neighbors within the training set for a new observation to predict the target class based on the majority class among the neighbors. The first formulation of the algorithm goes back to Fix & Hodges (1951). Besides the standard hyperparameter *number of neighbors* ($n_{neighbors}$), the implementation by Schliep & Hechenbichler (2016) also provides hyperparameter (*distance*) that allows to set the Minkowski distance and a choice between different kernels (up to 12, see Table 1). Hyperparameter *distance* helps finding the k-nearest training set vectors which are used for classification together with the maximum of the summed kernel densities provided by hyperparameter *kernel* (Schliep & Hechenbichler, 2016). Training observations that are closer to the predicted observation get a higher weight in the decision process, when a kernel other then *rectangular* is chosen. The original idea of the WKNN algorithm goes back to Dudani (1976).

Including weighting and kernel functions may increase predictive accuracy but can also lead to overfitting of the training data.

3.4.5. Generalized Linear Model and Generalized Additive Models

GLMs extend linear models by allowing also non-Gaussian distributions, e.g., binomial, Poisson or negative binomial distributions, for the response variable. The option to apply a custom link function between the response and the predictors already allows for some degree of non-linearity. GAMs are an extension of GLMs allowing the response-predictor relationship to become fully non-linear. For more details please refer to Zuur et al. (2009); Wood (2006); James et al. (2013).

4. Results

4.1. Tuning

While ten (or more) hyperparameter tuning iterations substantially improved the performance of BRT and SVM classifiers compared to default hyperparameter values, WKNN and RF hyperparameter tuning did not result in

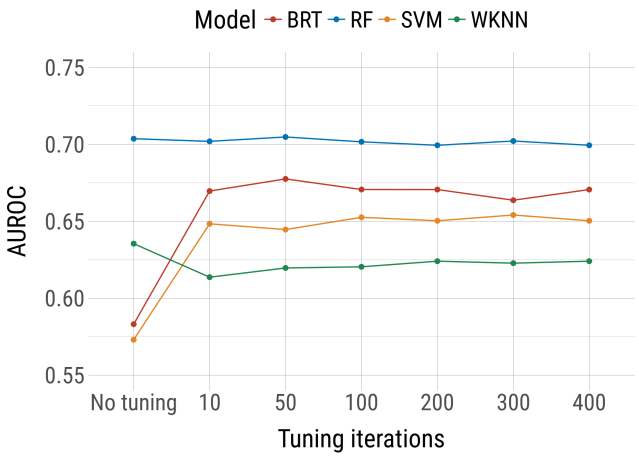

Figure 4: Hyperparameter tuning results of the *spatial/spatial* CV setting for BRT, WKNN, RF and SVM: Number of tuning iterations (1 iteration = 1 random hyperparameter setting) vs. predictive performance (AUROC).

relevant changes in AUROC (Figure 4). Fifty tuning iterations and more further improved accuracies only slightly (WKNN) or not at all (SVM, BRT). SVM showed the highest tuning effect of all models with an increase of ~0.08 AUROC (Figure 4).

There were notable differences in the estimated optimal hyperparameters between the spatial (*spatial/spatial*) and non-spatial (*spatial/non-spatial*, *non-spatial/non-spatial*) tuning settings (Figure 5). For example when being spatially tuned, the estimated m_{try} values of RF mainly ranged between 1 and 3 with $m_{try} = 1$ being chosen most often. In contrast, in a non-spatial tuning situation m_{try} was mainly favored between 2 and 4 with $m_{try} = 3$ being the mode setting.

4.2. Predictive performance

For the spatial settings (*spatial/spatial* and *spatial/no tuning*), GAM and RF show the best predictive performance followed by GLM, SVM and WKNN (Figure 6). The absolute difference between the best (RF/GAM) and worst

(WKNN) performing model in our setup is 0.081 (mean AUROC, WKNN vs. RF/GAM) (Table 2).

The tuning of hyperparameters resulted in a clear increase of predictive performance for BRT (0.661 (*spatial/spatial*) vs. 0.587 (*spatial/no tuning*) AUROC) and SVM (0.654 (*spatial/spatial*) vs 0.574 (*spatial/no tuning*) AUROC) (Table 2). The type of partitioning for hyperparameter tuning (spatial (*spatial/spatial*) or non-spatial (*spatial/non-spatial*)) only had an substantial impact for SVM (Figure 6).

