FRIEDRICH-SCHILLER-
UNIVERSITÄT
JENA

Benchmarking environmental machine-learning models: Methodological progress and an application to forest health

Dissertation
(kumulativ)

zur Erlangung des akademischen Grades
Doctor rerum naturalium (Dr. rer. nat.)

vorgelegt dem Rat der Chemisch-Geowissenschaftlichen Fakultät
der Friedrich-Schiller-Universität Jena

von **Patrick Schratz**, M.Sc.
geboren am 22.08.1991 in Lichtenfels, Deutschland

Dissertation, Friedrich-Schiller-Universität Jena, 2022

Gutachter:

1. Prof. Dr. Alexander Brenning, Universität Jena
2. Prof. Dr. Hanna Meyer, Universität Münster

Tag der Verteidigung: 30.11.2022

DECLARATIONS

Selbstständigkeitserklärung

Ich erkläre, dass ich die vorliegende Arbeit selbständig und unter Verwendung der angegebenen Hilfsmittel, persönlichen Mitteilungen und Quellen angefertigt habe.

Ort, Datum Unterschrift des Verfassers

Erklärung zu den Eigenanteilen

Für alle in dieser kumulativen Dissertation verwendeten Manuskripte liegen die notwendigen Genehmigungen der Verlage ("Reprint permissions") für die Zweitpublikation vor.

Die Co-Autorinnen/-Autoren der in dieser kumulativen Dissertation verwendeten Manuskripte sind sowohl über die Nutzung, als auch über die angegebenen Eigenanteile der weiteren Doktorandin- nen/-Doktoranden als Co-Autorinnen/-Autoren an den Publikationen und Zweitpublikationsrechten bei einer kumulativen Dissertation informiert und stimmen dem zu.

Die Anteile des Promovenden sowie der weiteren Doktorandinnen/Doktoranden als Co-Autorinnen/Co-Autoren an den Publikationen und Zweitpublikationsrechten bei einer kumulativen Dissertation sind in der Anlage aufgeführt.

Name des Promovenden	Datum	Ort	Unterschrift
Patrick Schratz			

ACKNOWLEDGEMENTS

Many people have supported me throughout the last five and a half years, who I would like to thank at this point.

First and foremost, I'd like to thank Prof. Dr. Alexander Brenning for all of his support, kindness and knowledge sharing throughout this dissertation and my previous studies. Alex, without your help and motivation I might have never discovered the joy in data analysis and the `R` programming language and may eventually never have pursued that path. I have learned many skills throughout these years discussing and interacting with you, also on the social side. Thanks for supporting me unconditionally, even through the complicated times when I was struggling and after deciding to finish the thesis outside of academia. Thank you!

Mama, Papa - danke für eure Unterstützung während meiner akademischen Abenteuer. Die Wissenschaft ist nicht eure Welt und ihr müsst euch unzählige Male gefragt haben, was ich da eigentlich den ganzen Tag lang mache - und ob das Alles wirklich jemals ein Ende findet. Ich möchte mich bei euch für die Herzenswärme während meiner Besuche Zuhause bedanken und euer positives Mitfiebern und Nachfragen über den aktuellen Stand. Ich werde mich nun auch nicht mehr beschweren wenn ihr erzählt, dass es "bald einen Doktor in der Familie gibt" ;)

I'd also like to thanks the cynkra team, especially Christoph and Kirill, who granted me the time needed to finish this work. I never felt stressed or pressured in a negative way, even when it became clearer that everything might take a little longer than initially expected. Special thanks to Christoph for the countless talks we had during the last phase which helped me to stay focused and continue.

Eugenia, I'd also like to thank you explicitly for all the help and support you've provided on the biological side of this thesis. Your hospitality during my visits in the Basque Country was phenomenal and it was inspiring to see how pathogens are treated and researched in biological practice.

Special thanks also goes out to the Goetz family for supporting me throughout this journey and making the time in Jena enjoyable at all times. Thanks for your emotional support, you've become really good friends.

Thanks to all the people who have proofread my works throughout all these years and provided thoughtful high-level feedback: Bernd, Jannes, Jakob, José, Marc, Marie, Matthias, Michel.

Last, I would like to say thanks to all people who have supported me in any other way, by talking about specific details, discussing the bigger picture of environmental modeling, sharing their own (PhD) experiences or by simply letting me forget about this world for a moment by creating special moments. In alphabetical order by first name: Andreas Schäf, Anna Berninger, Annika Künne, Axel Schmidt, Bernd Bischl, Bettina Böhm, Christian Berger, Christoph Niemann, Codrina Ilie, Daniela Schüpbach, Eugenia

Iturritxa, Fabian Schäfer, Felix Cremer, Helene Goetz, Jannes Münchow, Jason Goetz, Johannes Balling, John Truckenbrodt, José Cortés, Kai Heckel, Laura Krohn, Manfred Fink, Marc Becker, Marcel Urban, Marco Pena, Marie Schneider, Michel Lang, Raphael Knevels, Stefan Zimmermann, Sven Kralisch and Tina Trautmann. I have very likely forgotten to mention certain people at this point - apologies for that and please assume yourself to be part of the list ;)

This research was partly funded by the EU LIFE programme (LIFE14 ENV/ES/000179), the Carl Zeiss foundation and the Friedrich Schiller University Jena.

Patrick Schratz
Zurich, June 2022.

ABSTRACT

Benchmarking environmental machine-learning models: Methodological progress and an application to forest health

Geospatial machine learning is a versatile approach to analyze environmental data and can help to better understand the interactions and current state of our environment. Due to the artificial intelligence of these algorithms, complex relationships can possibly be discovered which might be missed by other analysis methods. Modeling the interaction of creatures with their environment is referred to as ecological modeling, which is a subcategory of environmental modeling. A subfield of ecological modeling is species distribution modeling (SDM), which aims to understand the relation between the presence or absence of certain species in their environments.

SDM is different from classical mapping/detection analysis. While the latter primarily aim for a visual representation of a species spatial distribution, the former focuses on using the available data to build models and interpreting these. The algorithms, optimization, interpretation and validation approaches that can be used therefor are manifold. Because no single best option exists and well-performing ones differ for each analysis, different settings need to be evaluated and compared against each other.

When conducting such modeling comparisons, which are commonly referred to as benchmarking, care needs to be taken throughout the analysis steps to achieve meaningful and unbiased results. These steps are composed out of data preprocessing, model optimization and performance assessment. While these general principles apply to any modeling analysis, their application in an environmental context often requires additional care with respect to data handling, possibly hidden underlying data effects and model selection. To conduct all in a programmatic (and efficient) way, toolboxes in the form of programming modules or packages are needed. This work makes methodological contributions which focus on efficient, machine-learning based analysis of environmental data. In addition, research software to generalize and simplify the described process has been created throughout this work.

The described methodological techniques were applied in studies analyzing the presence and effects of forest pathogens in northern Spain. Forest pathogens have the potential to cause severe damages to large forest areas all around the world in a relatively short time. Monoculture tree plots which are infected by (non-native) pathogens often suffer from a partial or complete die back in a relatively short time. *Diplodia sapinea* and *Fusarium circinatum* are two invasive pathogens which have seen an increased spread in southern Europe within the last decade and caused severe economical and environmental damage. To better understand the preferred living conditions and the current spread of these fungi, a combination of biological expertise, geographic information system (GIS) techniques and statistical modeling can be used. Statistical and machine learning modeling comes with the promising ability to reveal patterns within data and provides the option to make predictions into areas which are difficult to access. Additionally, remote sensing products can aid pathogen distribution analysis by facilitating the derivation of environmental variables for large areas which can be used as explanatory variables during modeling. Especially hyperspectral remote sensing data inherits high potential for the use in forest health monitoring analyses due

to the spectral sensitivity of its narrow bands, which are known to help in detecting vegetation-related changes.

This work puts a strong focus on reproducible workflow execution. All steps were executed in a programmatic way and made available publicly. The creation and generalization of (public) research software can help to execute large benchmark experiments in a controlled and reproducible way. It allows for public quality checks and adaptation or reuse by other researchers in future studies.

The first study of this work investigates differences in predictive performance when applying various spatial and non-spatial partitioning methods during cross-validation (CV)-based model evaluation. A special focus is put on evaluating the effects of different spatial and non-spatial resampling techniques during hyperparameter tuning. *Diplodia sapinea* infections of trees across the Basque Country are modeled using environmental variables. The learning algorithms boosted regression trees (BRT), weighted *k*-nearest neighbor (KNN), support vector machine (SVM), random forest (RF), generalized linear model (GLM) and generalized additive model (GAM) are compared using a model-based optimization approach. A large difference in predictive performance between models using spatial and non-spatial partitioning was found, caused by the underlying spatial autocorrelation in the data. Spatial autocorrelation often results in highly similar training and test sets when using non-spatial resampling splits. Spatial resampling in contrast creates partitions which are less affected by spatial autocorrelation due to a larger spatial distance between training and test sets. Results showed that using spatial partitioning during the model optimization stage of a nested CV setting does not result in reduced model performance compared to the use of non-spatial methods. While no substantial improvements were found either, some models showed slightly improved performances when being tuned on spatially distant training and test sets. The results indicate that spatial resampling methods should be used for both performance estimation and hyperparameter optimization in CV settings when analyzing spatial data.

The second case study analyzes the effectiveness of (ensemble) filter methods on high-dimensional datasets as an alternative to wrapper feature selection. The percentage defoliation of trees caused by *Fusarium circinatum* infection across four plots in the Basque Country is modeled using hyperspectral remote sensing data. For the analysis, a benchmark matrix of 156 different settings is created composed out of the learning algorithms RF, extreme gradient boosting (XGBoost), SVM, and penalized regression (L1 and L2), multiple diverse feature sets and various filter methods. The goal is to detect which combination of learner, feature set and filter method is able to model the high-dimensional, high-collinear data best. Ensemble filters, on average, did not improve predictive performance compared to single filters. The overall effectiveness of filters were dependent on the learner and feature set combination. The fact that many algorithms favored different filters highlights the need to test various combinations of algorithms, feature and hyperparameter optimization methods to find the best possible setting for the data at hand. Promising predictive performance increases were found for some filters which underlines the general potential of filters, which is further extended by their computational advantageous application in contrast to wrapper feature selection. Yet the question of whether filter methods are generally preferable over other feature selection approaches remained unresolved. Besides the analysis of filter methods, variable importance for the hyperspectral bands and vegetation indices feature sets were calculated. Using permutation-based variable importance, the highest mean decrease in RMSE across permutations was found for features around the "red edge" of the spectrum (around 680 nm - 720 nm). This confirms the importance of this spectral region for vegetation analysis. However, permutation-based variable importance might be biased in this setting due to the high correlation among variables. Other approaches such

as grouped permutation-based variable importance, accumulated local effects (ALE) plots or shapley additive explanations (SHAP) are less affected by high correlation between variables and might result in less biased estimates.

The third part of this thesis introduces the R package `mlr3spatiotempcv` which provides an interface to various spatiotemporal resampling methods within the `mlr3` machine learning framework. The package aims to simplify the application of spatiotemporal resampling methods in combination with any learning algorithm, hyperparameter optimization technique or feature selection method. `mlr3spatiotempcv` also includes spatiotemporal visualization functionality for partitioning methods which can help to better understand the created data splits and even allows highlighting omitted data points by specific resampling methods.

Emphasis is put on scientific reproducibility and the creation and maintenance of research software. Best practices with respect to reproducible workflow management are applied and discussed in greater detail. The R packages `drake` and `renv` are used during all studies of this work to ensure reproducibility and simplified study recreation. All resources (code, data and metadata) have been made available in individual public research compendia using the open-access data hoster Zenodo.

The predictive performance estimates achieved in both pathogen case studies were considered as fair. The small number of observations might have had a negative influence on the generalization capabilities of the fitted models. Additionally, artifacts in the in situ and remote sensing data might have affected model performances in a negative way. The collection of more in situ data and the use of additional variables might help to fit models which are able to achieve better performance estimates in similar studies in the future.

ZUSAMMENFASSUNG

Raumbezogenes, maschinelles Lernen ist ein vielseitiger Ansatz zum Modellieren von Umweltdaten und kann dabei behilflich sein, die Interaktionen und den aktuellen Zustand unserer Umwelt besser zu verstanden. Mit Hilfe der künstlichen Intelligenz dieser Lernalgorithmen können komplexe Beziehungen entdeckt werden, welche bei der Verwendung anderen Analysemethoden möglicherweise unentdeckt geblieben wären. Die Modellierung der Interaktionen von Lebewesen mit ihrer natürlichen Umgebung wird ökologische Modellierung genannt, eine Unterkategorie der Umweltmodellierung. Wiederum eine Unterkategorie der ökologischen Modellierung ist die Artverbreitungsmodellierung, welche darauf abzielt, die Beziehung zwischen der Präsenz oder Absenz bestimmter Spezies in einem bestimmten Lebensraum zu verstehen.

Die Artverbreitungsmodellierung hebt sich von reiner Kartierungsmodellierung ab. Während Letztere primär auf eine visuelle Representation der räumlichen Artenverbreitung abzielt, versucht die Artverbreitungsmodellierung mit Hilfe der vorliegenden Daten Modelle anzupassen und diese zu interpretieren. Die Algorithmen, Optimierungsmethoden, Interpretation - und Validierungsansätze, welche hierfür verwendet werden können, sind vielfältig. Da es keine Universallösung bei der Auswahl dieser Methoden gibt und die gut funktionierenden Kombinationen sich in den meisten Analysen unterscheiden, müssen in jeder Analyse verschiedene Kombinationen gegeneinander evaluiert werden.

Bei der Durchführung solcher Modellierungsvergleiche, welche auch als "benchmarking" bezeichnet werden, muss innerhalb der einzelnen Teilschritte darauf geachtet werden, dass aussagekräftige und unverzerrte Ergebnisse erzielt werden. Diese Analyseschritte setzen sich aus der Datenvorprozessierung, der Modelloptimierung und der Modellvalidierung zusammen. Wenngleich diese Prinzipien auf jede Modellierungsanalyse zutreffen, so benötigt deren Anwendung im Kontext der Umweltmodellierung oftmals zusätzliche Aufmerksamkeit in Bezug auf Datenverarbeitung, möglichen versteckten Effekten in den Daten und Modellselektion. Um all diese Analyseschritte in einer programmatischen (und effizienten) Weise auszuführen, sind Werkzeugkästen in der Form von Softwaremodulen oder Paketen notwendig. Diese Dissertation liefert methodische Beiträge in diese Richtung mit Fokus auf der effizienten Analyse von Umweltdaten unter Zuhilfenahme maschineller Lernmethoden. Zusätzlich wurde im Zuge dieser Arbeit Forschungssoftware erstellt mit dem Ziel, die beschriebenen Prozesse zu generalisieren und vereinfachen.

Die erwähnten methodischen Techniken wurden in Studien angewendet welche die Präsenz und Auswirkungen von auf Bäumen spezialisierten Krankheitserregern im Norden Spaniens untersucht haben. Diese spezialisierten Erreger haben das Potential weltweit in relativ kurzer Zeit starke und großflächige Schäden in Waldgebieten zu verursachen. Monokulturelle Baumparzellen, welche von solchen (nicht einheimischen) Erregern befallen werden, sehen sich oftmals mit dem teilweisen oder sogar kompletten Absterben der einzelnen Bäume konfrontiert. *Diplodia sapinea* und *Fusarium circinatum* sind zwei solcher invasiven Krankheitserreger welche sich in der letzten Dekade stark innerhalb Südeuropas ausgebreitet haben und starke ökonomische und ökologische Schäden nach sich gezogen haben. Um deren bevorzugte Lebensumstände und aktuelle Verbreitung besser zu verstehen, kann eine Kombination

aus biologischer Expertise, geographischen Informationssystemen und statistischer Modellierung genutzt werden. Statistische und maschinelle Lernmethoden haben die vielversprechende Eigenschaft Muster in Daten erkennen zu können und Vorhersagen in Gebiete machen zu können, welche für den Mensch nur schwierig zu erreichen sind. Zusätzlich können Fernerkundungsprodukte durch die Bereitstellung von Umweltvariablen für großflächige Gebiete die Verteilungsanalyse solcher Krankheitserreger unterstützen. Speziell hyperspektrale Fernerkundungsdaten beherbergen großes Potential für die Nutzung in der Waldgesundheitsüberwachung durch die spektrale Sensitivität ihrer schmalen Bänder, welche dafür bekannt sind vegetationsbedingte Änderungen erkennbar zu machen.

Diese Arbeit legt einen starken Fokus auf eine reproduzierbare Ausführung des Arbeitsablaufs. Alle Einzelschritte wurden in einer programmatischen Weise ausgeführt und öffentlich zugänglich gemacht. Die Erstellung und Generalisierung von (öffentlich verfügbarer) wissenschaftlicher Software kann dabei helfen, große Benchmarkexperimente in einer kontrollierten und reproduzierbaren Weise durchzuführen. Dieses Vorgehen erlaubt öffentliche Qualitätstests und die potentielle Adaption oder Weiterverwendung durch andere Wissenschaftler in zukünftigen Studien.

Die erste Studie dieser Arbeit untersucht die Unterschiede der Vorhersagegüte bei der Nutzung verschiedener räumlicher und nicht-räumlicher Partitionierungsmethoden innerhalb einer auf Kreuzvalidierung basierenden Modellevaluation. Ein besonderer Fokus liegt hierbei auf der Auswertung der Unterschiede welche durch die Nutzung der verschiedenen Partitionierungsmethoden während der Hyperparameteroptimierung entstehen. Infektionen mit *Diplodia sapinea* an Bäumen im Baskenland werden hierfür mit Hilfe von Umweltvariablen modelliert. Die Lernalgorithmen boosted regression trees (BRT), weighted k-nearest neighbor (KNN), support vector machine (SVM), random forest (RF), generalized linear model (GLM) und generalized additive model (GAM) werden hierfür mittels eines modelgestützten Optimierungsverfahren verglichen. Bedingt durch die unterliegende räumliche Autokorrelation in den vorliegenden Daten wurde ein grosser Unterschied in der Vorhersagekraft zwischen Modellen gefunden, welche entweder eine räumliche oder nicht-räumliche Partitionierungsmethode verwendeten. Räumliche Autokorrelation resultiert oft in sehr ähnlichen Lern- und Evaluierungssätzen, wenn ein nicht-räumliches Partitioningsverfahren verwendet wird. Im Gegensatz dazu resultiert die Verwendung einer räumlichen Wiederholungsprobennnahme in Partitionen, die weniger stark von räumlicher Autokorrelation betroffen sind, aufgrund einer größeren räumlichen Distanz zwischen den Lern-und Evaluierungssätzen. Die Resultate zeigen, dass die Nutzung räumlicher Partitionierungsmethoden während der Modelloptimierung einer verschachtelten Modellevaluation die Modellperformanz nicht reduziert, im Vergleich gegenüber einer nicht-räumlichen Partitionierungsmethode. Wenngleich auch keine substantiellen Verbesserungen gefunden wurden, zeigten einige Modelle eine leicht verbesserte Leistung, wenn diese auf räumlich distanzierten Lern- und Evaluierungssätzen optimiert wurden.

Die zweite Fallstudie analysiert die Effektivität von Ensemble-Filtermethoden als Alternative zu Wrapper-Variablenselektion auf hochdimensionalen Datensätzen. Hierfür wird die prozentuale Entlaubung von Bäumen, hervorgerufen durch eine Infektion mit *Fusarium circinatum*, mit Hilfe von hyperspektralen Fernerkundungsdaten im Baskenland modelliert. Eine Vergleichsmatrix mit 156 verschiedenen Einstellungen, welche sich aus den Lernalgorithmen RF, XGBoost, SVM und regularisierter Regression (L1 und L2), mehreren diversen Variablensätzen und unterschiedlichen Filtermethoden zusammensetzt, dient als Grundlage der Studie. Das Ziel ist zu erkennen, welche Kombination aus Lernmethode, Variablensatz und Filtermethode die hochdimensionalen, hochkollinearen Daten am besten verarbeiten kann. Im Mittel konnte die Nutzung von Ensemble-Filtermethoden, verglichen mit einfachen Filtermethoden,

die Vorhersagegüte nicht verbessern. Die allgemeine Effektivität von Filtermethoden war abhängig von der jeweiligen Kombination von Lern-und Filtermethode. Der Umstand, dass die meisten Algorithmen unterschiedliche Filtermethoden bevorzugten, hebt die Notwendigkeit des Testens verschiedenster Kombinationen von Lernalgorithmen, Variablen-und Hyperparameteroptimierungsmethoden hervor um die bestmögliche Einstellung für die zugrundeliegenden Daten zu finden. Vielversprechende Verbesserungen bei der Vorhersagegüte wurden bei der Anwendung einiger Filtermethoden gefunden. Dieser Fund unterstreicht das generelle Potential von Filtermethoden in solchen Anwendungen. Dies wird weiterhin durch deren rechentechnisch günstige Anwendung, im Vergleich zu Wrapper-Variablenselektion, unterstützt. Jedoch blieb die Frage, ob die Nutzung von Filtermethoden gegenüber anderen Variablenselektionsmethoden allgemein präferiert werden sollten, ungelöst. Neben der Analyse von Filtermethoden wurde auch die Variablenwichtigkeit der hyperspektralen Bänder und Vegetationsindizes berechnet. Unter Zuhilfenahme einer permutationsbasierten Variablenwichtigkeitsanalyse wurde der größte Abfall der Vorhersagekraft (gemessen anhand der Wurzel der mittleren Fehlerquadratsumme (RMSE)) für Variablen rund um die "red edge" des Spektrums (zwischen 680 nm - 700 nm) ausgemacht. Dieses Ergebnis bestätigt die Wichtigkeit dieser spektralen Region für Vegetationsnanalysen allgemein. Jedoch könnte das Ergebnis dieser Variablenwichtigkeitsanalyse aufgrund der hohen Korrelation einiger Variablen verzerrt sein. Andere Ansätze, wie zum Beispiel gruppierte permutationsbasierte Variablenwichtigkeitsanalyse, ALE plots oder SHAP, sind weniger stark von hoher Variablenkorrelation betroffen und könnten eventuell weniger stark verzerrte Ergebnisse erzielen.