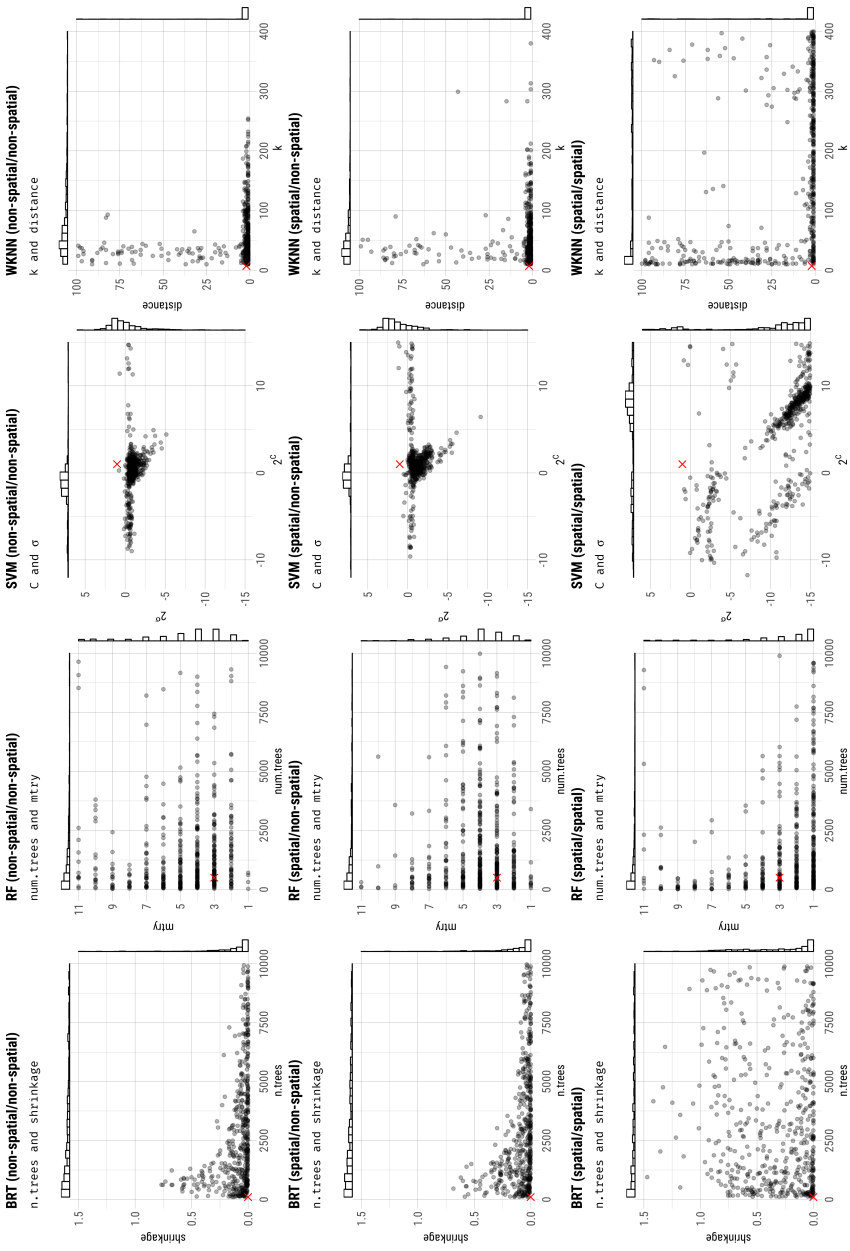

Figure 5: Best hyperparameter settings by fold (500 total) each estimated from 400 random search tuning iterations per fold using five-fold cross-validation. Split by spatial and non-spatial partitioning setup and model type. Red crosses indicate default hyperparameter values of the respective model. Black dots represent the winning hyperparameter setting out of each random search tuning of the respective fold.

Table 2: Mean AUROC (repetition level) for different 5-fold 100 times repeated cross-validation settings. Settings with tuning are based on 400 random search iterations. Highest values of each column are highlighted in bold. Note that non-spatial performance estimation is over-optimistic.

Performance estimation	Non-Spatial		Spatial		
Hyperparameter tuning	Non-Spatial	None	Non-Spatial	Spatial	None
GLM	-	0.859	-	-	0.665
GAM	-	0.874	-	-	**0.708**
BRT	0.908	0.792	**0.699**	0.671	0.583
RF	**0.912**	**0.913**	0.698	**0.699**	0.704
SVM	0.878	0.881	0.563	0.650	0.573
WKNN	0.872	0.870	0.657	0.624	0.635

Predictive performance estimates based on non-spatial partitioning (*non-spatial/non-spatial* or *non-spatial/no tuning*) are around 24 - 39% higher, i.e. overoptimistic, compared to their spatial equivalents (*spatial/spatial*). BRT and WKNN show the highest differences between these two settings (35% and 39%, respectively) while the GAM is least affected (24%).

5. Discussion

5.1. Tuning

Hyperparameter tuning becomes more and more expensive in terms of computing time with an increasing number of iterations. Hence, the goal is to use as few tuning iterations as possible to find a nearly optimal hyperparameter setting for a model for a specific data set. In this respect, random search algorithms are particularly promising in multidimensional hyperparameter spaces with possibly redundant or insensitive hyperparameters (low effective dimensionality; (Bergstra & Bengio, 2012). These as well as adaptive search algorithms offer computationally efficient solutions to these difficult global optimization problems in which little prior knowledge on optimal subspaces is avail-