Der dritte Teil dieser Arbeit stellt das `R` Paket `mlr3spatiotempcv` vor, welches eine Schnittstelle zu verschiedenen raumzeitlichen Partitionierungsmethoden innerhalb des `mlr3` Ökosystems bereitstellt. Ziel des Paketes ist es, die Anwendung von raumzeitlichen Stichprobenwiederholungen in Kombination mit Lernalgorithmen und Optimierungsmethoden für Hyperparameter- und/oder Variablenselektion in `R` zu vereinfachen. Weiterhin enthält `mlr3spatiotempcv` Funktionen für Visualisierungen von raumzeitlichen Partitionierungsmethoden, welche dabei helfen können, die Aufteilung des Datensatzes innerhalb der Stichprobenwiederholungen besser zu verstehen. Diese Funktionalität erlaubt es auch Datenpunkte, welche in den jeweiligen Aufteilungen weder für den Lern- noch Testsatz verwendet wurden gesondert hervorzuheben, soweit dies von der jeweils gewählten Methode unterstützt wird.

Diese Arbeit betont die Wichtigkeit wissenschaftlicher Reproduzierbarkeit sowie die Erstellung und Wartung von Forschungssoftware. Bewährte Praktiken in Bezug auf einen reproduzierbaren Arbeitsablauf werden angewandt und diskutiert. Die `R` Pakete `drake` und `renv` wurden in allen Studien dieser Arbeit verwendet um wissenschaftliche Reproduzierbarkeit zu gewährleisten und die exakte Nachprüfung der einzelnen Studien zu vereinfachen. Alle Quellen (Code, Daten, Metadaten) wurden in einem öffentlich zugänglichen Forschungskompendium auf der Platform Zenodo verfügbar gemacht.

Die Vorhersagegüte, welche in den beiden Fallstudien erreicht wurde, kann allgemein als ordentlich bezeichnet werden. Die geringe Anzahl an Beobachtungen hatte mutmaßlich einen negativen Effekt auf die Generalisierungseigenschaften der Modelle. Weiterhin haben möglicherweise Artefakte in den lokalen sowie in den Fernerkundungsdaten die Modellperformanz beeinträchtigt. Die Erhebung weiterer lokaler Beobachtungen und die Nutzung zusätzlicher Variablen könnte dabei helfen, Modelle zu erstellen, welche in zukünftigen ähnlichen Studien in der Lage sind, bessere Vorhersagegüten erzielen zu können.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

ALE	accumulated local effects
ALS	airborne laser scanning
AUROC	area under the receiver operating characteristics curve
BRT	boosted regression trees
CNN	convolutional neural networks
FAIR	findability, accessibility, interoperability, and reusability
CV	cross-validation
DAP	digital aerial photogrammetry
EFSA	European Food Safety Authority
EPPO	European and mediterranean plant protection organization
FFS	forward feature selection
FOSS	free and open source software
GUI	graphical user interface
FS	feature selection
GAM	generalized additive model
GCV	generalized cross-validation
GIS	geographic information system
GLM	generalized linear model
HR	hyperspectral reflectances
ICGC	Institut Cartografic i Geologic de Catalunya
KNN	weighted k-nearest neighbor
LiDAR	light detection and ranging
LIME	local surrogate models
LOOCV	leave-one-out cross-validation
MBO	model-based optimization
ML	machine learning
NDMI	normalized difference moisture index
NRI	normalized ratio index
OLS	ordinary least squares
OMNBR	optimized multiple narrow-band reflectance
PC	principal component
PCA	principal component analysis
PDP	partial dependence plots
PISR	potential incoming solar radiation
PLS	partial least squares
PPC	pine pitch canker
RF	random forest
RMSE	root mean square error
SAC	spatial autocorrelation
SCV	spatial cross-validation
SHAP	shapley additive explanations
SDM	species distribution modeling
SMBO	sequential model-based optimization
SVM	support vector machine
VI	vegetation index
XGBoost	extreme gradient boosting

CHAPTER 1

INTRODUCTION

1.1 Motivation

Environmental modeling is a very general term comprising pretty much any modeling activity that, to some degree, involves environmental components. As such it covers modeling tasks in different scientific fields like geo, bio, or social sciences. A modeling subcategory which shares components from both the geosciences and biosciences is ecological modeling. This field primarily focuses on creating mathematical representations of ecosystems through modeling, i.e., how living organisms interact with their environment (Hall & Day, 1977). This includes the description and analysis of (animal) populations and/or their interaction within a given biome. A subfield of such is SDM which focuses on the understanding and description of certain species, for example manmmals, pathogens, fishes or any other type of living being. The living conditions of all creatures are affected by climate-change related phenomena like sea level rise, glacier melting, temperature rise or destructive storms (IPCC, 2013), which has increased the importance of this research field throughout the last decades. Analyzing the presence and interactions of animals within their environment allows to eventually improve the understanding of the ever-changing ecosystem processes on earth. Such undertakings become even more important in periods of rapid climate change to understand their effects on various species. Furthermore, it can help to early-detect changes in behavioral patterns and apply appropriate guarding measures right in time.

One species type which is known to be sensitive to temperature changes are forest pathogens (Ayres & Lombardero, 2000). While many were historically only present in warmer climates like Australia or southern America, global warming transformed the (sub-)Mediterranean or western Europe into new suitable living areas (La Porta et al., 2008). The appearance of pathogens in new areas is often followed by severe damages to existing forest stands as local species have not yet developed resistance strategies against these (Loo, 2009). With respect to the Mediterranean region, timber wood production has been a major economic mainstay for decades. Due to widespread coverage of many monoculture plots in this area, the possible spread of (new) pathogens has the potential to cause severe environmental and economic damages across large regions in a relatively short time (Iturritxa et al., 2014).

Attempts to lower the prevalence of pathogens often start with the mapping of the current state of forest health and detection of possible dissemination areas (Sandino et al., 2018). For this task the use of remote sensing data can be of great use (A. R. Huete, 2012). The quality and availability of this data type increased within the last decade, especially with the launch of the Sentinel satellites in 2014 (Showstack, 2014). This development has made the use of this type of data in forest health mapping application more attractive. To use remote sensing data in pathogen dissemination analyses, the containing environmental information needs to be linked with pathogen presence. One way how this can be done is by making use of modeling algorithms. These can help to better understand existing relationships and predict the learned interactions into areas which were not yet populated by pathogens. This technique is known to be especially helpful in remote, large or sparsely populated areas (J. Meng et al., 2016).

The usability of machine learning algorithms have greatly improved over the last decade. The increase

in computational resources and efficiency within this time period has helped to cope with the increasing amount of (remote sensing) data, making it possible to process and analyze more data in less time while being able to compare different algorithms and optimization strategies (Y.-T. Tsai, 2018). When performing any kind of modeling, the thorough usage of all available building blocks (optimization, feature selection, validation) is key for producing meaningful and unbiased results (Lantz, 2019; Probst, Boulesteix, & Bischl, 2019). Modeling spatial data requires additional care as the presence of spatiotemporal effects within the data, e.g., spatiotemporal autocorrelation, can eventually lead to unwanted bias in the results. This, combined with the correct handling of non-tabular input data sources, for instance as remote sensing imagery, makes environmental modeling a challenging task. Advancing in the (bias-reduced) application of machine learning algorithms with the combined use of remote sensing data is one of the main research focuses of this work.

To execute suchlike analyses efficiently, a programmatic approach is beneficial in many ways. This work makes heavy use of the programming language R, which is one of the most widely used languages for data analysis and modeling. It's use allows for the creation of research software which can provide a comprehensible and reusable framework to conduct case studies and research in general (Osimo & Switters, 2019). Generalized software solutions, e.g., in the form of R packages, (i) simplify execution, (ii) improve implementation correctness and (iii) allow adaptation by others (Brett et al., 2017). Yet often this part of the analysis is treated as an inadvertent side product rather than an essential part of the scientific workflow (Brett et al., 2017). In addition to the machine-learning driven analysis of forest pathogens, this work also aims to emphasize the importance of research software within the scientific workflow.

1.2 Research objectives

The goal of this thesis is to contribute to the methodological progress in the field of SDM by modeling the effects of pathogen infections on trees in northern Spain. Various learning algorithms are benchmarked with a focus on methodological aspects such as model and hyperparameter optimization, feature selection and bias-reduced performance estimation. All analyses follow reproducibility best practices including the public sharing of code and data. The following research questions are addressed:

1. **How does the use of different (spatial) partitioning methods for model optimization and performance evaluation during CV affect the results of modeling *D. sapinea* infections at pine trees in northern Spain?**

   Spatial k-fold CV has become more popular to assess model performance in recent years. Yet the application of spatial partitioning methods during the hyperparameter optimization step within a nested CV setting has not yet been evaluated in greater detail. On the use case of modeling *D. sapinea*-affected trees in northern Spain, the magnitude of performance difference between spatial and non-spatial partitioning methods is evaluated.

   The concrete objectives are:

   - Quantification and interpretation of differences in predictive performance estimates between spatial and non-spatial partitioning methods.
   - Evaluation and inspection of predictive performance estimates across models for different (inconsistent) combinations of partitioning methods in a nested CV setting.

2. **Does the use of filter-based feature selection on highly correlated feature sets derived from hyperspectral data enhance the predictive performance when modeling tree defoliation caused by *F. circinatum*?**

   Hyperspectral remote sensing data is a widely used data source in forest health monitoring. Yet its use in modeling often results in highly correlated and high-dimensional datasets. Filter-based feature selection is a less known dimension reduction approach which can be combined with model optimization to reduce dimensionality while keeping computational resources relatively low.

   The concrete objectives are:

   - Do certain filter methods result in substantially better performance estimates across feature sets and algorithms?
   - How do ensemble filters compare to traditional filter methods?
   - Which variables were seen as most important in the respective feature sets for modeling tree defoliation?

3. **How can the process of selecting and applying spatial resampling methods be simplified?**

   At the start of this thesis, only few options for applying spatial resampling methods on multiple algorithms, including model optimization and feature selection, were available in R. By collecting these existing implementations and bundling those within an existing machine learning framework, the general application of spatial resampling methods in R should become easier.

   The concrete objectives are:

   - Simplified application of spatial resampling methods in CV and nested CV settings in a commonly used data science focused programming language, e.g. R.
   - Bundling of all currently available spatial resampling methods in a single tool, including references to their respective upstream implementations.

1.3 Thesis Outline

The organization of this cumulative thesis is as follows. In chapter 1, an introduction is provided outlining the motivation, research questions and study objectives. Next, basic concepts of parametric and non-parametric modeling are introduced followed by a section about spatial autocorrelation (SAC) and its influence on CV based model assessment. Last, an introduction into forest pathogens, specifically *Diplodia sapinea* and *Fusarium circinatum*, is given.

Chapter 2 compares multiple learning algorithms using different combinations of spatial cross-validation (SCV) methods in a nested CV setting. It focuses on evaluating differences between various combinations of resampling methods in the inner and outer loop of a nested CV setting. This is done on the use case of modeling *Diplodia sapinea*-infected trees in northern Spain using environmental variables.

Chapter 3 evaluates the use of various filter-based feature selection methods in a high-dimensional modeling scenario across a variety of modeling algorithms. Tree defoliation caused by *Fusarium circinatum* in four pine plots in northern Spain is modeled using hyperspectral remote sensing data. A

permutation-based variable importance analysis is performed on the best performing model-filter-dataset combination.

Chapter 4 introduces the R package `mlr3spatiotempcv` which aims to simplify the application of spatiotemporal resampling methods from various sources in a centralized place. It helps to simplify the execution of studies similar to the ones presented in chapter 2 and chapter 3.

Chapter 5 discusses the contributions and limitations of this work and chapter 6 summarizes the main conclusions.

1.4 Model design and construction

In supervised learning, a set of variables is fitted against a target variable, also known as the response variable. The explanatory variables are commonly referred to as predictor variables (Hastie et al., 2001). Parametric algorithms describe the underlying relationship between the explanatory variables and the target variable as a mathematical equation. They are based on probability distributions, e.g. mean or variance, and rely on various assumptions about the underlying distribution which must be met for the model to be valid (Pearce & Derrick, 2019). In contrast, non-parametric algorithms are not solely based on such distributions but allow for unspecified distribution parameters, which in turn relaxes the preliminary model assumptions (Sprent & Smeeton, 2016). The latter is one reason why machine learning algorithms, which are non-parametric, became popular in the last decade. In contrast to parametric models, non-parametric ones require so-called "hyperparameters" to be specified before the model fit (Sprent & Smeeton, 2016). The model fit and generalization capabilities of a fitted model are highly dependent on the chosen hyperparameters (Probst, Boulesteix, & Bischl, 2019). There can be any number of hyperparameters for a machine learning algorithm. Yet usually the number of hyperparameters ranges between two and five with only some algorithms, for example XGBoost, exposing ten or more (T. Chen & Guestrin, 2016).

Due to their importance on the resulting model and the large amount of possible definitions, hyperparameters are usually selected by a model optimization algorithm, e.g. Grid Search or Random Search (Bergstra & Bengio, 2012). The application of these, or even more sophisticated algorithms like Bayesian optimization, is usually performed in a (nested) CV setting. More detailed information about model performance assessment can be found in section 1.6. The application of model optimization algorithms is challenging these are not embedded within a machine learning algorithm. Hence, the application of an optimization method requires additional work and is often a time-consuming step, both because of setup-related tasks and execution time of the actual optimization (Bischl et al., 2021). Yet given the potentially large influence on the final model fit, the thorough execution is essential for fitting models which aim to represent the underlying data in the best possible way (Probst, Boulesteix, & Bischl, 2019).

The concepts for the selection of hyperparameters described above apply in a similar way to the selection of the machine learning algorithms themselves: no qualitative indicators exist which could predetermine the suitability of an algorithm for a specific dataset. Hence, multiple algorithm need be compared against each other in order to determine the best performing one in a "data-driven" manner (Hoffmann et al., 2019). The process of comparing different algorithms is commonly referred to as "benchmarking". The model construction and design points outlined above represent some of the focus areas of this thesis. These are addressed primarily within the case studies of chapter 2 and chapter 3.

1.5 Model interpretability

Model interpretability, also known as model inference, is an important part of the machine learning process. The ability to interpret a model allows to better understand how the underlying data was utilized to create the resulting predictions (Doshi-Velez & Kim, 2017). Due to their mathematical representation of the variable relationships, parametric models are easier to interpret than nonparametric ones. The formula resulting from the model fit can be used to inspect which value combinations of all variables are needed to yield a specific value of the response variable and how their mathematical relation to each other is (e.g., linear, quadratic) (Hastie et al., 2001). In contrast, the interpretability of non-parametric models is more difficult as these algorithms do not exhibit a interpretable mathematical representation of the underlying variable relationships. Due to this difference in interpretability, parametric models are usually called "white-box" models whereas machine learning ones are referred to as "black box" models (Molnar, 2019). To overcome the limited interpretability of machine learning models, model-agnostic methods like partial dependence plots (PDP), functional decomposition, ALE plots or permutation-based variable importance can be used to gain insights into the variable relationships (Molnar et al., 2022). These methods are often labeled as "global" methods because they describe how a model behaves as a whole when changing certain parts of the data (Ribeiro et al., 2016). Local methods, on the other hand, e.g., local surrogate models (LIME) or SHAP, aim to explain how individual predictions were estimated. Such insights can be of interest for local decision making or outlier analysis (Molnar, 2019).

In the spatial domain, model interpretability allows for potential insights into the underlying physical/environmental processes (Elith & Leathwick, 2009). Due to their high degree of interpretability the use of parametric models like GLM or GAM has a longstanding tradition in the ecological modeling field (Goetz et al., 2015; Guisan et al., 2002, 1999). Yet the development of new methods for interpreting nonparametric machine learning models has made their use more attractive in the last decade. Nevertheless, care needs to be taken to not overinterpret variable importance results from nonparametric models as these might be potentially biased due to underlying data effects such as autocorrelation (which might also affect model assessment in a negative way, see also section 1.6). In the case study presented in chapter 3, permutation-based variable importance is used on the best performing model to estimate the importance of individual variables on the resulting model performance. The use of this method and its limitations are further discussed in subsection 5.3.3.

1.6 Model validation

Model validation is the process of assessing a model's predictive performance. In the simplest case, an independent dataset exists against which the model predictions can be compared to (Hastie et al., 2001). It is important that validation datasets are representative for the actual prediction task the model should perform (Meyer et al., 2018). To reduce variance throughout the assessment, the validation process is usually executed on multiple test datasets.

As data is sparse and mainly aimed for learning purposes, model validation is commonly executed in a CV setting, i.e., available observations are used for both model training and testing (Efron & Gong, 1983; James et al., 2013). To reduce variance and increase the generalization of the estimate, the process is repeated many times (James et al., 2013). Prominent characteristics of CV are the use of partitioning methods which split the training data into equally sized data splits and the unique use of an observation

within the test set of a CV iteration (Stone, 1974). For each training set in a CV, model hyperparameters are typically optimized, even though this is an optional step. If so, the process is commonly referred to as "nested CV" as two layers of CV are used to estimate the final model performance.

The terms "validation set" and "test set" in this context are often used interchangeably in literature. Yet most often validation set refers to the data part that is withheld to optimize hyperparameters whereas the test set is used to estimate the final performance (James et al., 2013). In a nested CV setting, the validation set is not explicitly declared upfront but dynamically created from a subset of the training set in the inner loop of the CV.

Besides CV, other commonly used methods for model validation exist: bootstrapping, leave-one-out cross-validation (LOOCV), holdout, subsampling or in-sample resampling (Bischl et al., 2012). When compared to CV, these methods either allow for the reuse of observations in training and test sets (bootstrapping, subsampling), only use single observations for validation sets (LOOCV) or do not repeat the validation process multiple times (holdout).

Various generalized implementations of custom resampling methods were developed in the environmental/ecological modeling field during the last decade (e.g., Brenning, 2012; Meyer, 2020; Valavi et al., 2019). They primarily focus on accounting for SAC, which is one inherent data effect that might affect CV estimates in a negative way. A more detailed description of SAC, its role in (spatial) model validation and throughout this thesis, is given in section 1.7.

1.7 The role of spatial autocorrelation in environmental modeling

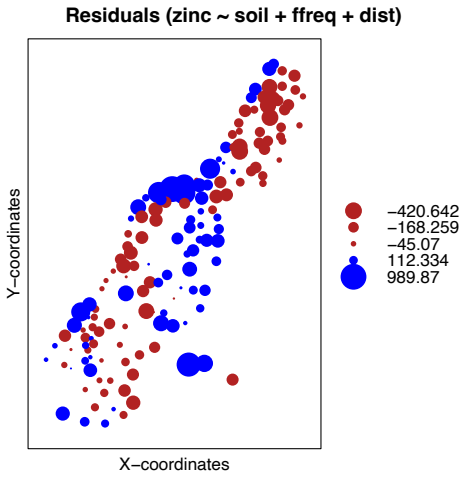
Environmental modeling is usually applied on spatial data. The spatial relation between data points is something one must be aware of if unbiased modeling results are desired. One particular phenomenon to account for is spatial autocorrelation, especially when estimating model performances (Brenning, 2012; Meyer et al., 2018; Roberts et al., 2017). The following sections provide a brief general introduction into (spatial) autocorrelation and a description of its influence on CV.

1.7.1 Spatial autocorrelation - general

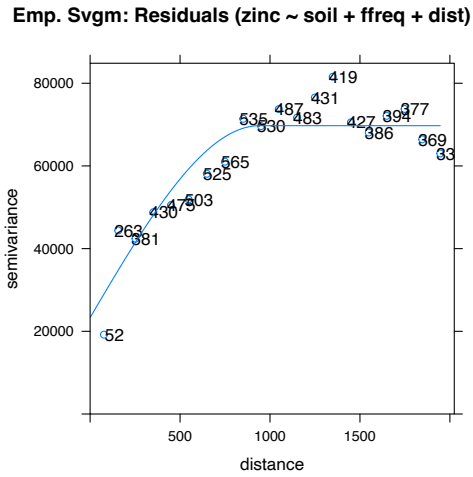
SAC is the observation that the degree of similarity between points in space is a function of their spatial distance, i.e., two points that are located in a close distance to each other are more likely to be very similar to each other than two very distant points (Koenig, 1999). When performing modeling, SAC often violates the independence assumption of model residuals, i.e., instead of being randomly distributed the residuals show a spatial clustering (Dale & Fortin, 2002; Liebhold & Sharov, 1998). This is an indicator that bias was introduced into the modeling process and the results cannot be fully trusted. SAC is known to have an influence on both model coefficients (Lichstein et al., 2002) and inference (Tognelli & Kelt, 2004).

To check whether the assumption of independence has been violated, one can plot the residuals of a fitted model against their Cartesian coordinates (Figure 1.1a). In the ideal case of independent model residuals, the diagnostic plots should show no pattern of grouping between positive or negative residuals. Figure 1.1a shows a dominant grouping which indicates the presence of spatial autocorrelation.

The creation of an empirical semivariogram based on the model residuals allows for further investigation (Figure 1.1b): In a semivariogram, the semi-variance is plotted against distance. The semi-variance is defined by O'Sullivan and Unwin (2003) as:



(a) Standardized residuals obtained by the linear regression model plotted versus their spatial coordinates. Red dots are negative and blue dots are positive residuals



(b) Empirical semivariogram of model residuals: Semi-variance against distance (m), calculated with a width of 100 m and a cutoff value of 2000 m. The numbers beside each observation denote the amount of point pairs which were used in each respective distance interval. The blue line represents the fitted spherical variogram values (nugget, sill, range) within the 2D variogram space.

Figure 1.1: Spatial autocorrelation diagnostic plots. Response and predictors were at random for this example and no inferences should be made on their selection. Data source: built-in 'meuse' dataset from R package `sp` (E. J. Pebesma & Bivand, 2005).

$$\hat{\gamma}(d) = \frac{1}{n(d)} \sum_{d_{ij}=d} (z_i - z_j)^2 \tag{1.1}$$

where:

$n(d)$ – pairs of observations per distance d,

z_i, z_j – observation

With the sum of squares of all observation pairs (z_i, z_j) for a given distance d being calculated and divided by their respective observation pair count for this distance (corresponding to the labels in Figure 1.1b). The resulting semi-variance $\hat{\gamma}$ is returned for the respective distance d.

A semivariogram is described by the parameters *range*, *sill* and *nugget*. The latter name originally goes back to the gold mining industry where it was common to find gold nuggets near a location where gold had already been discovered before. In statistics, "the nugget effect" describes the errors of measurement which are lying under the shortest sampling distance of which the semi-variance was calculated on. It is the semi-variance at which the fitted function crosses the y-axis ($d = 0$) (O'Sullivan & Unwin, 2003). The *range* is defined as the distance at which the semivariogram levels off, i.e., at which distance d the semi-variance starts to saturate. In the example semivariogram (Figure 1.1a), the range is roughly located at a distance

of 900 m. The *sill* is the semi-variance value at which the semi-variance saturates. An omnidirectional semivariogram as in Figure 1.1a assumes the spatial variation to be isotropic (i.e., equal in all directions). SAC may also only exist in one/multiple direction(s) or with a differing behavior for certain directions (e.g., spherical, Gaussian, exponential) (O'Sullivan & Unwin, 2003).

Accounting for such data patterns is essential in statistical model theory; otherwise the fitted model is not modeling the relationship but rather an underlying trend in the data (Zuur et al., 2009). One possible approach is to describe the existing autocorrelation structure in the data and include it into the model (Dormann, 2007; Dormann et al., 2007). Yet this requires an accurate description of the existing autocorrelation structure and models which are able to make use of these. This again applies only to a small number of models which are specialized on spatial data. Hence, alternative approaches to deal with SAC are required. Currently there are no solutions to explicitly account for SAC in a universal, model-agnostic way. Yet approaches exists to reduce the (negative) influence during different aspects of modeling (e.g., model validation). The following section gives more information about the influence of SAC in CV, which is also one of the main topics of this thesis.

1.7.2 Spatial autocorrelation in cross validation

The prevalent issue of SAC addressed by this work is its influence during performance estimation of models, specifically when using CV (Brenning et al., 2012; Ploton et al., 2020; Roberts et al., 2017). Here, SAC leads to (very) similar training and test sets due to their (close) spatial distance if random partitioning is used. This becomes an issue if the purpose of the model is extrapolation, i.e., prediction into remote areas. The more distant in space these prediction areas are to the data used for training and CV, the less representative the CV estimates become (Meyer et al., 2018).

SAC levels off at a certain distance (referred to as 'range' in a semivariogram). Beyond this distance, observations are assumed not to be autocorrelated anymore, i.e., being independent in a spatial context. However, if CV with random partitioning is performed, many observations in the training and test sets will be highly affected by the underlying spatial autocorrelation, i.e., they show high similarities, primarily because of their close distance to each other (Valavi et al., 2019). This similarity will most often result in a very high model performance as the fitted model will have a somewhat easy job making accurate predictions on the test set, which in turns lets the user make the assumption that the model is performing very well. Yet this only holds true for spatial areas which are located within the data used for the CV, i.e., when aiming to use the fitted model for interpolation purposes within the range of the SAC (Meyer & Pebesma, 2021). When predicting to remote areas, data is often substantially different from the ones used within the CV. This usually results in a substantially worse predictive performance than what has been estimated in the CV (Ploton et al., 2020).

Accounting for SAC by means of (spatial) data partitioning can help to reduce the influence of SAC during CV and result in predictive performances that are eventually closer to an extrapolation scenario, or, generally phrased, the desired prediction scenario in most ecological studies (Meyer & Pebesma, 2021). However, spatial data partitioning cannot fully account for SAC but only aim to reduce its influence (Meyer et al., 2018). With respect to the selection of a spatial partitioning method, no formal guidelines exists. The choice depends on the respective dataset at hand and in particular on its spatial distribution and available variables.

Analyzing the influence of SAC in CV plays an important role in this thesis. Specifically, different spa-

tial partitioning methods are used in both case studies (see chapter 2 and chapter 3) to reduce the influence of SAC during CV. Special focus is put on analyzing the influence when using spatial partitioning during the inner loop of a CV in chapter 2. In chapter 4 the `R` package `mlr3spatiotempcv` is presented, which aims to generalize and simplify the application of spatial partitioning methods. More details on this topic can also be found in chapter 4, specifically in section 4.2.

1.8 Reproducible research and software

1.8.1 Reproducible research

The idea of making research accessible and reproducible can be seen as the academic gold standard (Heil et al., 2021). It establishes trust in the conducted work and allows to reconstruct study experiments. It is important to distinguish between the terms "reproducibility" and "repeatability" (Plesser, 2018). The former is associated with "running the same software on the same input data and obtaining the same results" (Rougier et al., 2017) while the latter refers to the idea of copying/adapting a specific study setup and "obtaining results that are similar enough" (Rougier et al., 2017). The findability, accessibility, interoperability, and reusability (FAIR) principles from the data management field focus on data reproducibility (Wilkinson et al., 2016). Only following these principles does not yet make a study reproducible as the data management part is only one component. The following enumeration lists additional elements which are needed to accomplish full reproducibility of a programmatic study in the natural sciences:

1. Availability and reusability of code
2. Availability and reusability of data
3. Metadata information of software libraries used to execute the analysis
4. Order of execution (optional)

Bundling all of these alongside the written analysis part is called a "research compendium" (Gentleman & Temple Lang, 2007). Hereby the first two points provide the core reproducibility components of a study. While those points were intensively discussed and agreed on in various studies (X. Feng et al., 2019; Gundersen et al., 2018), the latter two points are less popular. These provide information on the used toolstack: which libraries were used, how the order of code execution needs to be to successfully recreate all parts of the study. Without this information, reproducibility attempts often fail.

Information about the execution order is marked as optional here as it does not block reproducibility in the first place. However, it can highly benefit reproduction efforts as it might prevent possible frustration related to the correct chaining of analysis steps. Other possible enhancements which could be added to the list above are the inclusion of hardware requirements coupled with time estimates of possibly long running analysis steps. This information can prevent the use of unsuitable hardware for reproduction purposes and allows for proper time planning.

All articles included in this work claim to be research compendia which ensure transparency and study reproducibility. The research compendia can be found under the following URLs/DOIs:

- Schratz et al. (2019): `https://doi.org/10.5281/zenodo.2582969`

- Schratz, Muenchow, et al. (2021): https://doi.org/10.5281/zenodo.2635403
- Schratz, Becker, et al. (2021): https://github.com/mlr-org/mlr3spatiotempcv

One challenge of this thesis was to find sophisticated solutions to points (3) and (4), i.e., simplifying the description of execution order and dynamically listing all used software libraries and their respective versions. The R packages `renv` (Ushey, 2021) and `drake` (Landau, 2018) were eventually chosen for these tasks. `renv` allows to create static project environments with a central recording of the used R packages. The state is saved in a file named `renv.lock` which can be used to recreate the full R project environment which was used for the original analysis (via `renv::restore()`).

The `drake` R package uses a makefile-based, domain-specific language approach to define the execution order of a project. This makes it possible to rerun an analysis without requiring specific knowledge on how individual analysis parts are connected internally. The connections are defined using R code and then read and interpreted by `drake`. During execution, `drake` takes care of resolving the individual components in the correct order, optionally in parallel. Additionally, already executed parts are cached to avoid redundant execution of already computed objects. If an analysis part is modified after execution, the subsequent components of the analysis which make use of this respective part get invalidated and will be recomputed. This is not only useful for reproduction purposes but also helps for general project management as it ensures that all downstream dependencies are getting recreated when a particular object in the analysis has changed, which is a known challenge in large projects.

1.8.2 Research software and code generalization in science

This thesis emphasizes the importance of research software to supplement scientific analyses and it's lasting value for the (research) community. Chapter 4 presents the R package `mlr3spatiotempcv`, which is an example of research software that was created to complement other analyses of this work (see studies in chapter 2 and chapter 3). It stands exemplarily for all research software contributions that were done throughout this work. Various code and documentation contributions were submitted to multiple R packages which were used in different analysis parts of this thesis (e.g., the `mlr` R package (Lang et al., 2019)) The overall incentive was to let a larger audience benefit from the undertaken efforts. A comprehensive list of all project contributions is left out on purpose as it would only add minor value to the overall scope of this work. The main point is to highlight the importance of code generalization through research software and its increasingly important role within data-driven analyses in academic research.

1.9 Forest pathogens

A large variety of fungal agents exist on the globe which can cause severe damage to pine trees: *Diplodia sapinea*, *Fusarium circinatum*, *Armillaria mellea*, *Heterobasidion annosum*, *Lecanosticta acicola* and *Dothistroma septosporum* are just a few which have been discussed in the literature in the past years (Aegerter & Gordon, 2006; Luchi et al., 2014; Mesanza et al., 2021, 2016; Watt et al., 2009). Besides the impact on the forest ecosystem itself, damages caused by these fungi also have an economic impact as pine trees are often grown in plantations to serve an economic purpose (e.g., timber production) (Iturritxa et al., 2014; Mead, 2013). In Europe, this applies especially to the north of Spain, more specifically to the Basque Country, which is the area this work focuses on.

Two of the most problematic fungi present in the Basque Country are *Fusarium circinatum* and *Diplodia sapinea* which will be introduced in greater detail in the following (Nirenberg & O'Donnell, 2018). Both are referred to as *F. circinatum* and *D. sapinea* from here onwards. The case studies make use of in situ data related to the presence of *D. sapinea* (chapter 2) and observed defoliation caused by *F. circinatum* (chapter 3).

According to an European Food Safety Authority (EFSA) report from 2010, over 10 million hectares of European pine forests are at risk by *F. circinatum* infection and its consequences. Due to the lack of an effective and environmentally friendly fungicide, most efforts focus on avoiding the dispersal of *F. circinatum* into disease-free countries and areas. Hence, great efforts have been undertaken in recent years to control the main pathways of spreading (EFSA, 2010). *F. circinatum* infections come with great economic losses due to the dieback of trees (Dwinell et al., 1985; Ganley et al., 2009; Wingfield et al., 2008).

Even though *D. sapinea* is known and being studied since the early 80s (Palmer et al., 1987; Paulpietersburg, 1980), the pathogen is still actively spreading across the globe. While *D. sapinea* does not necessarily cause the dieback of affected trees, it is able to put a tree under severe stress by causing shoot blight, stem cankers or blue stain (Iturritxa et al., 2013). The severe drought in 2003 substantially supported the epidemic spread of the pathogen in most of Europe (Blaschke & Cech, 2007). According to Chou (1982); Pérez-Sierra et al. (2007); Swart et al. (1988), the general range of *D. sapinea* susceptibility to various pine types is quite large, with a very high susceptibility to *P. Radiata*, which is also the most prevalent tree species in the Basque Country. With respect to economic damage potential, *D. sapinea* is roughly equivalent to *F. circinatum*. *D. sapinea* is likely to be released from its quiescent stage when a tree gets under water stress (Stanosz et al., 2001). With the ongoing climate change and it's temperature increase (IPCC, 2013), trees are more often facing water stress, favoring the release of this pathogen. The link to temperature, and climatic and environmental variables in general, was used as a starting point for the analysis of the pathogen in the first study (chapter 2) of this work: Linking the presence of *D. sapinea* against various environmental variables such as temperature, precipitation, soil type or pH to better understand the favorable living conditions of the fungus.

1.9.1 Historic origin and spread of *D. sapinea* and *F. circinatum* in Europe

Both *D. sapinea* and *F. circinatum* most likely originated in Chile and New Zealand. New Zealand's 1.8 million hectares of forest plantations are composed of 90% by *P. radiata*. The tree type is actually native in California, USA, and hence its presence in Europe can be classified as invasive (Holmes et al., 2009). Due to unmet production goals in Europe, infected *P. radiata* seeds were imported from Chile and New Zealand which most likely led to the biological expansion of these fungi in the early 2000's (Manzanos, Aragonés, & Iturritxa, 2019; Michel, 2003). Various studies reported outbreaks and detection of *D. sapinea* in (northern) Europe starting in the early 21st century: Brodde et al. (2019) reported the first sights of *D. sapinea* presence at single trees in Sweden in 2013 with the first large outbreak being reported in 2016. Even though *D. sapinea* presence was occasionally reported in Italy since the early 1900s, larger outbreaks in the Alpine region were observed at the beginning of the 21st century (Luchi et al., 2014). Müller et al. (2019) reported *D. sapinea* sightings in Estonia (2007) and Finland (2004). Later in 2013, fungus presence was confirmed even further East in Russia (Adamson et al., 2015), which demonstrated the ability of the fungus to survive under northeast-European climate conditions.

1.9.2 Diseases caused by *F. circinatum* and *D. sapinea*

D. sapinea causes shoot blight, which results in brown needles and mainly affects shoots. On stressed trees, the infection may spread from shoots into older branches, causing resinous cankers which are able to kill mature branches (Iturritxa et al., 2013). Other effects are the damping of seedlings, development of stem cankers and eventually the origination of blue stain (Paulpietersburg, 1980; Sinclair & Lyon, 2005). While according to Brodde et al. (2019), *D. sapinea* mainly affects the late wood growth, it may "kill the leader shoot(s), disrupt the shape of the growing crown, and decrease the quality of the stem." *D. sapinea* is one of the most common and widely distributed conifer pathogen worldwide (Whitehill et al., 2007). According to Manzanos, Stanosz, et al. (2019), *D. sapinea* was detected in 98% of dead tissue and at 24% of all survey points in their study.

(a) (b) (c)

Figure 1.2: Exemplary tree damages caused by *D. sapinea*: (a) shoot blight of seedlings, (b) blue stain, (c) patch of trees affected by shoot blight. Image copyright: E. Iturritxa

F. circinatum causes pine pitch canker (PPC) which was first mentioned in the literature in 1946 as a threat to pines in North Carolina, United States (Hepting & Roth, 1946). The first detection of the fungi in Spain, in Galicia, dates back to 1995 with additional findings in northern Spain in 1998 (Devey et al., 1999). The fungus causes primarily damage to *P. radiata* but is able to infect over 60 different pine species (Iturritxa et al., 2011). PPC causes "heavy exudation of resin" (Iturritxa et al., 2011), impacts tree growth and may also lead to tree death. Additional effects are "reduced seed germination, seedling blight, [and] canopy dieback" (Iturritxa et al., 2013). To infect a tree in the first place, *F. circinatum* needs open wounds (T. R. Gordon et al., 1998; Martín-García et al., 2019). These can have different origins; for example, from insect or weather damage (e.g., hail) (Schratz, 2016). *F. circinatum* can be present in seedlings without signs of infection. This involves the risk of introducing the fungi into new plantations (Martín-García et al., 2019).

1.9.3 Empirical and biological assessment of forest damage

This thesis analyzes forest damage caused by the forest pathogens *F. circinatum* and *D. sapinea*. How can forest damage be measured and quantified in the first place? To be able to address forest damage quantification, it helps to define the term "forest health". O'Laughlin et al. (1994) define forest health as "forest ecosystem health," i.e., the status of the forest ecosystem as a whole. Trees and tree health are just parts of such. The authors state that a comparison to human health is limited, yet both are complex systems and failing elements are able to potentially collapse the complete system. Forest health is measured by comparing quantitative indicators across time and other forests. Yet O'Laughlin et al. (1994) also state that indicators are often subjective (e.g., the presence of a root disease might be seen positively by people aiming for an increasing biodiversity). Their importance might highly differ between forest stands (e.g.,

(a) (b) (c)

Figure 1.3: Examples of pine pitch canker (PPC) caused by *F. circinatum* infections as observed in the field. Image copyright: E. Iturritxa

the importance of the presence of a specific soil type with respect to the respective species). Therefore indicators cannot easily be transferred to other forest stands, which limits their usefulness. However, they can be used for historic change detections of individual plots.

On the quantitative side, damages are often split into two categories: elemental damage and damage caused by insects or pathogens. Even though the data quality, especially the spatial resolution, was still quite low in the 1990s, studies aimed to make use of remote sensing imagery to assess forest health in a quantitative way (Entcheva et al., 1996; Royle & Lathrop, 1997; Treitz & Howarth, 1999). Elemental damages can be quantified relatively well using remote sensing data because of their widespread appearance on trees at roughly the same time (Chehata et al., 2014; Šimić Milas et al., 2015). Pathogen damage, on the other hand, happens below the tree crown level and causes damage which manifests slowly over an extended period and possible intensifies itself during the process, making damage assessments more difficult (Lovett et al., 2016). Nevertheless, remote sensing is a data source which can also be of help assessing pathogen damage. The underlying assumption is that damaged trees show symptoms of a disease which can be observed and identified remotely (G. Chen & Meentemeyer, 2016). These symptoms might allow for drawing inferences about the underlying sickness of the tree, which again allows for speculation about a potential pathogen infection. The data acquired from remote sensing imagery can be used in statistical models, thresholding or change detection approaches to help map the current state of a forest stand (G. Chen & Meentemeyer, 2016). The target variable in suchlike modeling analyses is primarily defoliation (Adelabu et al., 2014; de Beurs & Townsend, 2008). Yet studies also aimed to model tree mortality caused by insect damage (Verbesselt et al., 2009) or used mixed-effect models to account for environmental uncertainties (Rullán-Silva et al., 2015) during the modeling process. A thorough coverage of (high-risk) areas might allow for an early detection of pathogen infections and the possibility to apply adequate measures which prevent the active spread of a disease (G. Chen & Meentemeyer, 2016).

Besides the quantitative measurement of forest damage, biological measures are conducted to better understand the root cause of existing damage and identify the responsible pathogen species (Iturritxa et al., 2011; Manzanos, Aragonés, & Iturritxa, 2019). These can be categorized into morphological, molecular and inoculation methods. All methods require the extraction of isolates from needles, or roots of possibly affected trees. Morphological methods use microscopes to analyze the isolates, e.g., by inspecting the size and overall shape of the spores Iturritxa et al. (2013). These are combined with molecular analyses which

extract DNS sequences from the isolates and compare such with an existing DNS database (Pfenning et al., 2014). Last, pathogenic tests following the "Koch postulates" (Evans, 1976) are used in which healthy trees or seedlings are inoculated with extracted isolates in controlled greenhouse environments to observe their effects over time (Iturritxa et al., 2013; Mesanza et al., 2021).

1.9.4 Measures taken against *D. sapinea* and *F. circinatum*

Even though various laboratory studies exist analyzing the living conditions and infection processes of *F. circinatum* and *D. sapinea*, Iturritxa et al. (2013) claims that there might be a substantial difference between laboratory results (e.g., infection rates) compared to the ones observed in the field and that only few studies conducted analyses on this topic so far. Given the complicated and expensive actions that would need to be taken to actively treat infected plantations, the consensus seems to be to reduce the impact and spread of the fungi rather than trying to eradicate these (Aegerter & Gordon, 2006).