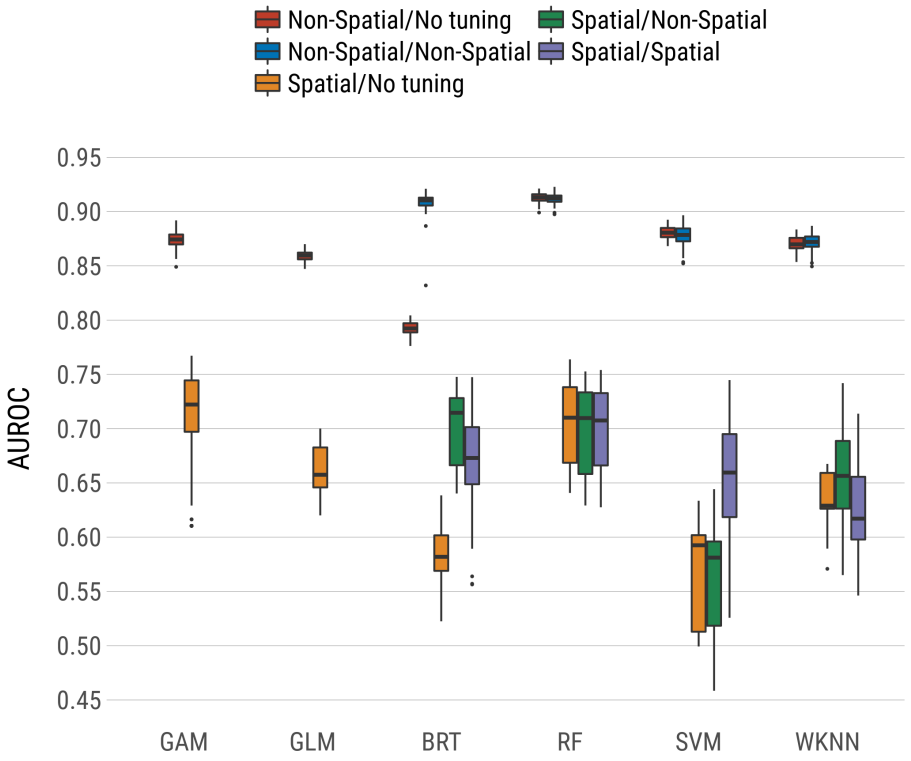

Figure 6: (Nested) CV estimates of model performance at the repetition level using 400 random search iterations. CV setting refers to performance estimation/hyperparameter tuning of the respective (nested) CV, e.g. "Spatial/Non-Spatial" means that spatial partitioning was used for performance estimation and non-spatial partitioning for hyperparameter tuning.

able. Bayesian Optimization and F-racing are other approaches that are widely used for optimization of black-box models (Birattari et al., 2002; Brochu et al., 2010; Malkomes et al., 2016). In this study, a random search with at least 50 iterations was sufficient for all considered algorithms.

Depending on the data set characteristics, some models (e.g. RF) can be insensitive to hyperparameter tuning (Biau & Scornet, 2016; Díaz-Uriarte & De Andres, 2006). As the effect of hyperparameter tuning always depends on the data set characteristics, we recommend to always tune hyperparameters. If

no tuning is conducted, it cannot be ensured that the respective model showed its best possible predictive performance on the data set.

Computing power, especially when conducting a random search, should focus on plausible parameters for each model. It should be ensured by visual inspection that the majority of the obtained optimal hyperparameter settings does not range closely to the limits of the tuning space. If the optimal hyperparameter settings are clustered at the edge of the parameter limits, this implies that optimal hyperparameters may actually lie outside the given range. However, extending the tuning space is not always possible nor practical as numerical problems within the algorithm may occur that may prohibit further extension of the tuning space. This especially applies to models with a numerical search space (e.g. SVM). In a practical sense one has to question oneself if extending the parameter ranges could possibly result in a significant performance increase and is worth the disadvantage of having an increased runtime. All these points show the need for a thorough specification of parameter limits for hyperparameter tuning. As the optimal parameter limits also depend on the dataset characteristics, it is not possible to define an optimal search space for an algorithm upfront. The chosen parameter limits of this work can serve as a starting point for future analysis but do not claim to be universally applicable. Users should analyze parameter search spaces of various studies to find suitable limits that match their dataset characteristics. Within the framework of the *mlr* project a database exists which stores tuning setups of various models from users that can serve as a reference point (Richter, 2017).

While in our study no major differences in model performances were found when using spatial versus non-spatial hyperparameter tuning procedures (e.g. 0.03 for BRT (0.624 vs. 0.652 AUROC), we recommend using the same (spatial) cross-validation procedure in the inner (tuning) cross-validation step as in the outer (performance estimation). Generally spoken, hyperparameters from a non-spatial tuning lead to models which are more adapted to the training data than models with hyperparameters estimated from a spatial tuning. Models fitted with hyperparameters from a non-spatial tuning can then profit from

the remaining spatial autocorrelation in the train/test split during performance estimation (compare results of settings *spatial/non-spatial* and *spatial/spatial* of BRT in Figure 3). Some software implementations (e.g., the SVM implementation of the *kernlab* package) provide an automated non-spatial CV for hyperparameter tuning. However, this is only useful for data without spatial and temporal dependencies.

Tuning of RF had no substantial effect on predictive performance in this study. Nevertheless, the estimated optimal hyperparameters of RF differ for the non-spatial and spatial tuning setting (Figure 5). In a non-spatial tuning setting, RF will prioritize spatially autocorrelated predictors as these will perform best in the optimization of the *Gini impurity measure* (Biau & Scornet, 2016; Gordon et al., 1984). 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 which would cause a bad performance on the test set. This means that mainly the predictors which profit from spatial autocorrelation will be selected. Although applying these non-spatially optimized hyperparameters on the spatially partitioned performance estimation fold has no advantages in predictive performance compared to using the spatially tuned hyperparameters, the resulting model will have a different structure. In the spatial tuning setting, mainly `mtry = 1` is chosen. This specific setting essentially removes the internal variable selection process by `mtry` as RF is forced to use the randomly chosen predictor. 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) is reduced. 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).

5.2. Predictive Performance

In this study we compared the predictive performance of six models using five different CV setups (subsection 4.2).

Our findings agree with previous studies in that non-spatial performance estimates appear to be substantially "better" than spatial performance estimates. However, this difference can be attributed to an overoptimistic bias in non-spatial performance estimates in the presence of spatial autocorrelation. (add references) Spatial cross-validation is therefore recommended 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 (Brenning & Lausen, 2008; Meyer et al., 2018; Peña & Brenning, 2015).