Although fungicides exist for both pathogens, their use in European forests is forbidden for environmental reasons (Directive 2009/128/EC) and more environmentally friendly approaches are encouraged (Martín-García et al., 2019). Examples are tebuconazole, oxycarbozin or debacarb which are injected into "small, shallow holes in the outer xylem of the root flare" (Hartman et al., 2009). According to Martín-García et al. (2019), long-term actions, which focus on prevention rather than eradication, are composed out of (1) biological control, (2) thermotherapy and atmospheric pressure non-thermal plasma treatments, (3) inducers of resistance, and (4) genetic resistance. The understanding of the term "biological control" (1) changed throughout the last decades within the scientific community from "any control achieved through a living system with the exception of man" (Cook & Baker, 1983) to "the use of microbial antagonists to suppress diseases, including natural products extracted or fermented from various sources" (Gardener & Fravel, 2002). The thermotherapy approach (2) follows the idea of eradicating the fungi from the seeds or shoots by ensuring a constant temperature over several weeks which would kill the fungi without damaging the shoots or seeds (Wang et al., 2018). The idea of induced resistance (3) "represents a physiological state of enhanced plant defensive capacity stimulated by biotic or abiotic elicitors, whereby innate defenses are enhanced against subsequent challenges" (Martín-García et al., 2019) whereas genetic resistance (4) aims to utilize the natural resistance of certain species or populations. The approach of hybridization (i.e., combining a resistant species with a susceptible one) has shown promising results (Kanzler et al., 2014). However, due to the promising native resistance of some species, Iturritxa et al. (2013) recommends to increase the planting of such species instead, i.e., *P. pinea*, *Pseudotsuga menziesii*, *P. nigra* (resistance against *F. circinatum*) and *Pseudotsuga menziesii P. halepensis* (resistance against *D. sapinea*).

1.9.5 Forest stand key figures of Spain

This section provides brief metadata information about forests in the Basque Country, Spain, which is the main study area of this thesis. In Spain, 44% of all forest area is occupied with pine trees. In the year 2002, 63% of all *P. radiata* grew in the Basque Country. Fundazioa (2016) reported that as of 2016 47% of all *P. radiata* plantations of Spain were located in the Basque Country of which 54.95% is forest area and 31.2% of such are composed out of *P. radiata* (124000 ha), respectively.

There are no detailed current or historic assessments on the number of infected trees by *F. circinatum* or *D. sapinea* on the country level as usually only individual outbreaks and detections are recorded, e.g.,

for Europe in the European and mediterranean plant protection organization (EPPO) database (European and Mediterranean Plant Protection Organization, 2021), (Drenkhan et al., 2020).

CHAPTER 2

HYPERPARAMETER TUNING AND PERFORMANCE ASSESSMENT OF STATISTICAL AND MACHINE-LEARNING ALGORITHMS USING SPATIAL DATA

Schratz, P., Muenchow, J., Iturritxa, E., Richter, J., and Brenning, A. 2019. "Hyperparameter tuning and performance assessment of statistical and machine-learning algorithms using spatial data." Ecological Modelling 406: 109-20. https://doi.org/10.1016/j.ecolmodel.2019.06.002

Reprint permissions have been obtained from the publisher.

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 or Python), 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 CV strategies have been proposed in recent years in environmental as well as medical contexts to reduce bias in predictive performance. In this study we show 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 CV methods were used to evaluate model 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 (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. Though the sensitivity to hyperparameter tuning differed between the ML algorithms, there were in most cases no substantial differences between spatial and non-spatial partitioning for hyperparameter tuning. However, spatial hyperparameter tuning maintains consistency with spatial estimation of classifier performance and should be favored over non-spatial hyperparameter optimization. High performance differences (up to 47%) between the bias-reduced (SCV) and overoptimistic (non-spatial CV) CV 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.

2.1 Introduction

Spatial predictions are of great importance in a wide variety of fields including hydrology (Naghibi et al., 2016), epidemiology (W. Adler et al., 2017), geomorphology (Brenning et al., 2015), remote sensing (Stelmaszczuk-Górska et al., 2017), climatology (Voyant et al., 2017), soil sciences (Hengl et al., 2017) and

ecology (Baasch et al., 2010; Muenchow et al., 2013; Murase et al., 2009; Vorpahl et al., 2012). Ecological applications range from species distribution models (Halvorsen et al., 2016; Quillfeldt et al., 2017; Wieland et al., 2017) over plant disease and soil type modeling (Brungard et al., 2015; Heim et al., 2018) to resource selection (Baasch et al., 2010).

A typical example for a spatial prediction approach in ecology is the detection of fungi infection on Monterey pines (Iturritxa et al., 2014). Fungal species such as *Diplodia sapinea* inflict severe damages to *Pinus radiata* trees 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.

2.1.1 The special role of spatial autocorrelation in predictive modeling

All of the previously mentioned scientific fields have one thing in common: The observations inherit spatial information. One of the main challenges that comes with this information is the accounting for the influence of spatial autocorrelation in the data (Legendre, 1993). Cross-validation and bootstrapping are two widely used performance estimation techniques (Efron & Gong, 1983; A. D. Gordon et al., 1984; Kohavi, 1995). However, in the presence of spatial autocorrelation, estimates obtained using regular (non-spatial) random resampling maybe biased and overoptimistic. This has led to the adoption of spatial resampling in CV 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 CV (Legendre, 1993). Random sampling in CV leads to the selection of test observations that are spatially close to training observations. According to the first law of geography, close observations are frequently more similar to each other than observations further apart. This violates the fundamental assumption of independence in CV. 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". In this work 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 (Geiß et al., 2017; Meyer et al., 2018; Wenger & Olden, 2012), unfortunately many studies have been published in recent years that did not account for this problem (Bui et al., 2015; Smoliński & Radtke, 2016; Wollan et al., 2008; Youssef et al., 2015).

2.1.2 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). Even though recently a lot of effort has been put into improving the interpretability of machine-learning algorithms, their ability to make inference is still limited compared to parametric ones (P. Adler et al., 2018; Henelius et al., 2014). The former gained in popularity due to their ability to handle high-dimensional and highly correlated data and their less important model assumptions.

2.1.3 The importance of hyperparameter optimization

To reach robust 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 CV are used in this step, the adequacy of non-spatial partitioning 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 (Geiß et 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.

2.1.4 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:

- Comparison of the predictive performance of spatial and non-spatial partitioning methods. We expect 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 optimal hyperparameter tuning has a substantial effect on model performance.
- Comparison of the predictive performance of parametric (GLM, GAM) and non-parametric algorithms (BRT, RF, SVM, KNN). We expect that the predictive performance of non-parametric algorithms is substantially higher.

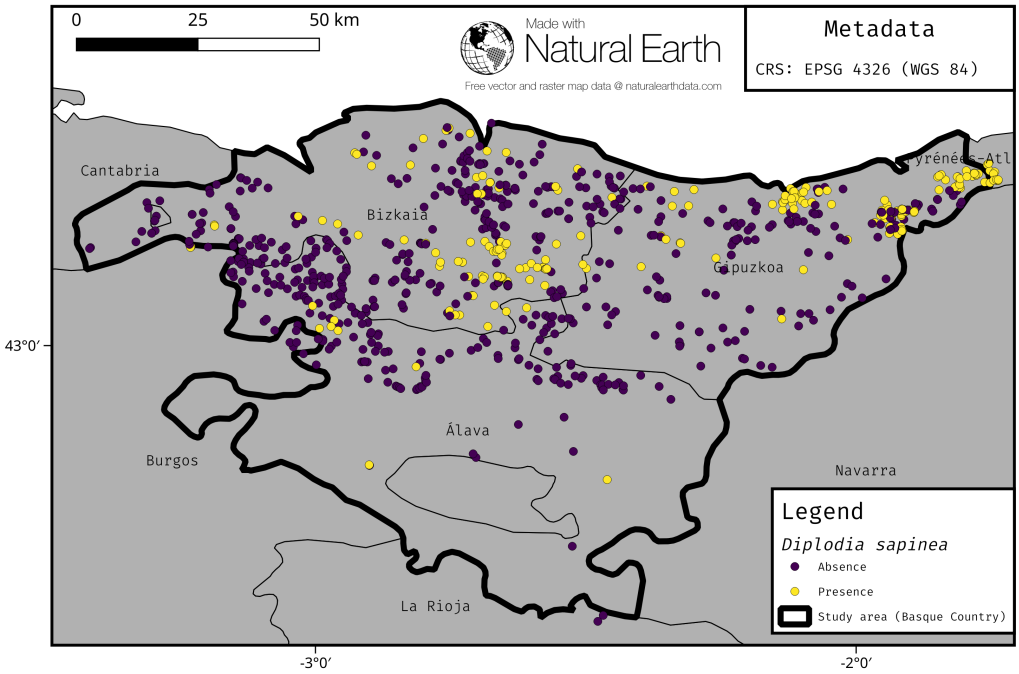


Figure 2.1: Spatial distribution of tree observations within the Basque Country, northern Spain, showing infection state by *Diplodia sapinea*.

2.2 Data and study area

2.2.1 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 the present study we also introduced additional predictors (probability of hail damage at trees, soil type, lithology type, pH) to possibly enhance the predictive power of the trained models.

This particular 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 (n=922), number of variables (n=11) and predictor types (numeric as well as nominal).

2.2.2 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). Precipitation, temperature and PISR were already attached to the dataset. All other variables were extracted to the point data from their raw sources.

(Iturritxa et al., 2014) showed in their study that the presence or absence of hail damage observed on trees is an important predictor when modeling 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 GAM (Schratz, 2016). In the following we shortly describe the source and modifications of the new variables. For the remaining ones, please see (Iturritxa et al., 2014).

Soil type was predicted by (Hengl et al., 2017) using approximately 150.000 soil profiles at a spatial resolution of 250 m. The age of trees was imputed and trimmed to a value of 40 to reduce the influence of outliers. The ph value 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 A2).

We removed three observations due to missing information in some variables leaving a total of 926 observations (Table A1). All nominal variables (soil and lithology-type) were dummy-encoded. To avoid introducing collinearity, the following reference levels of the dummy-encoded variables were removed from the data: soil type: "soils with clay enriched sub-soils". Lithology type: "surface deposits".

2.2.3 Study area

The Basque country in northern Spain represents the study area (Figure 2.1). It has a spatial extent of 7355 km². 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 km²), 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).

2.3 Methods

In this study we provide an exemplary analysis combining both tuning of hyperparameters (see subsection 2.1.3) using nested CV (see section 2.3.2) and the use of spatial CV to assess bias-reduced model performance (see subsection 2.1.1). We compared predictive performance using four settings: 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 the open-source statistical programming language R (R Core Team, 2019). The algorithm implementations of the following packages have been used: *gbm* (Ridgeway, 2017) (BRT, (Elith et al., 2008)), *mgcv* (Wood, 2017) (GAM), *kernlab* (Karatzoglou et al., 2004) (SVM, (Vapnik, 1998)), *kknn* (Schliep & Hechenbichler, 2016) (KNN, (Dudani, 1976)), and *ranger* (Wright & Ziegler, 2017) (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). The complete analysis including data is available as a research compendium at Zenodo (10.5281/zenodo.2582969).

2.3.1 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.

Parameter vs. hyperparameter

For parametric models the term "parameter" is often used to refer to the regression coefficients of each predictor of a 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.

Hyperparameters cannot be set manually if the best performance of a model is desired. Automatic optimization is necessary to determine the best setting. This optimization is done via procedures such as *random search* or *Bayesian optimization*. In contrast, parameters of parametric models are estimated when fitting them to the data (Kuhn & Johnson, 2013).

Table 2.1: 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	10000	100
	`shrinkage`	numeric	0.005	0.2	0.001
	`interaction.depth`	integer	1	20	1
KNN (kknn)	`k`	integer	1	100	7
	`distance`	integer	1	100	2
	`kernel`	nominal	*		
GAM (mgcv)	`sp`	numeric	0	10^6	-
RF (ranger)	m_{try}	integer	1	11	$\sqrt{p}$
	`min.node.size`	integer	1	10	1
	`sample.fraction`	numeric	0.2	0.9	1
SVM (e1071)	`cost`	numeric	2^{-5}	2^{12}	1
	γ	numeric	2^{-12}	2^3	1

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

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, 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).

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 the first outer fold that the number of tuning iterations set in the SMBO tuning was sufficiently large (Figure 2.4). 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.

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

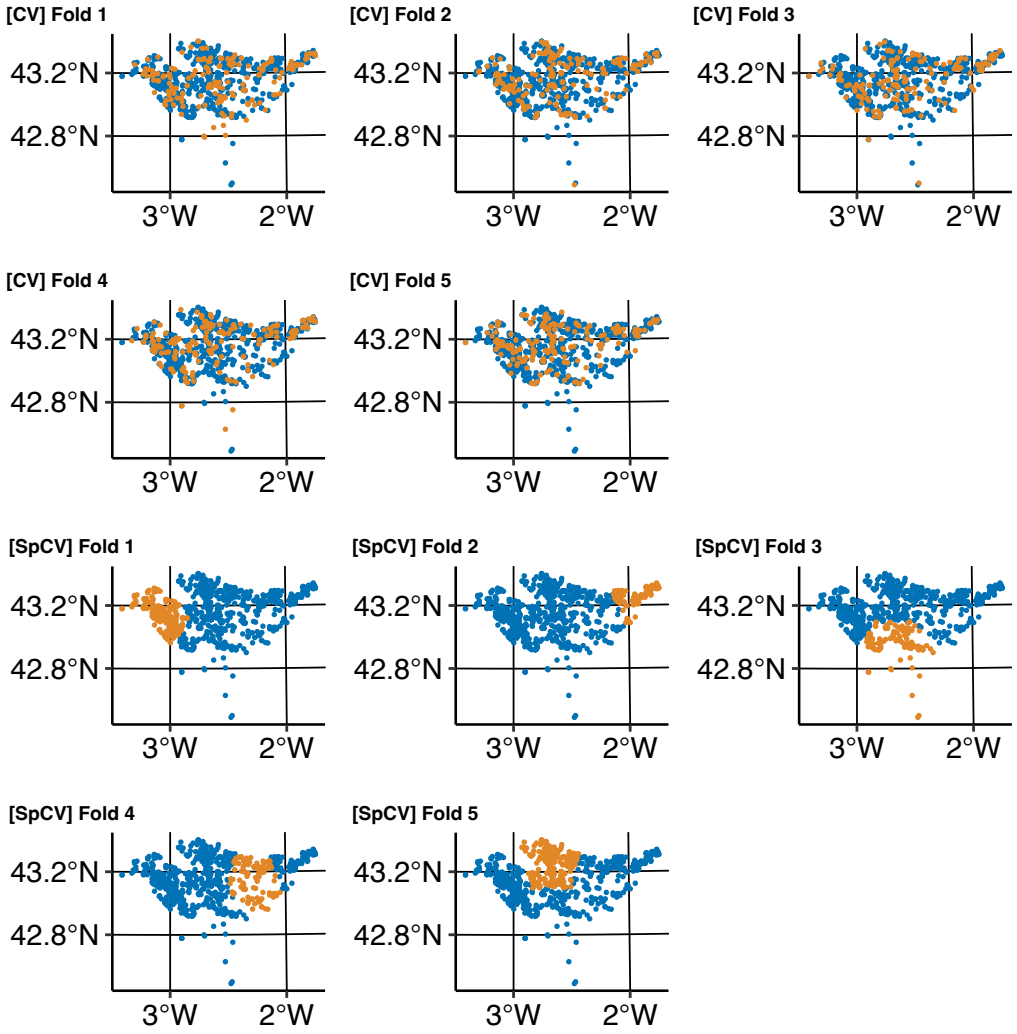


Figure 2.2: Comparison of spatial and non-spatial partitioning of the first five folds in spatial and non-spatial cross-validation performance estimation. Blue dots represent the training samples and orange dots the testing sample. "SpCV" stands for spatial cross-validation (spatial sampling of observations) and "CV" for cross-validation (random sampling of observations).

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).

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. Within the works of this study we implemented the spatial partitioning methods of package *sperrorest* into *mlr*.

2.3.2 Estimation of predictive performance

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 2.3). 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.

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 (Cliff & Ord, 1970; Legendre, 1993; 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 CV for tuning or validation (Burman et al., 1994; Cliff & Ord, 1970; Racine, 2000). Therefore, CV 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 SCV 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 2.3).

A 100 times repeated (to reduce random variability introduced by partitioning) five-fold partitioning setting was chosen for performance estimation (Figure 2.3). For hyperparameter tuning, again five folds were used to partition the training set of each fold. Hyperparameter tuning only applied to the machine-learning algorithms. A sequential model-based optimization approach was used for optimization (see subsection 2.3.1). 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).

Cross-Validation settings

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 in combination with non-spatial hyperparameter tuning (*spatial/non-spatial*)
- Nested spatial CV including spatial hyperparameter tuning (*spatial/spatial*)

- Spatial CV without hyperparameter tuning (*spatial/no tuning*)
- Non-spatial CV without hyperparameter tuning (*non-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.

Performance measure

Brier score was selected as a scoring rule 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)), 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).

A note on spatial autocorrelation structures in parametric models

In this work we expect that, on average, the predictive accuracy of parametric models with and without spatial autocorrelation structures incorporated into the model 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 CV 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.

2.4 Results

2.4.1 Tuning

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 2.4). Using 100 SMBO iterations, all shown folds show decrease in Brier score along the optimization path (Figure 2.4). 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.

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-spatial*, *non-spatial/non-spatial*) tuning settings (Figure 2.5): 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

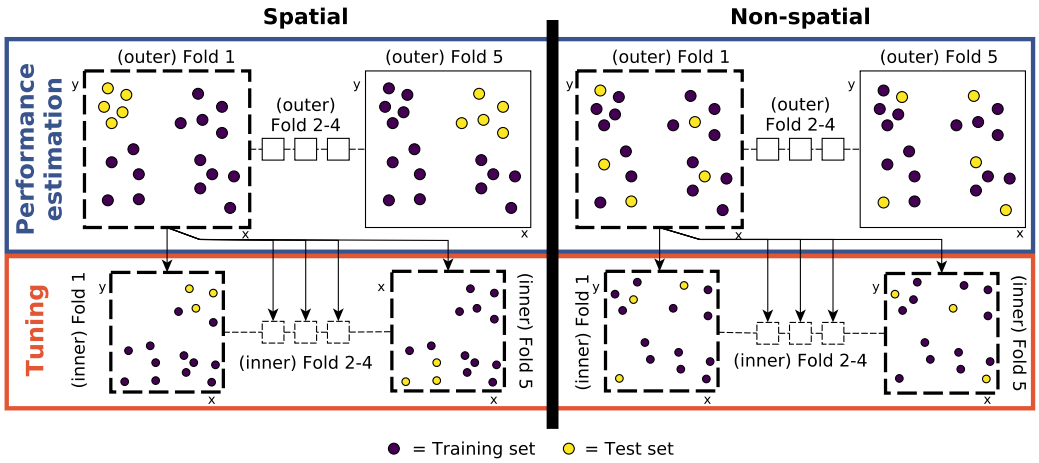


Figure 2.3: Theoretical concept of spatial and non-spatial nested CV using five folds for hyperparameter tuning and performance estimation. Yellow/purple dots represent the training and test set for performance estimation, respectively. The tuning sample is based on the respective performance estimation 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.

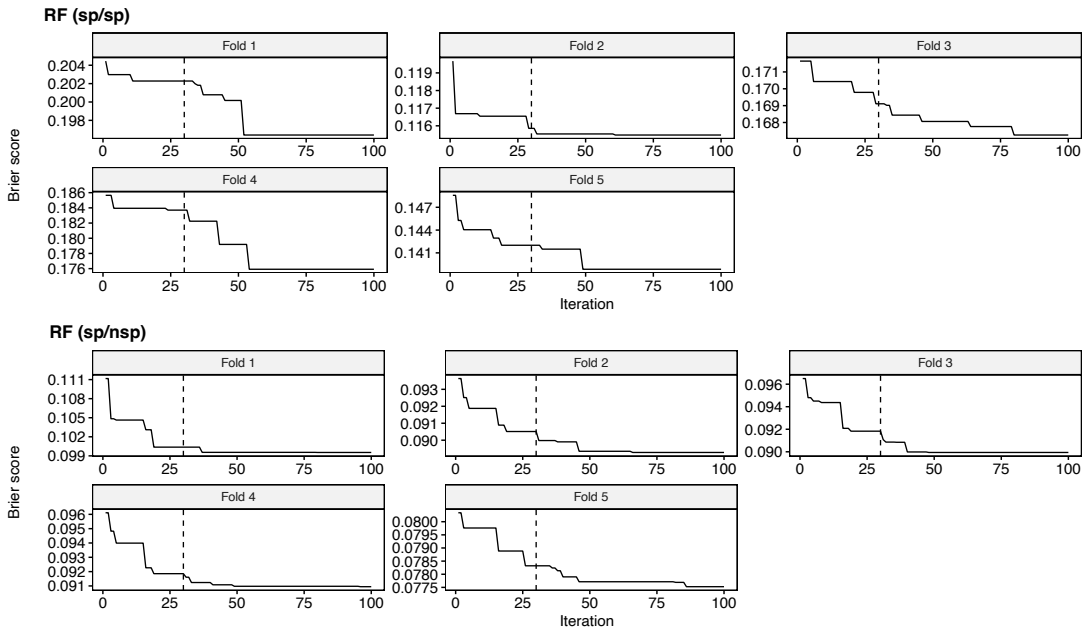


Figure 2.4: SMBO optimization paths of the first five folds of the *spatial/spatial* and *spatial/non-spatial* CV setting for RF. The dashed line marks the border between the initial design (30 randomly composed hyperparameter settings) and the sequential optimization part in which each setting was proposed using information from the prior evaluated settings. Optimization paths of the remaining models can be found in the appendix. Visualizations for other algorithms can be found in the research compendium of this study.

and often clusters around a few specific settings (e.g. compare the spatial and non-spatial results of the SVM) (Figure 2.5).

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 stands 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 2.5 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) which cannot be visualized in two dimensions.

2.4.2 Predictive performance

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 2.6). The absolute difference between the best (RF) and worst (GAM) performing model in our benchmark comparison is 0.039 (mean Brier score (*spatial/spatial*)). The GAM showed a high variance for all spatial settings compared to all other algorithms.

test

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%).

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 2.6). 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 a substantial positive impact for BRT and a negative one for GAM and KNN (Figure 2.6).

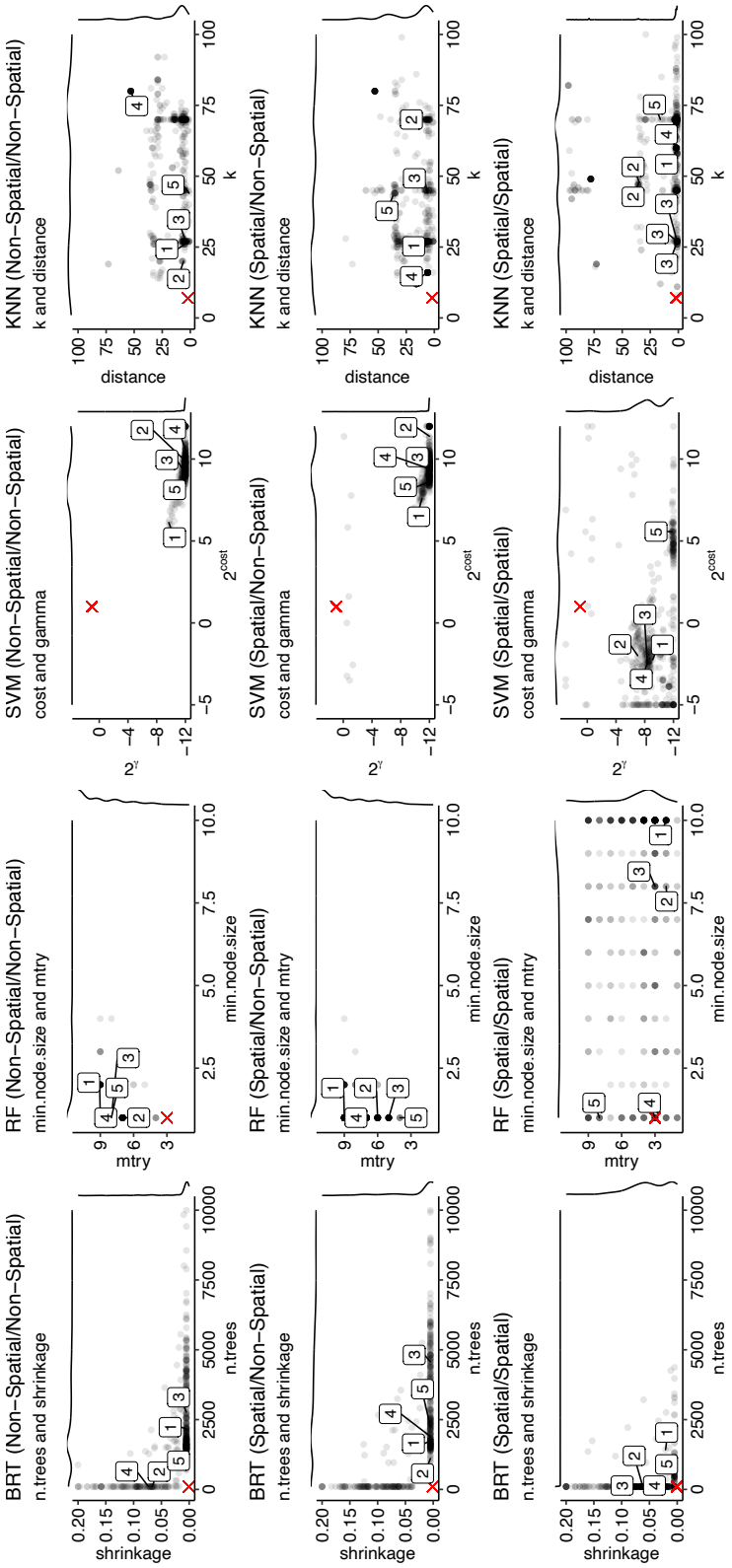


Figure 2.5: Best hyperparameter settings by fold (500 total) each estimated from 100 (30/70) SMBO tuning iterations per fold using five-fold CV. Split by spatial and non-spatial partitioning setup and model type. Red crosses indicate the default hyperparameters of the respective model. Black dots represent 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. Visualizations for other algorithms can be found in the research compendium of this study.

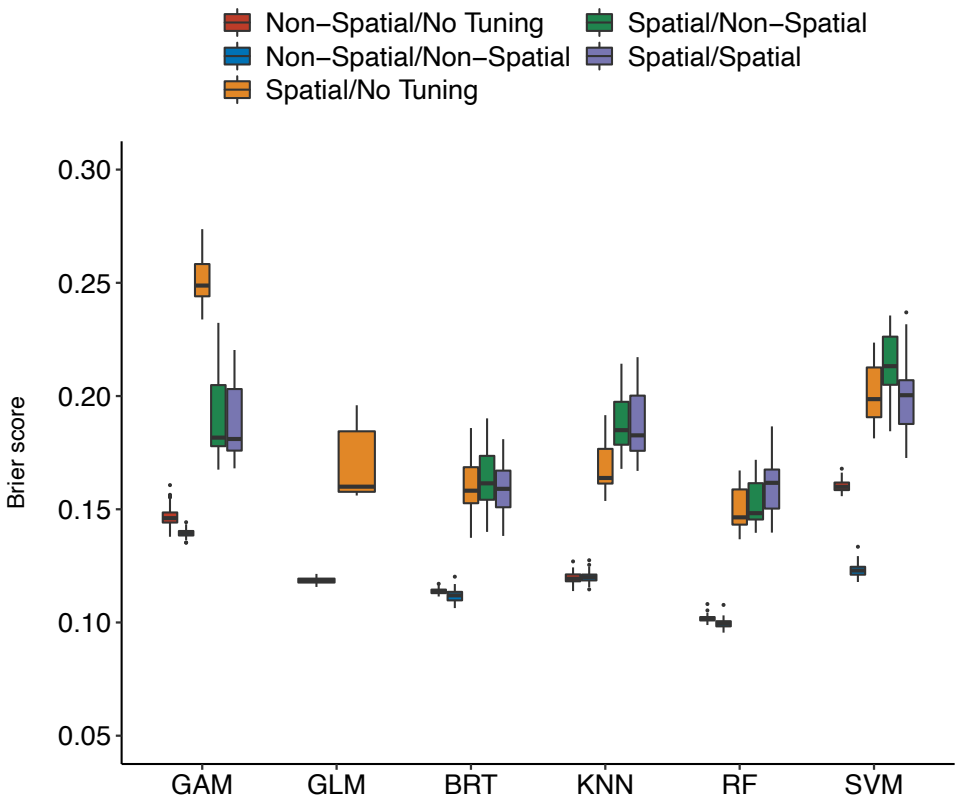


Figure 2.6: (Nested) CV estimates of model performance at the repetition level using 100 SMBO iterations for hyperparameter tuning. CV setting refers to performance estimation/hyperparameter tuning of the respective (nested) CV, e.g. "Spatial/Non-Spatial" means that spatial partitioning was used for performance estimation and non-spatial partitioning for hyperparameter tuning.

2.5 Discussion

2.5.1 Tuning

Tuning methods

The question on the most efficient approach of hyperparameter tuning has often been discussed (Bengio, 2000; Probst, Wright, & Boulesteix, 2019; 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 be expected to yield almost identical results but at the cost of increased computational efficiency and less

robustness in terms of finding the local minimum.

Algorithm sensitivity to tuning

Some models (e.g. RF) are known to be relatively insensitive to hyperparameter tuning (Probst, Boulesteix, & Bischl, 2019). 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.

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 substantial 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 2.5). However, using the extended setting, the algorithms did not converge anymore for some folds and at the same time runtime increased without a substantial 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).

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 2.6), the estimated optimal hyperparameters of RF differ for the non-spatial and spatial tuning setting (Figure 2.5). We split the following discussion into two points: (1) Explanation of the differences and (2) the implications of choosing a specific resampling method.

(1) The resampling method has no direct effect on how RF prioritizes variables internally. The Gini Impurity Index which is used to choose the variable that is used for splitting a node is calculated on a bootstrapped sample from the training data of the respective fold (Breiman, 2001). This applies for both spatial and non-spatial tuning. The Gini Index is not affected by spatial autocorrelation in this setting and RF will select the same variables in both resampling settings. Next RF is trained using the specific

hyperparameter set which was given in this fold (for example $m_{try} = 3$ and $min.node.size = 4$). Now the effect of choosing different resampling strategies applies:

- In the spatial setting, RF scored a low performance on the test set. The trained model overfitted on the training set.
- In the non-spatial setting, RF scored a good performance on the test. Here, the test set was highly similar to the training set and fitting the model closely to the training data resulted in good test set results.

The higher m_{try} and the lower $min.node.size$ are chosen, the more RF will overfit on the given data. This statement is backed up by the visualization of the chosen hyperparameter settings in each fold (Figure 2.5).
Ultimately, a spatial resampling in the tuning setting forces the algorithm to create a model that is more regularized than it would be in the case of a non-spatial resampling setting. This applies to all algorithms.

(2) Even though the estimated hyperparameters from a spatial and non-spatial sampling differ, they sometimes achieve the same performance when being evaluated at the performance estimation level of the CV (Figure 2.6). This outcome depends on the specific characteristics of the chosen dataset and algorithm. For example, SVM showed substantial differences between the resampling methods chosen during tuning while the effect for KNN was negligibly small. The findings of this work need to be verified by using other spatial datasets (and algorithms). 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 nlibot differ significantly.

2.5.2 Predictive Performance

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; Ruß & Brenning, 2010a; Steger et al., 2016). SCV is therefore required for performance estimation in spatial predictive modeling, and similar grouped CV 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).

The effect of hyperparameter tuning on predictive accuracy

Although hyperparameter tuning certainly increased the predictive performance for 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 2.6) although the SVM usually shows substantial increases when being tuned (Rojas-Dominguez et al., 2018).

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) although this is not always the case (Peña & Brenning, 2015). 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 was probably not able to generalize enough (i.e. it overfitted on the training set) in the spatial resampling setting. The high variance of the GAM performances in the spatial setting support this assumption: If the training set is 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 showed Brier score estimates up to 0.30. The linear approach of the GLM was able to generalize better in this study and subsequently resulted 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 capabilities of a GLM make this algorithm a valid choice, especially if the differences in predictive accuracy compared to black-box models are small.

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 which had the effect of GAM showing a similar performance as RF. So by only changing the measure, GAM went from the best (AUROC) to the worst (Brier score) algorithm. This example highlights the importance of selecting a measure for benchmarking purposes that is classified as a proper scoring rule. However, analyzing the effect of different measures on benchmarking performance across algorithms exceeds the scope of this work. Nevertheless, the use of the AUROC is justifiable in situations where relative indices of susceptibility are sought instead of predicted probabilities (e.g., hazard susceptibility modeling, (Goetz et al., 2015)).

A note on spatial autocorrelation structures in parametric models

In this work we expect that, on average, parametric models with and without residual autocorrelation structures are comparable. However, since model comparisons have focused on model behavior in statistical inference 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 re-estimate the spatial autocorrelation structure for every fold of a CV using a data-driven approach (i.e. automati-

cally fit a residual autocorrelation on each 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. However, if the aim is statistical inference, it is of utmost importance to include a spatial autocorrelation structure during model fitting.

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 ecological restoration 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. We encourage the use of spatial CV for performance estimation (Brenning, 2012; Ruß & Kruse, 2010), variable importance assessment (Brenning, 2012; Brenning et al., 2012) and hyperparameter tuning (this study).

2.5.3 Outlook

In this study, we focused on comparing resampling methods (spatial vs. non-spatial strategies) including hyperparameter tuning on a typical ecological dataset. Also we showed how to retrieve a bias-reduced performance estimate in the presence of spatial autocorrelation. Since we only used one dataset, the numeric outcomes are not generalizable. Still, we believe that future studies adapting the approach presented in this work will help with finding general patterns. It would be interesting to see if a spatial hyperparameter tuning (Figure 2.3) shows a more pronounced effect when other datasets are used. Most freely available datasets in the major repositories (Olson et al., 2017; Vanschoren et al., 2014) lack spatial information which obviously is the prerequisite for spatial data analysis.

Finally, ecological observations are often observed repeatedly at the same locations. In this case, the observations are most likely affected by both spatial and temporal autocorrelation. Therefore, one would have to adjust the methodology presented in this manuscript by incorporating the temporal dimension into the spatial resampling strategy.

2.6 Conclusions

In this study, we compared six statistical and machine-learning models in terms of predictive performance. With the exception of SVM, all machine-learning models outperformed (semi-)parametric models. More importantly, we found that non-spatial partitioning yields largely overoptimistic performance results in the presence of spatial autocorrelation.

By contrast, the effect of hyperparameter tuning on the predictive performance was less obvious, varies by algorithm and was overall smaller than the performance differences between algorithms. Additionally, the performance differences between spatial and non-spatial hyperparameter tuning were rather small. Still, we would recommend to use spatial CV instead of non-spatial CV for hyperparameter tuning when working with spatial data as only this ensures the assessment of bias-reduced predictive performance

results. This is especially important when the corresponding results form the basis of ecological and conservation decision making.

Finally, we recommend to clearly identify the main goal of an analysis from the beginning: If the goal is to disentangle environmental-ecological relationships with the help of statistical inference, (semi-)parametric models should be favored even if they fare less well in terms of predictive accuracy. By contrast, if the intention is to produce highly accurate spatial prediction maps, spatially tuned machine-learning models maybe the better choice.

Reproduced with permission from International Journal on Ecological Modelling and Systems Ecology; published by Elsevier, 2019.

CHAPTER 3

MONITORING FOREST HEALTH USING HYPERSPECTRAL IMAGERY: DOES FEATURE SELECTION IMPROVE THE PERFORMANCE OF MACHINE-LEARNING TECHNIQUES?

Schratz, P., Muenchow, J., Iturritxa, E., Cortés, J., Bischl, B., and Brenning, A. 2021. "Monitoring forest health using hyperspectral imagery: Does feature selection improve the performance of machine-learning techniques?" Remote Sensing, 2021, 13(23), 4832, https://doi.org/10.3390/rs13234832

Reprint permissions have been obtained from the publisher.

Abstract

This study analyzed highly correlated, feature-rich datasets from hyperspectral remote sensing data using multiple statistical and machine-learning methods. The effect of filter-based feature selection methods on predictive performance was compared. In addition, the effect of multiple expert-based and data-driven feature sets, derived from the reflectance data, was investigated. Defoliation of trees (%), derived from in situ measurements from fall 2016, was modeled as a function of reflectance. Variable importance was assessed using permutation-based feature importance. Overall, the support vector machine (SVM) outperformed other algorithms, such as random forest (RF), extreme gradient boosting (XGBoost), and lasso (L1) and ridge (L2) regressions by at least three percentage points. The combination of certain feature sets showed small increases in predictive performance, while no substantial differences between individual feature sets were observed. For some combinations of learners and feature sets, filter methods achieved better predictive performances than using no feature selection. Ensemble filters did not have a substantial impact on performance. The most important features were located around the red edge. Additional features in the near-infrared region (800 nm–1000 nm) were also essential to achieve the overall best performances. Filter methods have the potential to be helpful in high-dimensional situations and are able to improve the interpretation of feature effects in fitted models, which is an essential constraint in environmental modeling studies. Nevertheless, more training data and replication in similar benchmarking studies are needed to be able to generalize the results.

3.1 Introduction

The use of machine learning (ML) algorithms for analyzing remote sensing data has seen a huge increase in the last decade (Lary et al., 2016). This coincided with the increased availability of remote sensing imagery, especially since the launch of the first Sentinel satellite in the year 2014. At the same time, the implementation and usability of learning algorithms has been greatly simplified with many contributions from the open-source community. Scientists can nowadays process large amounts of (environmental) information with relative ease using various learning algorithms. This makes it possible to easily extend benchmark comparison matrices of studies in a semi-automated way, possibly stumbling upon unexpected findings, such as process settings, that would not have been explored otherwise (Ma et al., 2015).

ML methods in combination with remote sensing data are used in many environmental fields, such as vegetation cover analysis and forest carbon storage mapping (Mascaro et al., 2014; Urban et al., 2018). The ability to predict these environmental variables in unknown regions qualifies ML algorithms as a helpful tool for such environmental analyses. One aspect of this research field is to enhance the understanding of biotic and abiotic stress triggers, for example, by analyzing tree defoliation (Hawryło et al., 2018). Defoliation is known to be a proxy for pathogen and insect damage (Pollastrini et al., 2016). While common symptoms are observable across species, some effects and their degree of severity are species-specific (Gottardini et al., 2020). Defoliation has also been shown to increase predisposition of tree death from secondary biotic factors up to ten years after the actual defoliation event (Oliva et al., 2016). Other approaches for analyzing forest health include change detection (Zhang et al., 2016) or describing the current health status of forests on a stand level (Townsend et al., 2012).

Vegetation indices have shown the potential to provide valuable information when analyzing forest health (Adamczyk & Osberger, 2015; Jiang et al., 2014). Most vegetation indices were developed with the aim of being sensitive to changes in specific wavelength regions, serving proxies for underlying plant physiological processes. In some cases, indices developed for different purposes than the one to be analyzed (e.g., defoliation of pine trees) may help to explain complex underlying relationships that are not obvious at first. This emphasizes the need to extract as much information as possible from the available input data to generate promising features that can help in improving our understanding of the modeled relationship (Thenkabail et al., 2018). A less known index type that can be derived from spectral information is the NRI. In contrast to most vegetation indices, NRIs do not use an expert-based formula following environmental heuristics; instead, they make use of a data-driven feature engineering approach by combining (all possible) combinations of spectral bands (Thenkabail et al., 2000). When working with hyperspectral data, thousands of NRI features can be derived this way.

Even though ML algorithms are capable of handling highly correlated input variables, model fitting becomes computationally more demanding and model interpretation more challenging. Feature selection approaches can help to address this issue, reducing possible noise in the feature space, simplifying model interpretability, and possibly enhancing predictive performance (Cai et al., 2018). Instead of using wrapper feature selection methods, which add a substantial overhead to a nested model optimization approach, especially for datasets with many features, this study focuses on (ensemble) filter methods, which can be directly integrated into the hyperparameter optimization step during model construction.

Overall, this study aims to show how high-dimensional datasets can be handled effectively with ML methods while still allowing the interpretation of the fitted models. The predictive power of non-linear methods and their ability to handle highly correlated predictors is combined with common and new approaches for assessing feature importance and feature effects. However, this study focuses mainly on investigating the effects of filter methods and feature set types on predictive performance rather than interpreting individual feature effects.

Considering these opportunities and challenges, the research questions of this study are as follows:

- How do different feature selection methods influence the predictive performance of ML models of the defoliation of trees?
- Do different (environmental) feature sets show differences in performance?
- Can predictive performance be substantially improved by combining feature sets?
- Which features are most important and how can these be interpreted in this context?