Although hyperparameter tuning certainly increases the predictive performance for some models (e.g. BRT and SVM) in our case, the magnitude always depends on the meaningful/arbitrary defaults of the respective algorithm and the characteristics of the data set. For SVM, we refrained from using automatic tuning algorithms (e.g. *kernlab* package) or optimized default values (e.g. Meyer et al. (2017)) for all "no tuning" settings. While the *kernlab* approach clearly violates the "no tuning" criterion, there are no globally accepted default values for σ and `C`. Subsequently we set both σ and `C` to an arbitrary value of 1. Naturally, the tuning effect is higher for models without meaningful defaults (such as BRT and SVM) than for models with meaningful defaults such as RF. Aside from the optimization of predictive performance the aim of hyperparameter tuning is the retrieval of bias-reduced performance estimates.

The biased-reduced outcomes of RF (*spatial/spatial* setting) and the GAM (*spatial/no tuning* setting) showed the best predictive performance in our study. Various other ecological modeling studies confirm the finding that RF is among the best performing models (Bahn & McGill, 2012; Jarnevich et al., 2017; Smoliński & Radtke, 2016; Vorpahl et al., 2012). It is noteworthy that the performance of the GLM is close to the one of the GAM and RF for this dataset.

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.

Comparing the results of this work to the study of Iturritxa et al. (2014), an increase in AUROC of ~0.05 AUROC was observed (comparing the spatial CV result of the GLM from this study to the spatial CV result of *Diplodia sapinea* without predictor *hail* from Iturritxa et al. (2014)). However, the gain in performance is minimal if predictor *hail_prob* is removed from the model of this study (0.667 (this work) vs. 0.659 Iturritxa et al. (2014) AUROC). Subsequently, the influence of the additional predictors *slope*, *soil*, *lithology* and *pH* that were added to this study is negligible small. The relatively small performance increase of predictor *hail_prob* (0.667 to 0.694 AUROC) compared to predictor *hail* (0.659 to 0.962 AUROC) from Iturritxa et al. (2014) can be explained by the high correlation of the latter (0.93) with the response. This inherits from the binary type of the response and predictor *hail*. The spatially modeled predictor *hail_prob* of this work is of type numeric (probabilities) and therefore shows a much lower correlation to the response. In summary, the inclusion of the new predictors increased the predictive accuracy by 0.05 AUROC compared to Iturritxa et al. (2014).

We want to highlight the importance of spatial partitioning for an bias-reduced estimate of model performance. If only non-spatial CV had been used in this study, the main results of this study would look as follows: (i) The best model would have been RF instead of GAM. (ii) The predictive performance would have been reported with a mean value of 0.912 AUROC which is ~0.204 (29%) AUROC higher than the best bias-reduced performance estimated by spatial CV (*spatial/spatial*) (0.708 AUROC, GAM).

5.3. Other Model Evaluation Criteria

This work focuses only on the evaluation of models by comparing their predictive performances. However, in practice other criteria exist that might influence the selection of a algorithm for a specific data set in a scientific field.

Using multiple performance measures suited for binary classification may be a possible enhancement. However, looking at possible invariances (invariance = not being sensible to changes in the confusion matrix) of performance measures, Sokolova & Lapalme (2009) found that AUROC is among the best suitable measures for binary classification in all tested scenarios. This is the reason why most model comparison studies with a binary response (e.g. Goetz et al. (2015); Smoliński & Radtke (2016)) only use AUROC as a single error measure.

High predictive performance does not always mean that a model also has a high practical plausibility. Steger et al. (2016) showed that in the field of landslide modeling, models achieving high AUROC estimates may have a low geomorphic plausibility.

Although the process of automated variable selection is not a criterion that can be compared in a quantitative way, users should always be aware of the selection process of predictor variables when interpreting the plausibility of a model in the ecological modeling field. While in our case the predictor variables have been selected by expert knowledge, automated variable selection processes (e.g. stepwise variable selection) for parametric models may lead to potentially biased input data (Steger et al., 2016). As a consequence, the user might receive high performance estimates with unrealistic susceptibility maps (Demoulin & Chung, 2007).

Another non-quantitative model selection criterion within the spatial modeling field is the surface quality of a predicted map. Homogeneous prediction surfaces might be favored over predictive power if the difference is acceptable small. Inhomogeneous surfaces can be an indicator for a poor plausibility of the predicted map, simply caused by the nature of the algorithm (e.g. RF) which splits continuous predictors into classes (Steger et al., 2016). In comparison a spatial prediction map from a GAM, GLM or SVM shows much smoother

prediction surfaces.

5.4. Model Interpretability

Although there is an ongoing discussion about the usage of parametric vs. non-parametric models in the field of ecological modeling (Perretti & Munch, 2015), most studies prefer parametric ones due to the ability to interpret relationships between the predictors and the response (Aertsen et al., 2010; Jabot, 2015). However, when interpreting the coefficients of (semi-)parametric spatial models (e.g. GLM, GAM), spatial autocorrelation structures should be included within the model fitting process (e.g. possible in R with *MASS::glmmPQL()* or *mgcv::gamm()*). Otherwise, the independence assumption might be violated which in turn might lead to biased coefficients and p-values and hence wrong (ecological) conclusions (Cressie, 1993; Dormann et al., 2007; Telford & Birks, 2005).

Variable importance information as provided by machine-learning algorithms is only suitable to provide an overview of the most important variables but does not give detailed information about the predictor-response relationships (Hastie et al., 2001). Using the concept of variable permutation during cross-validation (Brenning, 2012), Ruß & Brenning (2010) showed how to analyze variable importance of machine-learning models in the context of spatial prediction.