In recent years, various studies that have used hyperspectral data to analyze pest/fungi infections on trees have been published. Pinto et al. (Pinto et al., 2020) successfully used hyperspectral imagery to characterize pest infections on peanut leaves using random forest, while Yu et al. (Yu et al., 2021) aimed to detect pine wilt disease in pine plots in China using vegetation indices derived from hyperspectral data. Other studies which applied hyperspectral data for forest health monitoring are (J. P. Dash et al., 2017; Kayet et al., 2019; H. Lin et al., 2014). Building upon these successful applications of hyperspectral remote sensing usage in the field of leaf and tree health monitoring, this work analyzes tree defoliation in northern Spain using airborne hyperspectral data. The methodology of this study uses ML methods in combination

with feature selection and hyperparameter tuning. In addition, feature importance was analyzed. Incorporating the idea of creating data-driven NRIs, this study also discusses the practical problems of high dimensionality in environmental modeling (Trunk, 1979; H. Xu et al., 2016).

3.2 Materials and Methods

3.2.1 Data and study area

Airborne hyperspectral data with a spatial resolution of one meter and 126 spectral bands were available for four Monterey Pine (*Pinus radiata D. Don*) plantations in northern Spain. The trees in the plots suffer from infections with pathogens such as *Diplodia sapinea (Fr.) Fuckel*, *Fusarium circinatum Nirenberg & O'Donnell*, *Armillaria mellea (Vahl) P. Kumm*, *Heterobasidion annosum (Fr.) Bref*, *Lecanosticta acicola (Thüm) Syd.*, and *Dothistroma septosporum (Dorogin) M. Morelet* causing (among others) needle blight, pitch canker, and root diseases (Iturritxa et al., 2017; Mesanza et al., 2016). The first two fungi are mainly responsible for the foliation loss of the trees analyzed in this study (Iturritxa et al., 2014). In situ measurements of defoliation of trees (serving as a proxy for tree health) were collected by visual inspection by experts. *Defoliation* in percent was used as the response variable (Figure 3.1).

It is assumed that the fungi infect the trees through open wounds, possibly caused by previous hail damage (Iturritxa et al., 2014). The dieback of these trees, which are mainly used as timber, causes substantial economic losses (Ganley et al., 2009).

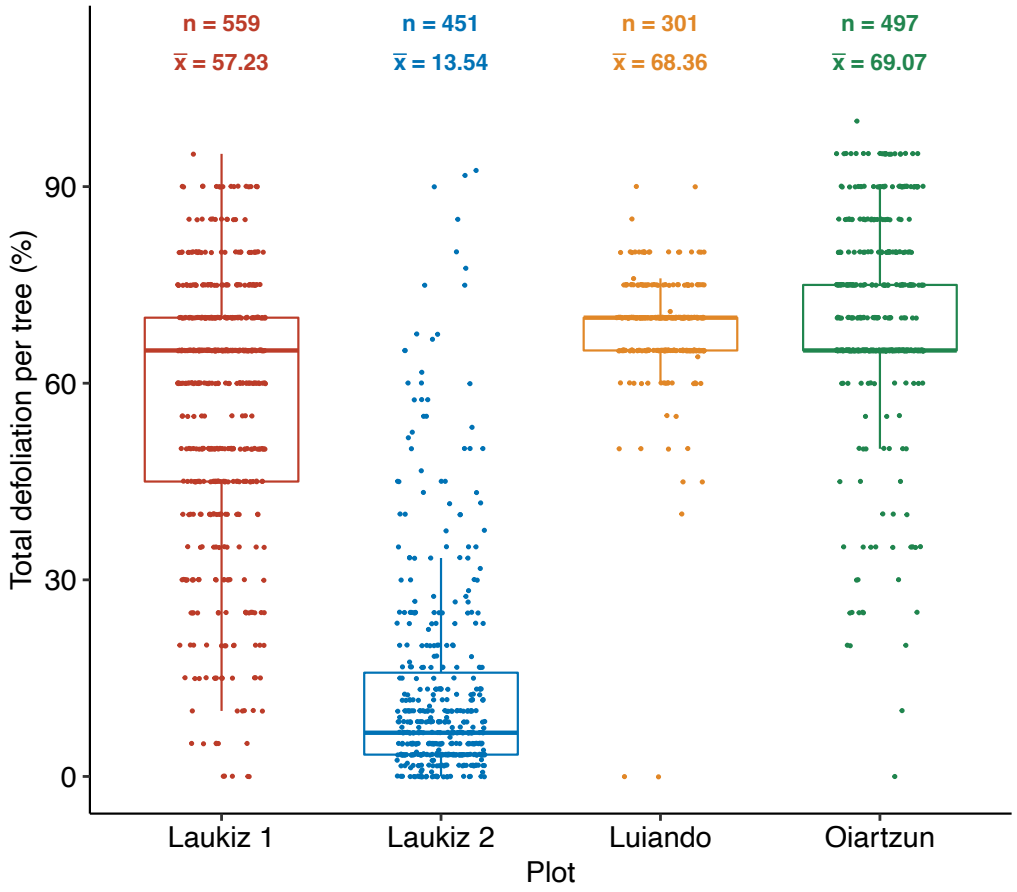


Figure 3.1: Response variable "defoliation of trees" for plots Laukiz1, Laukiz2, Luiando, and Oiartzun. n is the total number of trees in each plot, and $\bar{x}$ the mean defoliation. Values for Laukiz1, Luiando, and Oiartzun were observed in 5% intervals; for Laukiz2, defoliation was observed at multiple heights and then averaged, leading to smaller defoliation differences than 5%.

In situ data

The *Pinus radiata* plots of this study, namely Laukiz1, Laukiz2, Luiando, and Oiartzun, are located in the northern part of the Basque Country (Figure 3.2). Oiartzun has the largest number of observations (n = 559 trees), while Laukiz2 is the largest in area (1.44 ha). All plots besides Luiando are located within 100 km from the coast (Figure 3.2). A total of 1808 observations are available (Laukiz1: 559, Laukiz2: 451, Luiando: 301, Oiartzun: 497). Field surveys were conducted in September 2016 by experienced forest pathologists. Defoliation was measured in 5% steps through visual inspection with the help of a score card. For Laukiz2, values at three height levels (bottom, mid, and top) were available and averaged into an overall defoliation value, resulting in values that are not multiples of 5% (e.g., 8.33%). The magnitude of human observer errors in such surveys, including the present one, is not precisely known and has being discussed for many years (Innes, 1993). Ref. (MacLean & Lidstone, 1982) estimated human observer errors in defoliation surveys to range between 7% and 18%.

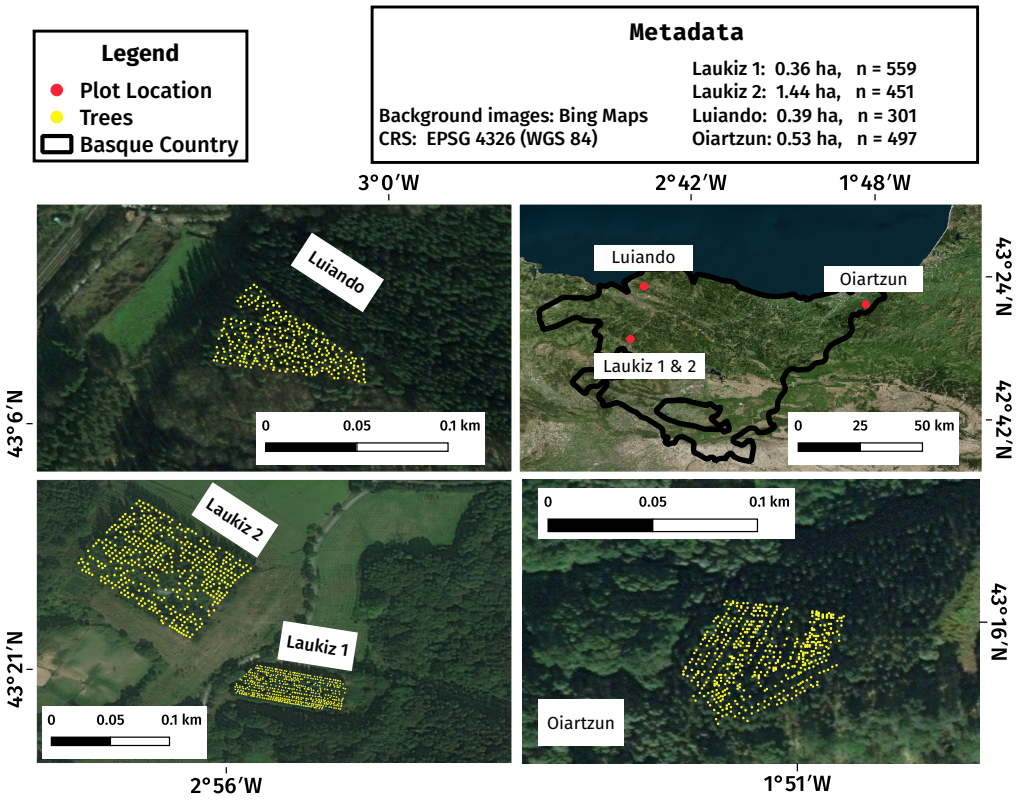


Figure 3.2: Study area maps showing information about location, size, and spatial distribution of trees for all plots (Laukiz1, Laukiz2, Luiando, and Oiartzun). The background maps give a visual impression of the individual plot area but do not necessarily represent the plot's state during data acquisition.

Hyperspectral data

The airborne hyperspectral data were acquired by an AISA EAGLE-II sensor during two flight campaigns on 28 September and 5 October 2016 at noon. All preprocessing steps (geometric, radiometric, and atmospheric) were conducted by the Institut Cartografic i Geologic de Catalunya (ICGC). The first four bands were corrupted, leaving 122 bands with valid information. Additional metadata information is available in Table 3.1.

Table 3.1: Specifications of hyperspectral data

Characteristic	Value
Geometric resolution	1 m
Radiometric resolution	12 bit
Spectral resolution	126 bands (404.08 nm–996.31 nm)
Correction:	Radiometric, geometric, atmospheric

3.2.2 Derivation of indices

To use the full potential of the hyperspectral data, all possible vegetation indices supported by the R package `hsdar` (89 in total) as well as all possible NRI combinations were calculated. NRIs follow the optimized

multiple narrow-band reflectance (OMNBR) concept of data-driven information extraction from narrow-band indices of hyperspectral data (Thenkabail et al., 2018, 2000). While various index formulations, such as band ratios or normalized ratios, are available, indices involving the same bands are strongly correlated. Since the widely used NDVI index belongs to the group of normalized ratio indices (NRIs), which are implemented in the `hsdar` R package, we used the following normalized difference index (NDI) formula to combine all pairs of reflectances:

$$NRI_{i,j} = \frac{band_i - band_j}{band_i + band_j} \tag{3.1}$$

where i and j are the respective band numbers.

To account for geometric offsets within the hyperspectral data, which were reported by ICGC to be potentially up to one meter, a buffer of one meter radius around the centroid of each tree was used when extracting the reflectance values. A pixel was considered to fall into a tree's buffer zone if the centroid of the respective pixel was touched by the buffer. The pixel values within a buffer zone were averaged and formed the final reflectance value of a single tree, and they were used as the base information to derive all additional feature sets. In total, $\frac{121*122}{2} = 7381$ NRIs were calculated.

3.2.3 Feature selection

High-dimensional, feature-rich datasets come with several challenges for both model fitting and evaluation:

- Model fitting times increase.
- Noise is possibly introduced into models by highly correlated variables (Johnstone & Titterington, 2009).
- Model interpretation and prediction become more challenging (Johnstone & Titterington, 2009).

To reduce the feature space of a dataset, several conceptually differing approaches exist: wrapper methods, filters, penalization methods (lasso and ridge), and principal component analysis (PCA) (Bommert et al., 2020; Das, 2001; Guyon & Elisseeff, 2003; Jolliffe & Cadima, 2016). In contrast to wrapper methods, filters have a much lower computational cost, and their tuning can be added to the hyperparameter optimization step. In addition, filters are less known than wrapper methods, and, in recent years, ensemble filters, which have shown promising results compared to single filter algorithms, were introduced (Drotár, Šimoňák, et al., 2017). These two points mainly led to the decision to focus on filter methods for this work and evaluate their effectiveness on highly correlated, high-dimensional datasets. Due to this focus, only this subgroup of feature selection methods is be introduced in greater detail in the following sections.

Filter methods

The concept of filters originates from the idea of ranking features following a score calculated by an algorithm (Guyon & Elisseeff, 2003). Some filter methods can only deal with specific types of variables (e.g., numeric or nominal). Filters only rank features; they do not decide which covariates to drop or keep (Drotár et al., 2015). The decision of which features to keep for model fitting can be integrated into the optimization phase during model fitting, along with hyperparameter tuning. Thus, the number of

top-ranked features to be included in the model is treated as an additional hyperparameter of the model. This hyperparameter is tuned to optimize the model's performance.

Beyond the use of individual filter methods to rank and select features, recent studies have shown that combining several filters by using statistical operations such as "minimum" or "mean" may enhance the predictive performance of the resulting models, especially when applied to multiple datasets (Abeel et al., 2010; Drotár, Šimoňák, et al., 2017). This approach is referred to as "ensemble filtering" (Dietterich, 2000). Ensemble filters align with the recent rise of the ensemble approach in ML, which uses the idea of stacking to combine the predictions of multiple models, aiming to enhance predictive performance (Bolón-Canedo & Alonso-Betanzos, 2019; Feurer et al., 2015; Polikar, 2012). In this work, the Borda ensemble filter was used (Drotár, Šimoňák, et al., 2017). Its overall feature order is determined by the sum of filter ranks of all individual filters in the ensemble.

Filter methods can be categorized based on three binary criteria: multivariate or univariate feature use, correlation or entropy-based importance weighting, and linear and non-linear filter methodology. Care needs to be taken to not weigh certain classes more than others in the ensemble, as, otherwise, the ranking will be biased. In this study, this was taken care of by checking the rank correlations (Spearman's correlation) of the generated feature rankings of all methods against each other. If filter pairs showed a correlation of 0.9 or higher, only one of the two was included in the ensemble filter, selecting it at random.

Description of used filter methods

Filter methods can be classified as follows (Table 3.2):

- Univariate/multivariate (scoring based on a single variable/multiple variables).
- Linear/non-linear (usage of linear/non-linear calculations).
- Entropy/correlation (scoring based on derivations of entropy or correlation-based approaches).

Table 3.2: Article 2: List of filter methods used in this work, their categorization, and scientific reference

Name	Group	Ref.
Linear correlation (Pearson)	univariate, linear, correlation	(Pearson, 1901)
Information gain	univariate, non-linear, entropy	(Quinlan, 1986)
Minimum redundancy, maximum relevance	multivariate, non-linear, entropy	(X.-M. Zhao, 2013)
Carscore	multivariate, linear, correlation	(Zuber & Strimmer, 2011)
Relief	multivariate, linear, entropy	(Kira & Rendell, 1992)
Conditional minimal information maximization	multivariate, linear, entropy	(Fleuret, 2004)

The filter "Information Gain" in its original form is only defined for nominal response variables:

$$H(Class) + H(Attribute) - H(Class, Attribute) \tag{3.2}$$

where H is the conditional entropy of the response variable (class or Y) and the feature (attribute or X). $H(X)$ is Shannon's entropy (Shannon, 1948) for a variable X, and $H(X, Y)$ is a joint Shannon's entropy for a variable X with a condition to Y. $H(X)$ itself is defined as

$$H(X) = -\sum_{i=1}^{n} P(x_i) \log_b P(x_i) \tag{3.3}$$

where b is the base of the logarithm used, most commonly 2.

In order to use this method with a numeric response (percentage defoliation of trees), the variable was discretized into equal bins $n_{bin} = 10$ and treated as a categorical variable.

3.2.4 Benchmarking design

Algorithms

The following learners were used in this work:

- Extreme gradient boosting (XGBoost);
- Random forest (RF);
- Penalized regression (with L1/lasso and L2/ridge penalties);
- Support vector machine (SVM, radial basis function Kernel);
- Featureless learner.

RF and SVM are well-established algorithms and widely used in environmental remote sensing. Extreme gradient boosting (commonly abbreviated as XGBoost) has shown promising results in benchmarking studies in recent years. Penalized regression is a statistical modeling technique capable of dealing with highly correlated covariates by penalizing the model coefficients (Hastie et al., 2001). Common penalties are "lasso" (L1) and "ridge" (L2). Ridge regression does not remove variables from the model (penalization to zero), but it shrinks them towards zero, keeping them in the model. A featureless learner was included for a baseline comparison.

In total, the benchmarking grid consisted of 156 experiments (6 feature sets × 3 ML algorithms × 8 feature-selection methods and for the L1/L2 models, 6 feature sets × 2 models. The selected hyperparameter settings are shown in Table A3. All code and data are included in the research compendium of this study (`https://doi.org/10.5281/zenodo.2635403`, accessed on 27 November 2021).

Feature sets

Three feature sets were used in this study, each representing a different approach to feature engineering:

- The raw hyperspectral band information (HR): no feature engineering
- Vegetation indices (VIs): expert-based feature engineering;
- Normalized ratio indices (NRIs): data-driven feature engineering.

The idea of splitting the features into different sets originated from the question of whether feature-engineered indices derived from reflectance values have a positive effect on model performance. Peña et al. 2017 (Peña et al., 2017) is an exemplary study that used this approach in a spectro-temporal setting. Benchmarking learners on these feature sets while keeping all other variables, such as model type, tuning strategy, and a partitioning method, fixed makes it possible to draw conclusions on their individual impact. Each feature set has distinct capabilities that differentiate it from the others. This can have both a positive

and negative effect on the resulting performance, which is one question this study aims to explore. For example, feature set VI misses certain parts of the spectral range, as the chosen indices only use specific spectral bands. Feature set NRI will introduce highly correlated features, for which some algorithms may be more suitable than others.

In addition to these individual feature sets, the following combinations of feature sets were also compared:

- HR + VI
- HR + NRI;
- HR + VI + NRI.

Some individual features were removed before using the datasets for modeling when being numerically equivalent to another feature based on the pairwise correlation being greater than $1 - 10^{-10}$. This preprocessing step reduced the number of covariates for feature set VI to 86 (from 89). This decision was made to prevent numerical issues that may occur in the subsequent tuning, filtering, and model fitting steps when offering features with a pairwise correlation of (almost) one. The remaining features were then used as input for the filter-based feature selection within the CV.

Hyperparameter optimization

Hyperparameters were tuned using model-based optimization (MBO) within a nested spatial CV (Binder et al., 2020; Bischl et al., 2017; Schratz et al., 2019). In MBO, first, n hyperparameter settings are randomly chosen from a user-defined search space. After these n settings have been evaluated, one new setting, which is evaluated next, is proposed by a fitted surrogate model (by default, a kriging method). This strategy continues until a user-defined stopping criterion is satisfied (Hutter et al., 2011; Jones et al., 1998).

In this work, an initial design of 30 randomly drawn hyperparameter settings in combination with a stopping criterion of 70 iterations was used, resulting in a total budget of 100 evaluated hyperparameter settings per fold. The advantage of this tuning approach is a substantial reduction of the tuning budget that is required to find a setting close to the global optimization minimum. MBO may outperform methods that do not use information from previous iterations, such as random search or grid search (Bergstra & Bengio, 2012). Tuning ranges used in this work are shown in Table A3.

To optimize the number of features used for model fitting, the percentage of features was added as a hyperparameter during the optimization stage (Binder et al., 2020). For PCA, the number of principal components was tuned. The RF hyperparameter m_{try} was re-expressed as $m_{\text{try}} = p_{\text{sel}}^t$, a function of the number of selected features, p_{sel}. It was thus tuned on a logarithmic scale by varying t between 0 (i.e., $m_{\text{try}} = 1$) and 0.5 (i.e., $m_{\text{try}} = \sqrt{p_{\text{sel}}}$). This was necessary to ensure that m_{try} did not exceed the number of features available after optimizing the feature percentage during tuning.

Spatial resampling

A spatial nested cross-validation on the plot level was chosen to account for spatial autocorrelation within the plots and assess model transferability to different plots (Brenning, 2012; Schratz et al., 2019). The RMSE was chosen as the error measure. Each plot served as one cross-validation fold, resulting in four

iterations in total. The inner level of cross-validation for hyperparameter tuning also used plot-level cross-validation.

3.2.5 Feature importance and feature effects

Estimating feature importance for datasets with highly correlated features is a challenging task for which numerous model-specific and model-agnostic approaches exist (J. H. Friedman, 2001; Greenwell et al., 2018; Hastie et al., 2001). The strong correlations among features make it challenging to calculate an unbiased estimate for single features (Molnar, 2019). Methods such as PDP or permutation-based approaches may produce unreliable estimates in such scenarios because unrealistic combinations of feature values are created (Molnar, 2019). The development of robust methods that enable an unbiased estimation of feature importance for highly correlated variables is subject to current research (Brenning, 2021).

In this work, permutation-based feature importance was calculated to estimate feature importance or effects (Apley & Zhu, 2020). With the limitations in mind when applied to correlated features, the aim was to get a general overview of the feature importance of the hyperspectral bands while trying to avoid an over-interpretation of results. The best-performing algorithm on the HR task (i.e., SVM) was used for the feature importance calculation.

To facilitate interpretation, the ten most important indices of the best performing models using feature sets HR and VI were linked to the spectral regions of the hyperspectral data that went into their calculation. The aim was to visualize the most important features along the spectral curve of the plots to better understand which spectral regions were most important for the model.