6. Conclusion

A total of six statistical and machine-learning models have been compared in this study focusing on predictive performance. For our test case, all machine learning models outperformed parametric models in terms of predictive accuracy with RF and GAM showing the best results. The effect of hyperparameter tuning of machine learning models depends on the algorithm and data set. However, it should always be performed using a suitable amount of iterations and well defined parameter limits. The accuracy of detecting *Diplodia sapinea* was increased by 0.05 AUROC compared to Iturritxa et al. (2014) with predictor

"hail damage at trees" being the main driver. Spatial CV should be favored over non-spatial CV when working with spatial data to obtain bias-reduced predictive performance results for both hyperparameter tuning and performance estimation. Furthermore, we recommend to be clear on the analysis aim before conducting spatial modeling: 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 project Healthy Forest: LIFE14 ENV/ES/000179.

8. Appendix

Appendix A. Package selection

Appendix A.1. Random Forest

Several RF implementations exist in R. We used package *ranger* because of its fast runtime. The RF implementation in package *ranger* is up to 25 times faster, taking number of observations as benchmark criteria, and up to 60 times if hyperparameter n_{trees} is the benchmark measure, respectively, compared to package *randomForest* (Wright & Ziegler, 2017). Other packages such as *randomForestSRC*, *bigrf*, *Random Jungle* or *Rborist* lie in between.

Appendix A.2. Support Vector Machine

Package *kernlab* (Karatzoglou et al., 2004) was chosen in favor of the widely used *e1071* (Meyer et al., 2017) package because *kernlab* offers more kernel

options. Other kernels than RBF have been tested partly but not analyzed in detail in this work.

Appendix A.3. Boosted Regression Trees

For BRT, only one implementation exists in R (to our knowledge) in package *gbm* (Ridgeway, 2017).

Appendix A.4. Generalized Linear/Additive Model

We used the base implementation of GLMs in the *stats* package which belongs to the core packages of R. For GAMs, the *mgcv* package was chosen in favor of *gam* because it provides several optimization methods to find the optimal smoothing degree of each variable and the ability to include random effects within the model. The *mgcv* package lets the user specify different smooth terms and limits for the degree of non-linearity (Wood, 2006). By default, the upper limit of parameter k, which limits the degree of non-linearity, is set to $k - 1$ with k being the number of variables. Note: It is important to ensure that during optimization k does not hit the upper limit in any of the optimized smooth terms of a predictor variable. Otherwise, the degree of non-linearity of a predictor variable would be restricted and cannot be modeled accurately. Subsequently, model performance would not be optimal. Setting k to a high value relative to the final smoothing degree result leads to highly increased run-time or even convergence problems.

Appendix B. 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 B.3: 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 B.4: Summary of nominal predictor variables

Appendix C. Additional hyperparameter tuning results

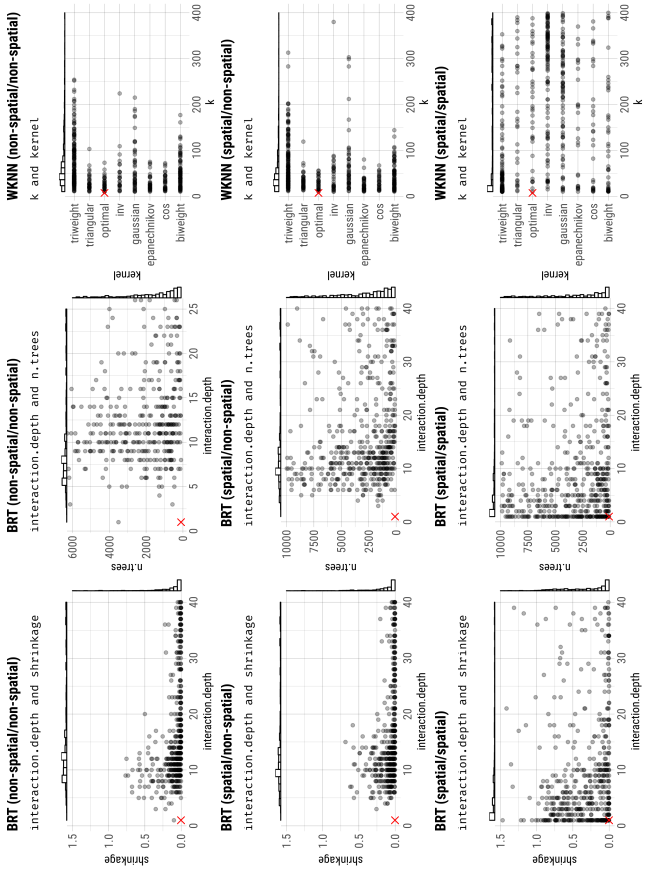

Figure C.7: Best hyperparameter settings by fold (500 total) each estimated from 400 random search tuning iterations per fold using five-fold cross-validation. Split by spatial and non-spatial partitioning setup and model type. Red crosses indicate default hyperparameter values of the respective model. Black dots represent the winning hyperparameter setting out of each random search tuning of the respective fold.

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.

Aertsen, W., Kint, V., van Orshoven, J., Özkan, K., & Muys, B. (2010). Comparison and ranking of different modelling techniques for prediction of site index in mediterranean mountain forests. *Ecological Modelling*, *221*, 1119–1130. URL: https://doi.org/10.1016%2Fj.ecolmodel.2010.01.007. doi:10.1016/j.ecolmodel.2010.01.007.

Bahn, V., & McGill, B. J. (2012). Testing the predictive performance of distribution models. *Oikos*, *122*, 321–331. URL: https://doi.org/10.1111%2Fj.1600-0706.2012.00299.x. doi:10.1111/j.1600-0706.2012.00299.x.

Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *J. Mach. Learn. Res.*, *13*, 281–305. URL: http://dl.acm.org/citation.cfm?id=2188385.2188395.

Biau, G., & Scornet, E. (2016). A random forest guided tour. *TEST*, *25*, 197–227. URL: https://doi.org/10.1007/s11749-016-0481-7. doi:10.1007/s11749-016-0481-7.

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.

Breiman, L. (2001). Random forests. *Machine Learning*, *45*, 5–32. URL: https://doi.org/10.1023%2Fa%3A1010933404324. doi:10.1023/a:1010933404324.

Brenning, A. (2005). Spatial prediction models for landslide hazards: review, comparison and evaluation. *Natural Hazards and Earth System Sci-*

ence, *5*, 853–862. URL: https://doi.org/10.5194%2Fnhess-5-853-2005. doi:10.5194/nhess-5-853-2005.

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. URL: https://doi.org/10.1109%2Figarss.2012.6352393. 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. URL: https://doi.org/10.1002%2Fsim.3310. doi:10.1002/sim.3310.

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. URL: http://www.nat-hazards-earth-syst-sci.net/15/45/2015/.

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*. URL: http://arxiv.org/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. URL: https://doi.org/10.1007%2Fs10346-015-0557-6. doi:10.1007/s10346-015-0557-6.

Candy, J. V., & Breitfeller, E. F. (2013). *Receiver Operating Characteristic (ROC) Curves: An Analysis Tool for Detection Performance*. Technical Report. URL: https://doi.org/10.2172%2F1093414. doi:10.2172/1093414.

Cressie, N. A. C. (1993). *Statistics for Spatial Data*. John Wiley & Sons, Inc. URL: https://doi.org/10.1002%2F9781119115151. doi:10.1002/9781119115151.

Demoulin, A., & Chung, C.-J. F. (2007). Mapping landslide susceptibility from small datasets: A case study in the pays de herve (e belgium). *Geomorphology*, *89*, 391–404. URL: https://doi.org/10.1016%2Fj.geomorph.2007.01.008. doi:10.1016/j.geomorph.2007.01.008.

Díaz-Uriarte, R., & De Andres, S. A. (2006). Gene selection and classification of microarray data using random forest. *BMC bioinformatics*, *7*, 3.

Dormann, C. F. (2007). Effects of incorporating spatial autocorrelation into the analysis of species distribution data. *Global Ecology and Biogeography*, *16*, 129–138. URL: https://doi.org/10.1111%2Fj.1466-8238.2006.00279.x. doi:10.1111/j.1466-8238.2006.00279.x.

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. URL: https://doi.org/10.1111%2Fj.2007.0906-7590.05171.x. doi:10.1111/j.2007.0906-7590.05171.x.

Duan, K., Keerthi, S., & Poo, A. N. (2003). Evaluation of simple performance measures for tuning SVM hyperparameters. *Neurocomputing*, *51*, 41–59. URL: https://doi.org/10.1016%2Fs0925-2312%2802%2900601-x. doi:10.1016/s0925-2312(02)00601-x.

Duarte, E., & Wainer, J. (2017). Empirical comparison of cross-validation and internal metrics for tuning SVM hyperparameters. *Pattern Recognition Letters*, *88*, 6–11. URL: https://doi.org/10.1016%2Fj.patrec.2017.01.007. doi:10.1016/j.patrec.2017.01.007.

Dudani, S. A. (1976). The distance-weighted k-nearest-neighbor rule. *IEEE Transactions on Systems, Man, and Cybernetics*, *SMC-6*, 325–327. URL: https://doi.org/10.1109%2Ftsmc.1976.5408784. doi:10.1109/tsmc.1976.5408784.

Elith, J., Leathwick, J. R., & Hastie, T. (2008). A working guide to boosted regression trees. *Journal of Animal Ecology*, *77*, 802–813. URL: http://dx.doi.org/10.1111/j.1365-2656.2008.01390.x. doi:10.1111/j.1365-2656.2008.01390.x.

European Commission, J. R. C. (2010). *'Map of Soil pH in Europe', Land Resources Management Unit, Institute for Environment & Sustainability*. URL: http://esdac.jrc.ec.europa.eu/content/soil-ph-europe.

Fassnacht, F., Hartig, F., Latifi, H., Berger, C., Hernández, J., Corvalán, P., & Koch, B. (2014). Importance of sample size, data type and prediction method for remote sensing-based estimations of aboveground forest biomass. *Remote Sensing of Environment*, *154*, 102–114. URL: https://doi.org/10.1016%2Fj.rse.2014.07.028. doi:10.1016/j.rse.2014.07.028.

Fix, & Hodges (1951). *Discriminatory analysis, nonparametric discrimination: Consistency properties*. Technical Report U.S. Air Force, School of Aviation Medicine, Randolph Field, TX.

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. URL: https://doi.org/10.1139%2Fx09-131. doi:10.1139/x09-131.

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. URL: 10.1007/s00374-003-0579-4. doi:10.1007/s00374-003-0579-4.

Garofalo, M., Botta, A., & Ventre, G. (2016). Astrophysics and big data: Challenges, methods, and tools. *Proceedings of the International Astronomical Union*, *12*, 345–348. doi:10.1017/S1743921316012813.

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:10.1109/LGRS.2017.2747222.

GeoEuskadi (1999). *Litologia y permeabilidad*. URL: http://www.geo.euskadi.eus/geonetwork/srv/spa/main.home.