3.2.6 Research compendium

All tasks of this study were conducted using the open-source statistical programming language R (R Core Team, 2019). A complete list of all R packages used in this study can be found in the linked repositories mentioned in the next paragraph. Due to space limitations, only selected packages with high impact on this work are explicitly cited.

The algorithm implementations of the following packages were used: xgboost (T. Chen & Guestrin, 2016) (extreme gradient boosting), kernlab (Karatzoglou et al., 2004) (support vector machine) and glmnet (J. Friedman et al., 2010) (penalized regression). The filter implementations of the following packages were used: praznik (Kursa, 2018) and FSelectorRcpp (Zawadzki & Kosinski, 2019). Package mlr (Bischl et al., 2016) was used for all modeling related steps, and drake (Landau, 2018) was used for structuring the work and for reproducibility. This study is available as a research compendium on Zenodo (`10.5281/zenodo.2635403`, (accessed on 27 November 2021). Apart from the availability of code and manuscript sources, a static webpage is available at `https://pat-s.github.io/2019-feature-selection` (accessed on 27 November 2021) which includes additional side analyses that were carried out during the creation of this study.

Table 3.3: The overall best individual learner performance across any task and filter method for RF, SVM, XGBoost, Lasso and Ridge, sorted ascending by RMSE (p.p.) including the respective standard error (SE) of the cross-validation run. For `regr.featureless` the Task is not applicable and was therefore removed.

	Task	Model	Filter	RMSE	SE
1	NRI-VI	SVM	Info Gain	27.915	18.970
2	NRI	RF	Relief	30.842	12.028
3	HR	XGBoost	Info Gain	31.165	15.025
4	NRI	Lasso-MBO	No Filter	31.165	15.025
5	NRI	Ridge-MBO	No Filter	31.165	15.025
6	-	regr.featureless	No Filter	31.165	15.025

Table 3.4: Test fold performances in RMSE (p.p.) for learner SVM on the HR dataset without using a filter, showcasing performance variance on the fold level. For each row, the model was trained on observations from all others plots but the given one and tested on the observations of the given plot.

	RMSE	Test Plot
1	28.12	Laukiz1
2	54.26	Laukiz2
3	9.00	Luiando
4	21.17	Oiartzun

3.3 Results

3.3.1 Principal component analysis of feature sets

PCA was used to assess the complexity of the three feature sets. Depending on the feature set, 95% of the variance is explained by two (HR), twelve (acVI), and 42 (NRI) principal component (PC)s. HR features are therefore highly redundant, while the applied feature transformations enrich the data set, at least from an exploratory linear perspective.

3.3.2 Predictive performance

Overall, the response variable "tree defoliation" could be modeled with an RMSE of 28 percentage points (p.p.) (Figure 3.3). SVM showed almost no differences in RMSE across feature sets whereas other learners (RF, SVM, XGBoost, lasso and ridge) differed up to five p.p. (Figure 3.3). SVM showed the best overall performance with a mean difference of around three p.p. to the next best model (XGBoost) (Table 3.3). Performance differences between test folds were large: Predicting on Luiando resulted in an RMSE of 9.0 p.p. for learner SVM (without filter) but up to 54.3 p.p. when testing on Laukiz2 (Table 3.4).

The combination of feature sets showed small increases in performance for some learners. XGBoost scored slightly better on the combined datasets HR-NRI, NRI-VI, and HR-NRI-VI compared to their standalone variants (NRI and VI) (Figure 3.3). However, the best performances for RF and XGBoost were scored on NRI and HR, respectively. RF showed a substantial performance increase when using only NRI compared to all other feature sets, whereas for XGBoost, the worst performances were associated with the VI- and NRI-only feature sets (Figure 3.3).

Table 3.5: Best ten results among all learner-task-filter combinations, sorted in decreasing order of RMSE (p.p.) and their respective standard error (SE).

	Task	Model	Filter	RMSE	SE
1	NRI-VI	SVM	Info Gain	27.915	18.970
2	NRI	SVM	CMIM	28.044	19.101
3	VI	SVM	Relief	28.082	19.140
4	NRI-VI	SVM	Borda	28.102	19.128
5	HR	SVM	CMIM	28.119	19.123
6	HR	SVM	MRMR	28.119	19.123
7	VI	SVM	Info Gain	28.121	19.123
8	NRI	SVM	PCA	28.121	19.123
9	HR-NRI	SVM	PCA	28.121	19.123
10	HR-NRI-VI	SVM	PCA	28.121	19.123

Table 3.6: Selected feature portions during tuning for the best performing learner-filter settings (SVM Relief, RF Relief, XGBoost CMIM) across folds for task HR-NRI-VI, sorted by plot name. 'Features (#)' denotes the absolute number of features selected and 'Features (%)' refers to the percentage relative to the overall features available in the training sets for each plot (Laukiz1 = 1249, Laukiz2 = 1357, Luiando = 1507, Oiartzun = 1311). Results were estimated in a separate model tuning step, not within the main cross-validation comparison.

Learner	Test Plot	Features (%)	Features (#)
	Laukiz1	0.00245	1/1249
RF	Laukiz2	0.00359	1/1357
Car	Luiando	0.12448	2/1507
	Oiartzun	2.80356	37/1311
	Laukiz1	16.76686	210/1249
SVM	Laukiz2	40.77700	554/1357
Car	Luiando	43.80604	661/1507
	Oiartzun	81.23205	1065/1311
	Laukiz1	79.54091	994/1249
XGB	Laukiz2	0.96545	14/1357
Borda	Luiando	66.27871	999/1507
	Oiartzun	41.89759	550/1311

SVM combined with the "Information Gain" filter achieved the best overall performance (RMSE of 27.915 p.p.) (Table 3.5). Regressions with ridge (L2) and lasso (L1) penalties showed their best performances on the NRI feature set (Table 3.3). Combining feature sets for lasso and ridge did not help to increase performance, and while there was no substantial difference for lasso, the performance of ridge improved by around two percentage points. XGBoost showed very poor performances for some feature sets and fills the last ten places of the ranking (Table 3.7). Especially when combined with PCA, the algorithm was not

able to achieve adequate performances.

The effects of filter methods on performance differed greatly between the algorithms: SVM showed no variation in performance across filters (Figure 3.4). The use of filters for RF resulted in a substantial increase in performance in all tasks, especially on the HR feature set where all filters showed an improved score compared to using no filter (Figure 3.4). XGBoost's performance depended strongly on feature selection. In two out of six tasks (HR, VI), using no filter resulted in the worst performance. XGBoost showed the highest overall differences between filters for a single task—for feature set HR, the range is up to 13 p.p. ("CMIM" vs. "no filter") (Figure 3.4).

When comparing the usage of filters against using no filter at all, there were no instances in which a non-filtered model scored a better performance than the best filtered one (Figure 3.4). For SVM, all filters and "no filter" achieved roughly the same performance on all tasks.

The Borda ensemble filter was not able to score the best performance in any learner filter setting (Figure 3.5). For RF and XGBoost, it most often ranked within the better half among all filters of a specific task.

The number of features selected during model optimization strongly varied across learners and plots. RF selected the least features of all three learners, and with the exception of Oiartzun, only one or two variables were selected. SVM used 210 features or more in all instances and selected between 16% (Laukiz1) and 81% (Oiartzun) of the features (Table 3.6). XGBoost also favored using several hundred features with the exception of Laukiz2, for which only 14 (0.96%) were selected.

Table 3.7: Worst ten results among all learner-task-filter combinations, sorted in decreasing order of RMSE (p.p.) and their respective standard error (SE).

	Task	Model	Filter	RMSE	SE
1	VI	XGBoost	No Filter	45.366	6.672
2	HR	XGBoost	No Filter	44.982	5.378
3	VI	XGBoost	PCA	44.539	8.187
4	HR	XGBoost	PCA	44.032	6.183
5	NRI	XGBoost	PCA	43.433	9.543
6	HR-NRI	XGBoost	PCA	43.220	2.557
7	HR-NRI-VI	XGBoost	PCA	41.076	9.862
8	VI	RF	CMIM	39.980	10.144
9	VI	RF	Info Gain	39.623	10.616
10	NRI	XGBoost	Pearson	39.492	11.548

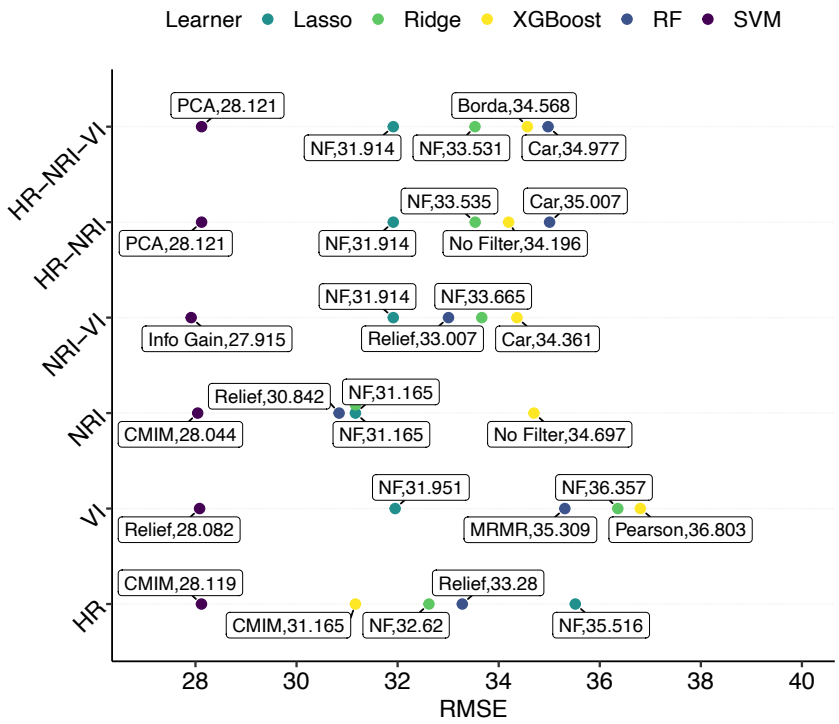


Figure 3.3: Predictive performance in RMSE (p.p.) of models across tasks. Different feature sets are shown on the y-axis. Labels show the feature selection method (e.g., NF = no filter, Car = "Carscore", Info Gain = "Information Gain", Borda = "Borda").

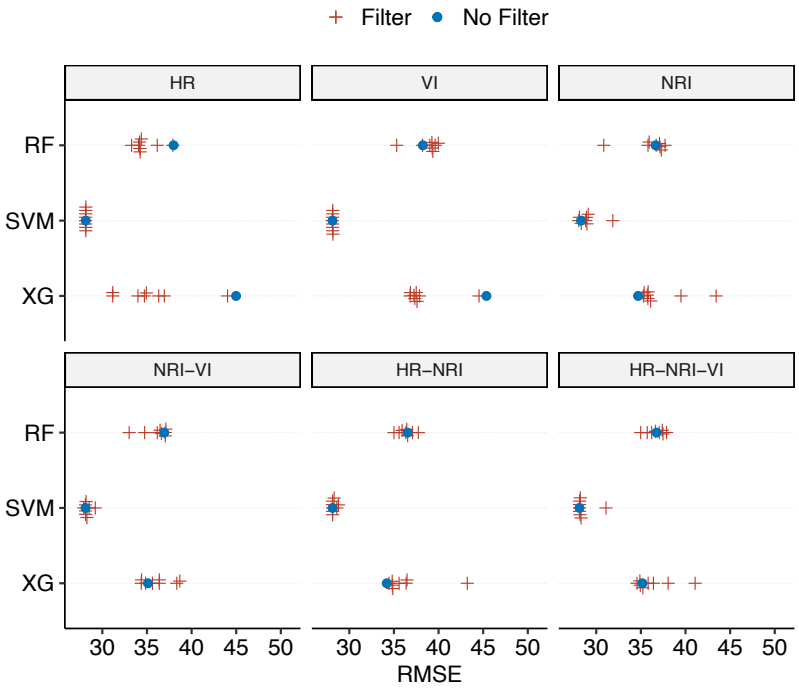


Figure 3.4: Model performances in RMSE across all tasks, split up in facets, when using no filter method (blue dot) compared to any other filter method (red cross) for learners RF, SVM, and XGBoost (XG).

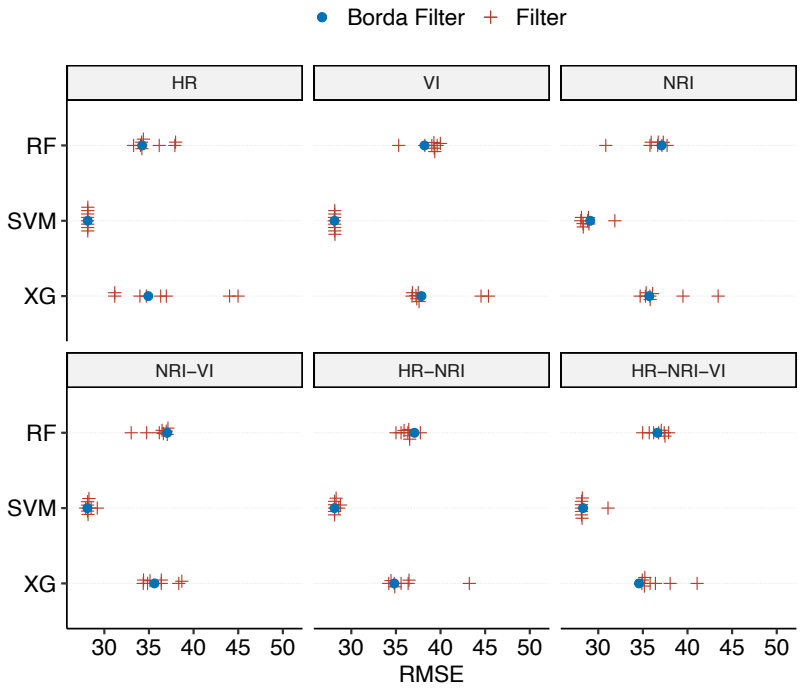


Figure 3.5: Predictive performances in RMSE (p.p.) when using the Borda filter method (blue dot) compared to any other filter (red cross) for each learner across all tasks.

3.3.3 Variable importance

Permutation-based variable importance

The most important features for datasets HR and VI showed an average decrease in RMSE of 1.06 p.p. (HR, B65) and 1.93 p.p. (VI, Vogelmann2) when permuted (Figure 3.6). For the HR dataset, most (i.e., six out of ten) relevant features clustered around the infrared region (920 nm–1000 nm), while for VI, eight out of ten concentrate within the wavelength range of 700 nm–750 nm (the so/called "red edge"). For HR, four features in the infrared region (920 nm–1000 nm) were identified by the model to be most important, being associated with a mean decrease in RMSE of around 1 p.p. Overall, apart from the top five features, the vast majority of features showed only a small importance with average decreases in RMSE below 0.5 p.p.

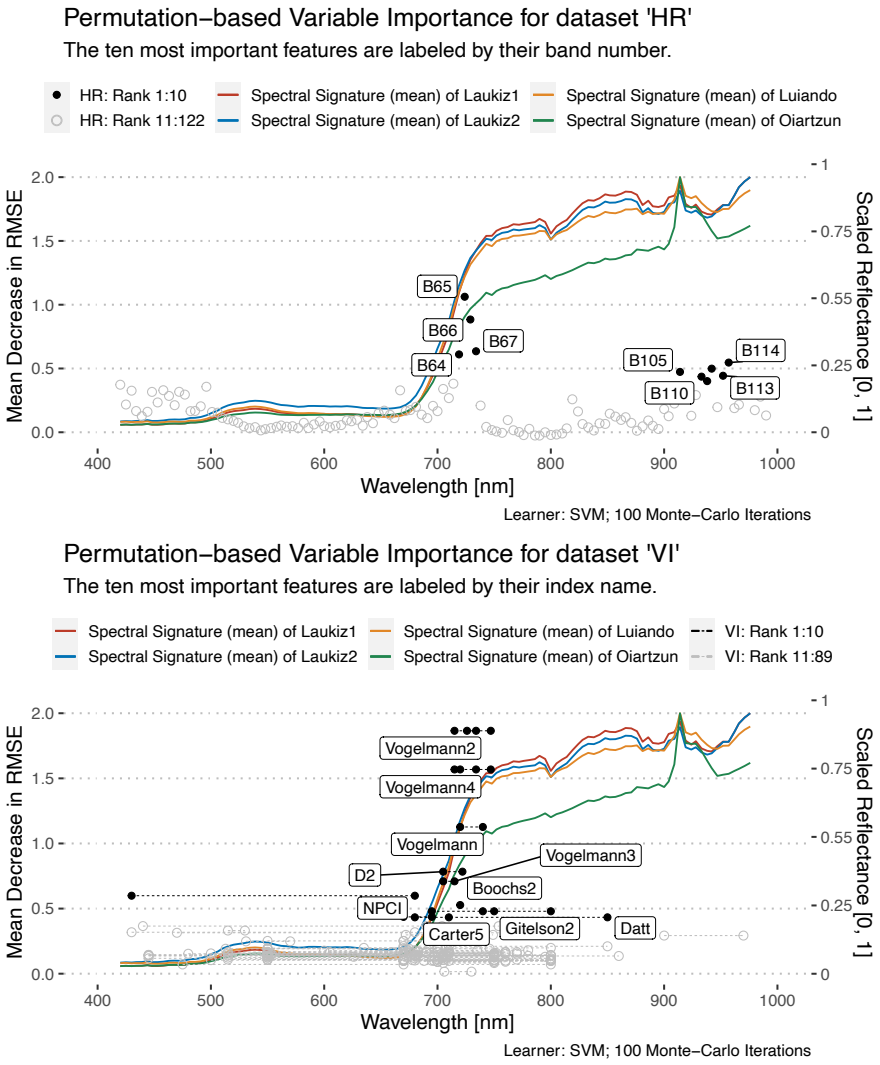


Figure 3.6: Variable importance for feature sets HR and VI: Mean decrease in RMSE for one hundred feature permutations using the SVM learner. The wavelength range on the x-axis matches the range of the hyperspectral sensor (400 nm–1000 nm). For each dataset, the ten most important features are highlighted as black dots and labeled by name. Gray dots represent features from importance rank 11 to last. The spectral signature (mean) of each plot was added as a reference on a normalized reflectance scale [0, 1] (secondary y-axis). VI features were decomposed into their individual formula parts, all instances being connected via dashed lines. Each VI feature is composed out of at least two instances.

3.4 Discussion

3.4.1 Predictive performance

The best overall performance achieved in this study (SVM with the "Info Gain" filter, RMSE 27.915 p.p.) has to be seen in the light of model overfitting (see subsection 3.4.2). Leaving out the performance on Laukiz2 when aggregating results, the mean RMSE would be around 19 p.p. However, leaving out a single plot would also change the prediction results for the other plots because the observations from Laukiz2 would not be available for model training. Due to the apparent presence of model overfitting in this study, it is suggested that more training data representing a greater variety of situations are needed. A model can

only make robust predictions if it has learned relationships across the whole range of the response. Hence, care should be taken when predicting on the landscape scale using models fitted on this dataset due to their lack of generalizability caused by the limitations of the available training data. However, when inspecting the fold/level performances, it can be concluded that the models performed reasonably well, predicting defoliation greater than 50%, but they failed for lower levels. This applied to all learners of this study. In this study, the overall performance across all learners can be classified as "poor" given that only the SVM learner was able to substantially outperform the featureless learner (Table 3.3). It is worth noting that data quality issues may have affected model performances, as discussed below in detail (subsection 3.4.5).

Model differences

An interesting finding is the strength of the SVM algorithm when comparing its predictive performance to its competitors (Table 3.3). These cluster around a performance of 31 p.p., while SVM scored about 3 p.p. better than all other methods. However, we refrain from comparing these results (both relatively and absolute) to other studies since many study design points have an influence on the final result (optimization strategy, data characteristics, feature selection methods, etc.).

A potential limiting factor in this study could be the upper limit of 70 iterations used for the XGBoost algorithm (hyperparameter `nrounds`), especially for feature sets including NRIs (Table A3). This setting was a compromise between runtime and tuning space extension with the goal to work well for most feature sets. It may be recommendable to increase this upper limit to a value closer to the number of features in the dataset in order to be able to exploit the full potential of this hyperparameter.

Feature set differences

One objective of this study was to determine whether expert-based and data-driven feature engineering have a positive influence on model performance. With respect to Figure 3.3, no overall positive or negative pattern related to specific feature sets was found that would be valid across all models. The performance of RF and XGBoost on the VI feature set was around 4 to 6 p.p. lower than on others. One reason could be the lack of coverage in the wavelength range between 810 nm and 1000 nm (Figure 3.6). In addition, for all learners but SVM, a better performance was observed when NRI indices were included in the feature set (i.e., NRI-VI, HR-NRI, and HR-NRI-VI).