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. URL: https://doi.org/10.1007%2F978-3-319-09057-3_160. 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. URL: https://doi.org/10.2307%2F2530946. doi:10.2307/2530946.

Grotzinger, J., & Jordan, T. (2016). Sedimente und sedimentgesteine. In *Press/Siever Allgemeine Geologie* (pp. 113–144). Springer Berlin Heidelberg. URL: https://doi.org/10.1007%2F978-3-662-48342-8_5. doi:10.1007/978-3-662-48342-8_5.

Guo, Q., Kelly, M., & Graham, C. H. (2005). Support vector machines for predicting distribution of sudden oak death in california. *Ecological Modelling*, *182*, 75–90. URL: https://doi.org/10.1016%2Fj.ecolmodel.2004.07.012. doi:10.1016/j.ecolmodel.2004.07.012.

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. URL: https://doi.org/10.1016%2Fj.ecolmodel.2016.02.021. doi:10.1016/j.ecolmodel.2016.02.021.

Hastie, T., Friedman, J., & Tibshirani, R. (2001). *The Elements of Statistical Learning*. Springer New York. URL: https://doi.org/10.1007%2F978-0-387-21606-5. doi:10.1007/978-0-387-21606-5.

Heaton, J. B., Polson, N. G., & Witte, J. H. (2016). Deep learning in finance. *CoRR*, *abs/1602.06561*. URL: http://arxiv.org/abs/1602.06561.

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. URL: https://doi.org/10.1371%2Fjournal.pone.0169748. doi:10.1371/journal.pone.0169748.

Hong, H., Pradhan, B., Jebur, M. N., Bui, D. T., Xu, C., & Akgun, A. (2015). Spatial prediction of landslide hazard at the Luxi area (China) using support vector machines. *Environmental Earth Sciences*, *75*. URL: https://doi.org/10.1007%2Fs12665-015-4866-9. doi:10.1007/s12665-015-4866-9.

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. URL: https://doi.org/10.1007%2F978-3-642-25566-3_40. doi:10.1007/978-3-642-25566-3_40.

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:10.1111/ppa.12328.

Jabot, F. (2015). Why preferring parametric forecasting to nonparametric methods? *Journal of Theoretical Biology*, *372*, 205–210. URL: https://doi.org/10.1016%2Fj.jtbi.2014.07.038. doi:10.1016/j.jtbi.2014.07.038.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An Introduction to Statistical Learning*. Springer New York. URL: https://doi.org/10.1007%2F978-1-4614-7138-7. doi:10.1007/978-1-4614-7138-7.

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. URL: https://doi.org/10.1016%2Fj.ecolmodel.2017.08.017. doi:10.1016/j.ecolmodel.2017.08.017.

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. URL: http://www.jstatsoft.org/v11/i09/. R package version 0.9-25.

Kohavi, R. et al. (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. URL: https://doi.org/10.1007%2F978-1-4614-6849-3. doi:10.1007/978-1-4614-6849-3.

Legendre, P., & Fortin, M. J. (1989). Spatial pattern and ecological analysis. *Vegetatio*, *80*, 107–138. URL: https://doi.org/10.1007%2Fbf00048036. doi:10.1007/bf00048036.

Leung, M. K. K., Delong, A., Alipanahi, B., & Frey, B. J. (2016). Machine learning in genomic medicine: A review of computational problems and data sets. *Proceedings of the IEEE*, *104*, 176–197. doi:10.1109/JPROC.2015.2494198.

Maenner, M. J., Yeargin-Allsopp, M., Van Naarden Braun, K., Christensen, D. L., & Schieve, L. A. (2016). Development of a machine learning algorithm for the surveillance of autism spectrum disorder. *PLOS ONE*, *11*, 1–11. URL: https://doi.org/10.1371/journal.pone.0168224. doi:10.1371/journal.pone.0168224.

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. URL: http://papers.nips.cc/paper/6466-bayesian-optimization-for-automated-model-selection.pdf.

Mets, K. D., Armenteras, D., & Dávalos, L. M. (2017). Spatial autocorrelation reduces model precision and predictive power in deforestation analyses. *Ecosphere*, *8*, e01824. URL: https://doi.org/10.1002%2Fecs2.1824. doi:10.1002/ecs2.1824.

Meyer, D., Dimitriadou, E., Hornik, K., Weingessel, A., & Leisch, F. (2017). e1071: Misc functions of the department of statistics, probability theory group (formerly: E1071), TU Wien, . URL: https://CRAN.R-project.org/package=e1071. R package version 1.6-8.

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. URL: https://doi.org/10.1016%2Fj.envsoft.2017.12.001. doi:10.1016/j.envsoft.2017.12.001.

Muenchow, J., Feilhauer, H., Bräuning, A., Rodríguez, E. F., Bayer, F., Rodríguez, R. A., & Wehrden, H. (2013a). 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. URL: http://onlinelibrary.wiley.com/doi/10.1111/jvs.12038/full.

Muenchow, J., Hauenstein, S., Bräuning, A., Bäumler, R., Rodríguez, E. F., & von Wehrden, H. (2013b). Soil texture and altitude, respectively, widely determine the floristic gradient of the most diverse fog oasis in the peruvian desert. *Journal of Tropical Ecology*, *29*, 427–438. doi:10.1017/S0266467413000436.

Múgica, J. R. M., Murillo, J. A., Ikazuriaga, I. A., Peña, B. 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.

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.

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. URL: https://doi.org/10.1016%2Fj.rse.2015.10.029. doi:10.1016/j.rse.2015.10.029.

Perretti, C. T., & Munch, S. B. (2015). On estimating the reliability of ecological forecasts. *Journal of Theoretical Biology*, *372*, 211–216. URL: https://doi.org/10.1016%2Fj.jtbi.2015.02.031. doi:10.1016/j.jtbi.2015.02.031.

Pourghasemi, H. R., & Rahmati, O. (2018). Prediction of the landslide susceptibility: Which algorithm, which precision? *CATENA*, *162*, 177–192. URL: https://doi.org/10.1016%2Fj.catena.2017.11.022. doi:10.1016/j.catena.2017.11.022.

Pradhan, B. (2013). A comparative study on the predictive ability of the decision tree, support vector machine and neuro-fuzzy models in landslide susceptibility mapping using GIS. *Computers & Geosciences*, *51*, 350–365. URL: https://doi.org/10.1016%2Fj.cageo.2012.08.023. doi:10.1016/j.cageo.2012.08.023.

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*, . URL: https://doi.org/10.1111%2Fjav.01238. doi:10.1111/jav.01238.

R Core Team (2017). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing Vienna, Austria. URL: https://www.R-project.org/ R version 3.3.3.

Richter, J. (2017). *mlrHyperopt: Easy Hyperparameteroptimization with mlr and mlrMBO*. URL: http://doi.org/10.5281/zenodo.896269 R package version 0.1.1.

Ridgeway, G. (2017). *gbm: Generalized Boosted Regression Models*. URL: https://CRAN.R-project.org/package=gbm 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. URL: https://doi.org/10.1111%2Fecog.02881. doi:10.1111/ecog.02881.

Ruß, G., & Brenning, A. (2010). Spatial variable importance assessment for yield prediction in precision agriculture. In *Lecture Notes in Computer Science* (pp. 184–195). Springer Berlin Heidelberg. URL: https://doi.org/10.1007%2F978-3-642-13062-5_18. doi:10.1007/978-3-642-13062-5_18.

Ruß, G., & Kruse, R. (2010). Regression models for spatial data: An example from precision agriculture. In *Advances in Data Mining. Applications and Theoretical Aspects* (pp. 450–463). Springer Berlin Heidelberg. URL: https://doi.org/10.1007%2F978-3-642-14400-4_35. doi:10.1007/978-3-642-14400-4_35.

Schernthanner, H., Asche, H., Gonschorek, J., & Scheele, L. (2017). Spatial modeling and geovisualization of rental prices for real estate portals. *International Journal of Agricultural and Environmental Information Systems*, *8*, 78–91. URL: https://doi.org/10.4018%2Fijaeis.2017040106. doi:10.4018/ijaeis.2017040106.

Schliep, K., & Hechenbichler, K. (2016). *kknn: Weighted k-Nearest Neighbors*. URL: https://CRAN.R-project.org/package=kknn 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*. Master's thesis Friedrich-Schiller-University Jena. doi:https://doi.org/10.5281/zenodo.814262 (unpublished).

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). URL: https://doi.org/10.1093%2Ficesjms%2Ffsw136. doi:10.1093/icesjms/fsw136.

Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, *45*, 427–437. URL: https://doi.org/10.1016%2Fj.ipm.2009.03.002. doi:10.1016/j.ipm.2009.03.002.

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. URL: https://doi.org/10.1016%2Fj.geomorph.2016.03.015. doi:10.1016/j.geomorph.2016.03.015.

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. URL: https://doi.org/10.1002%2F9781118793787.ch3. 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. URL: https://doi.org/10.1016%2Fj.quascirev.2005.05.001. doi:10.1016/j.quascirev.2005.05.001.

Telford, R., & Birks, H. (2009). Evaluation of transfer functions in spatially structured environments. *Quaternary Science Reviews*, *28*, 1309–1316. URL: https://doi.org/10.1016%2Fj.quascirev.2008.12.020. doi:10.1016/j.quascirev.2008.12.020.

Vapnik, V. (1998). The support vector method of function estimation. In *Nonlinear Modeling* (pp. 55–85). Springer US. URL: https://doi.org/10.1007%2F978-1-4615-5703-6_3. 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. URL: https://doi.org/10.1016%2Fj.ecolmodel.2011.12.007. doi:10.1016/j.ecolmodel.2011.12.007.

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.

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. URL: https://doi.org/10.1111%2Fj.2041-210x.2011.00170.x. doi:10.1111/j.2041-210x.2011.00170.x.

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. URL: https://doi.org/10.1016%2Fj.ecolmodel.2017.02.029. doi:10.1016/j.ecolmodel.2017.02.029.

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. URL: https://doi.org/10.1071%2Fap08036. doi:10.1071/ap08036.

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. URL: https://doi.org/10.1111%2Fj.1365-2699.2008.01965.x. doi:10.1111/j.1365-2699.2008.01965.x.

Wood, S. (2006). *Generalized Additive Models: An Introduction with R*. Chapman and Hall/CRC.

Working Group WRB, I. (2015). *World Reference Base for Soil Resources 2014, update 2015 International soil classification system for naming soils and creating legends for soil maps. World Soil Resources Reports No. 106. FAO, Rome*.

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:10.18637/jss.v077.i01.

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. URL: https://doi.org/10.1007%2Fs10346-015-0667-1. doi:10.1007/s10346-015-0667-1.

Zuur, A. F., Ieno, E. N., Walker, N., Saveliev, A. A., & Smith, G. M. (2009). *Mixed effects models and extensions in ecology with R*. Springer New York. URL: https://doi.org/10.1007%2F978-0-387-87458-6. doi:10.1007/978-0-387-87458-6.