3.4.2 Performance vs. plot characteristics

The large differences in RMSE obtained on different test folds can be attributed to model overfitting (Table 3.4). An RMSE of 54.26 p.p. reveals the model's inability to predict tree defoliation on this plot (Laukiz2). Laukiz2 differs highly in the distribution of the response variable defoliation compared to all other plots (Figure 3.1). In the prediction scenario for Laukiz2, the model was trained on data containing mostly medium-to-high defoliation values and only few low ones. This caused overfitting on the medium-to-high values, degrading the model's predictive performance in other scenarios. When Laukiz2 was in the training set, the overall mean RMSE was reduced by up to 50% with single fold performances as good as nine p.p. RMSE (with Luiando as test set).

There was also no clear pattern in the percentage of features selected based on hyperparameter tuning (Table 3.6). The optimal value for the number of features (interpreted as a percentage of available features),

which are selected from the ranked filter results, is determined by the internal surrogate learner of the MBO tuning method using the results from the previous tuning iterations. Due to this iterative approach, MBO is in some ways able to evaluate how well a learning algorithm plays together with a certain amount of selected features and is subsequently able to adjust the number of variables to an optimal value. In general, considering SVM's relative success, the use of at least a few hundred features from the combined feature set appears to be beneficial, or at least not harmful when the model's built-in regularization is capable of dealing with the resulting high-dimensional situation.

Realizing early during hyperparameter optimization that only a few features are needed to reach adequate performances can reduce the overall computational runtime substantially. Hence, regardless of the potential advantage of using filters for increased predictive performance, these can have a strong positive effect on runtime, especially of models making use of hyperparameters that depend on the available number of features, such as RF with m_{try}.

Ultimately, the results of Table 3.6 should be taken with care, as they rely on single model–filter combinations and are subject to random variation. More in-depth research is needed to investigate the effect of filters on criteria other than performance (such as runtime), leading to a multi-criteria optimization problem.

3.4.3 Feature selection methods

The usefulness of filters with respect to predictive performance in this study varied. While the performance of some models (up to 5 p.p. for RF and XGBoost) was improved by specific filters, some models achieved a poorer performance with filters than without them (Figure 3.4). There was no pattern of specific filters consistently resulting in better scores. Hence, it is recommended to test multiple filters in a study if it is intended to use filters. While filters can improve the performance of models, they may be more appealing based on other aspects than performance. Reducing variables can reduce computational efforts in high-dimensional scenarios and may enhance the interpretability of models. Filters are a lot cheaper to compute than wrapper methods, and the final feature subset selection can be integrated as an additional hyperparameter into the model optimization stage.

Models that used the Borda ensemble method in this study did not perform better on average than models that used a single filter or no filter at all. Ensemble methods have higher stability and robustness than single ones and have shown promising results in (Drotár, Šimoňák, et al., 2017). Hence, their expected main advantage is stable performances across datasets with varying characteristics. Single filter methods might yield better model performances on certain datasets but fail on others. The fact that this study used multiple feature sets but only one dataset and tested many single filters could be a potential explanation of why, in all cases, a single filter outperformed the ensemble filter. However, studies that use ensemble filters are still rare, and these are usually not compared against single filters (Ghosh et al., 2019). In summary, in this study, Borda did not perform better than a randomly selected filter method. More case studies applying ensemble filter methods are needed to verify this finding. Nevertheless, ensemble filters can be a promising addition to an ML feature selection portfolio.

PCA, acting as a filter method, more often showed less than optimal results, especially for algorithms RF and XGBoost. XGBoost in particular had substantial problems when using PCA as a filter method and accounted for four of the six worst results (Table 3.7). However, PCA was able to reduce model fitting times substantially across all algorithms. Depending on the use case, PCA can be an interesting option to reduce

dimensionality while keeping runtime low. However, information about the total number of features used by the model is lost when applying this technique. Since filter scores only need to be calculated once for a given dataset in a benchmark setting, the runtime advantage of a PCA vs. filter methods might in fact be negligible in practice.

3.4.4 Linking feature importance to spectral characteristics

Unsurprisingly, the most important features for both HR and VI datasets were identified around the red edge of the spectra, specifically in the range of 680 nm to 750 nm.

This area has the highest ability to distinguish between reflectances related to a high density/high foliage density and thus the health status of vegetation and its respective counterpart (Horler et al., 1983). However, four out of ten of the most important features of dataset HR are located between 920 nm and 1000 nm. Looking at the spectral curves of the plots, apparent reflectance differences can be observed in this spectral range—especially for plot Oiartzun—which might explain why these features were considered important by the model.

A possible explanation for the poor performances of most models scored on the VI dataset compared to all other feature sets could be the lack of features covering the area between 850 nm and 1000 nm (Figure 3.6). The majority of VI features covers the range between 550 nm–800 nm. Only one index (PWI) covers information in the range beyond 900 nm.

3.4.5 Data quality

Environmental datasets always come with some constraints that can have potential influence on the modeling process and its outcome. Finding a suitable approach to extract the remote sensing information from each tree was a complex process. Due to the reported geometric offset of up to one meter within the hyperspectral data, the risk of assigning a value to an observation that would actually refer to a different, possibly non-tree, pixel was reasonably high. It was concluded that using a buffer radius of one meter can be a good compromise between the inclusion of information from too many surrounding trees and an under-coverage of the tree crown. With the chosen radius, we are confident that we were able to map individual tree crowns while accounting for a possible geometric offset. This results in all cases in four contributing pixels (=four square meters) for the extraction of hyperspectral information for a given tree. Even though no results showing the influence of different buffer values on the extraction were provided, it is hypothesized that the relationships between features would not change substantially, leading to almost identical model results. Instead of using a buffer to extract the hyperspectral information, segmentation could have been considered. However, this method would have required more effort for no clear added value in our view and would have moved the focus of this manuscript more to data preprocessing and away from feature selection methods.

Trees located within grid cells on the border of a plot are a notable exception where the exact number of pixels contributing to the observation's feature value may be reduced since the image was cropped to the plot's extent. Cropping was applied to avoid the accidental inclusion of background data such as forest roads. However, this effect was deemed to be of negligible importance.

The available hyperspectral data covered a wavelength between 400 nm and 1000 nm. Hence, the spectral range of the shortwave infrared (SWIR) region is not covered in this study. Given that this range

is often used in forest health studies (Hais et al., 2019), e.g., when calculating the normalized difference moisture index (NDMI) index (B.-C. Gao, 1996), this marks a clear limitation of the dataset at hand.

The dataset consists of in situ data collected during September 2016, which was matched against remote sensing data acquired at the end of September 2016. A multi-temporal dataset consisting of in situ data from different phenology stages would possibly improve the achieved model performances. However, this would also require the costly acquisition of hyperspectral data of these additional timestamps.

The R package `hsdar` was used for the calculation of vegetation indices (Lehnert et al., 2018). All indices that could be calculated with the given spectral range of the data (400 nm–1000 nm) were used. This means that even though Table A4 lists all indices available in the package, not all listed indices were used in this study. Although this selection included a large number of indices, some possibly helpful indices might have been missed due to the restriction of the hyperspectral data.

Overall, the magnitude of uncertainty introduced by the mentioned effects during index derivation cannot be quantified. Such limitations and uncertainties apply to most environmental studies and cannot be completely avoided.

3.4.6 Practical implications on defoliation and tree health mapping

Even though this work has a strong methodological focus by comparing different benchmark settings on highly correlated feature sets, implications on tree health should be briefly discussed in the following. Due to the outlined dataset issues in subsection 3.4.5, which are mainly responsible for the resulting poor model performances, the trained models are not suited for practical use, e.g., to predict defoliation in unknown areas, due to the high mapping uncertainty. However, the general approach of utilizing hyperspectral data to classify the health status of trees partly led to promising results. For example, due to the narrow bandwidth of the hyperspectral sensor, small parts of the spectrum (mainly in the infrared region) were of higher importance to the models (e.g., see Figure 3.6), meaning that they helped the models to distinguish between low and high tree defoliation. If spatial offset errors of the image data and possible background noise can be reduced (possibly by making use of image segmentation), we believe that model performances could be substantially enhanced. Such improved models, starting around an RMSE of 20% and less, should be able to provide added value to support the long-term monitoring of forest health and early detection of fungi-affected tree plots. Nevertheless, overall the use of defoliation as a proxy for forest health should be critically reconsidered as it comes with various practical issues, starting from potential offsets during data collection, varying leaf density due to tree age, and differing effects between tree species, to name just a few.

3.4.7 Comparison to other studies

While most defoliation studies operate on the plot level using coarser resolution multispectral satellite data (de Beurs & Townsend, 2008; Rengarajan & Schott, 2016; Townsend et al., 2012), there are also several recent studies using airborne or ground-based sensors at the tree level. Among these, refs. (Kälin et al., 2019; R. Meng et al., 2018) used ground-level methods, such as airborne laser scanning (ALS) and light detection and ranging (LiDAR).

Studies focusing on tree-level defoliation mainly used ground-level methods, such as ALS or LiDAR (Kälin et al., 2019; R. Meng et al., 2018). Ref. (R. Meng et al., 2018) used ordinary least squares (OLS) regression methods while (Kälin et al., 2019) retrieving information from ground-level RGB photos using

convolutional neural networks (CNN). However, neither of them used spatial CV for model assessment, and (Kälin et al., 2019) did not perform feature selection (FS). The authors of (Goodbody et al., 2018) used a partial least squares (PLS) model with high-resolution digital aerial photogrammetry (DAP) to predict cumulative defoliation caused by the spruce budworm. Study results indicated that spectral features were found to be most helpful for the model. Incorporating such features (both spectral and structural) could be a possible enhancement for future works. No studies were found to model defoliation caused by *Diplodia sapinea (Fr.) Fuckel* with remote sensing data, and most studies focused on describing the tree conditions based on local sampling (Hlebarska & Georgieva, 2018; Kaya et al., 2019).

The field of (hyperspectral) remote sensing has had a strong focus on using RF for modeling in recent years (Belgiu & Drăguţ, 2016). However, in high-dimensional scenarios, tuning the parameter m_{try} becomes computationally expensive. To account for this and the high dimensionality in general, studies used feature selection approaches, such as semi-supervised feature extraction (Xia et al., 2015), wrapper methods (Fassnacht et al., 2014; J. Feng et al., 2016; Georganos et al., 2018), PCA, and adjusted feature selection (Rochac & Zhang, 2016). In general, applying feature selection methods on hyperspectral datasets has shown to be effective, regardless of the method used (Keller et al., 2016; Pal & Foody, 2010). However, no studies were found that made explicit use of filter methods in combination with hyperparameter tuning in the field of (hyperspectral) remote sensing. Potential reasons for this absence could be an easier programmatic access to wrapper methods and a higher general awareness of these compared to filter methods. Applying the filter-based feature selection methodology shown in this study and its related code provided in the research compendium might be a helpful reference for future studies using hyperspectral remote sensing data.

When looking for remote sensing studies that compare multiple models, it turned out that these often operate in a low-dimensional predictor space (S. Xu et al., 2019) or use wrapper methods explicitly (Georganos et al., 2018).

Refs. (Ludwig et al., 2019; Shendryk et al., 2016) are more similar in their methodology but focus on a different response variable (woody cover). Ref. (Shendryk et al., 2016) used machine learning with ALS data to study dieback of trees in eucalyptus forests. A grid search was used for hyperparameter tuning and forward feature selection (FFS) for variable selection. Ref. (Ludwig et al., 2019) analyzed woody cover in South Africa using a spatial CV and FS approach (Meyer et al., 2018) with an RF classifier. Ref. (Zandler et al., 2015) shows a similar setup; they used hyperspectral vegetation indices and a nested CV approach for performance estimation, and they estimated variable importance targeting woody biomass as the response. In the results, lasso showed the best performance among the chosen methods. However, the authors did not optimize the hyperparameters of RF, which makes a fair comparison problematic since the other models used internal optimization. The discussion section of (Zandler et al., 2015) lists additional studies that made use of shrinkage models for high-dimensional remote sensing modeling.

In summary, no studies could be found that used filter methods for FS or made use of NRI indices in their work and had a relation to tree health. This might relate to the fact that most environmental datasets are not high dimensional. In fact, many studies use fewer than ten features, and issues related to correlations are often solved manually instead of relying on an automated approach. These manual approaches might suffer from subjectivity and may limit the reproducibility of results.

Other fields (e.g., bioinformatics) encounter high-dimensional datasets more often. Hence, more studies using (filter-based) feature selection approaches can be found in this field (Guo et al., 2019; Radovic et al., 2017). However, bioinformatics differs conceptually in many

ways from environmental modeling, and, therefore, no greater focus was put into comparing studies of this field. The availability of high-dimensional feature sets will increase in the future due to higher temporal and spectral resolutions of sensors. In addition, a high-spatial resolution comes with the possibility of calculating many textural features. Hence, the ability to deal with high-dimensional datasets becomes more important, and unbiased robust approaches are needed. We hope that this work and its methodology raise awareness about the application of filter methods to tackle high-dimensional problems in the environmental modeling field.

3.5 Conclusions

This study analyzed the effectiveness of filter-based feature selection in improving various machine-learning models of defoliation of trees in northern Spain based on hyperspectral remote-sensing data. Substantial differences in performance occurred depending on which feature selection and machine learning methods were combined. SVM showed the most robust behavior across all highly correlated datasets and was able to predict the response variable of this study substantially better than other methods.

Filter methods were able to improve the predictive performance on datasets in some instances, although there was no clear and systematic pattern. Their effectiveness depends on the algorithm and the dataset characteristics. Ensemble filter methods did not show a substantial improvement over individual filter methods in this study.

The addition of derived feature sets was, in most cases, able to improve predictive performance. In contrast, feature sets that focused on only a small fraction of the available spectral range (i.e., dataset VI) showed a worse performance than the ones that covered a wider range (400 nm–1000 nm; HR, NRI). NRIs can be seen as a valuable addition to the optimization of predictive performance in the remote sensing of vegetation.

Features along the red-edge wavelength region were most important for models during prediction. With respect to dedicated vegetation indices, all versions of the Vogelmann index were seen as the most important indices for the best performing SVM model. This matches well with the actual purpose of these indices—they were invented to detect defoliation on sugar maple trees (*Acer saccharum Marsh.*) caused by pear thrips (*Taeniothrips inconsequens Uzel*) (Vogelmann et al., 1993). However, assessing feature importance for highly correlated features remains a challenging task. Results might be biased and should be taken with care to avoid overgeneralizing from individual studies.

Finally, the potential of predicting defoliation with the given study design was rather limited with respect to the average RMSE of 28 p.p. scored by the best performing model. More training data covering a wider range of defoliation values in a larger number of forest plantations are needed to train better models that can create more robust predictions.

CHAPTER 4

MLR3SPATIOTEMPCV: SPATIOTEMPORAL RESAMPLING METHODS FOR MACHINE LEARNING IN R

Schratz, P., Becker M., Lang M., and Brenning A. 2021. "mlr3spatiotempcv: Spatiotemporal resampling methods for machine learning in R." arXiv:2110.12674, https://arxiv.org/abs/2110.12674

Abstract

Spatial and spatiotemporal machine-learning models require a suitable framework for their model assessment, model selection, and hyperparameter tuning, in order to avoid error estimation bias and over-fitting. This contribution reviews the state-of-the-art in spatial and spatiotemporal CV, and introduces the `R` package mlr3spatiotempcv as an extension package of the machine-learning framework **mlr3**. Currently various `R` packages implementing different spatiotemporal partitioning strategies exist: `blockCV`, `CAST`, `kmeans` and `sperrorest`. The goal of `mlr3spatiotempcv` is to gather the available spatiotemporal resampling methods in `R` and make them available to users through a simple and common interface. This is made possible by integrating the package directly into the `mlr3` machine-learning framework, which already has support for generic non-spatiotemporal resampling methods such as random partitioning. One advantage is the use of a consistent nomenclature in an overarching machine-learning toolkit instead of a varying package-specific syntax, making it easier for users to choose from a variety of spatiotemporal resampling methods. This package avoids giving recommendations which method to use in practice as this decision depends on the predictive task at hand, the autocorrelation within the data, and the spatial structure of the sampling design or geographic objects being studied.

4.1 Introduction

Spatial and spatiotemporal prediction tasks are common in applications ranging from environmental sciences to archaeology and epidemiology. While sophisticated mathematical frameworks have long been developed in spatial statistics to characterize predictive uncertainties under well-defined mathematical assumptions such as intrinsic stationarity (Cressie, 1993), computational estimation procedures have only been proposed more recently to assess predictive performances of spatial and spatiotemporal prediction models (Brenning, 2005; Brenning et al., 2012; Pohjankukka et al., 2017; Roberts et al., 2017).

Although alternatives such as the bootstrap exist since some decades (Efron & Gong, 1983; Hand, 1997), cross-validation (CV) is a particularly well-established, easy-to-implement algorithm for *model assessment* of supervised machine-learning models (Efron & Gong, 1983, and next section) and *model selection* (Arlot & Celisse, 2010). In its basic form, CV is based on resampling the data without paying attention to any possible dependence structure, which may arise from, e.g., grouped or structured data, or underlying environmental processes inducing some sort of spatial coherence at the landscape scale. In treating dependent observations as independent, or ignoring autocorrelation, CV test samples may in fact be heavily correlated with, or even pseudo-replicates of, the data used for training the model, which introduces a potentially severe bias in assessing the transferability of flexible machine-learning (ML) models.

This CV bias is well-known in spatial as well as non-spatial prediction (Arlot & Celisse, 2010; Brenning, 2005; Brenning & Lausen, 2008; Roberts et al., 2017) and in forecasting (Bergmeir et al., 2018). It is most easily understood from a predictive modeling perspective by focusing on the question of where (and when) the model should be used for prediction. In crop classification from remotely-sensed data, for instance, learning samples routinely contain multiple grid cells from a sample of fields with known crop type, for instance 2000 grid cells from 100 fields scattered across a large study region. The purpose of training a model on this particular sample is to make predictions on other, new fields within the same geographic domain (Brenning, 2005, *intra-domain* prediction) — not *within* the same field, which obviously presents only a single crop type that is already known from the training sample. In this specific situation it would therefore seem rather unwise to train a model on a simple random subsample of grid cells, and to test it on the remaining data, using other grid cells from the same fields, as if one wanted to predict within a field. The results from this performance assessment would be over-optimistic, and perhaps badly so. To mimic the predictive situation for which the model is trained, one would rather have to resample at the level of fields, not grid cells (Peña & Brenning, 2015). If the model was to be applied to adjacent agricultural regions, i.e., outside the learning sample's spatial domain (Brenning, 2005, *extra-domain* prediction), it would even seem necessary to resample at a higher level of spatial aggregation, i.e. at the level of subregions within the learning sample, in order to realistically mimic the actual prediction task. The CV resampling needed therefore depends as much on the prediction task itself as on the data structure or dependency at hand.

While it is not the purpose of this article to recommend specific resampling schemes for specific use cases, the example from above may suffice to motivate the use of appropriate spatial and spatiotemporal cross-validation techniques, and the need for a unified framework and computational toolbox that accommodate a variety of prediction tasks that may be applicable to a broad range of application scenarios. `mlr3spatiotempcv` is such a toolbox.

This toolbox, implemented as an open-source R package, builds upon and generalizes several existing toolboxes that have been developed in recent years for more specific settings (Table 4.1). The earliest and most comprehensive of these implementations is the `sperrorest` `R` package (Brenning et al., 2012), which provides an extensible framework and includes predefined resampling strategies based on geometric blocking, clustering, and buffering. In contrast, packages `blockCV` and `ENMeval` were developed for block and buffer resampling with a focus on species distribution modeling (Muscarella et al., 2014; Rest et al., 2014; Valavi et al., 2019). However, neither of these have been integrated into established machine-learning frameworks such as `mlr`/`mlr3` (Lang et al., 2019) or `caret`/`tidymodels` (Kuhn & Wickham, 2020), and all of them lack support for temporal prediction tasks. The `CAST` package, in contrast, focuses on spatiotemporal prediction tasks and makes use of some functions of the `caret` framework (Meyer, 2020; Meyer et al., 2018). One limitation of all these packages is the sole focus on model assessment, while the proposed implementation within the `mlr3` framework also offers seamless integration into model selection and provides parallel execution and enhanced logging abilities. It is worth noting that a SCV library named `spacv` has recently been developed for Python3, which can be used with the `scikit-learn` machine-learning framework (Pedregosa et al., 2011).

Thus, `mlr3spatiotempcv` implements for the first time a comprehensive state-of-the-art compilation of spatial and spatiotemporal partitioning schemes that is well-integrated into a comprehensive machine-learning framework in R, the `mlr3` ecosystem. This package is furthermore equipped with a variety of two- and three-dimensional visualization capabilities. The hope is that this implementation will simplify and

facilitate reproducible geospatial modeling and code-sharing across a broad range of application domains.

The purpose of this article is to give an overview of the methods implemented in the `R` package `mlr3spatiotempcv`. After presenting the conceptual background in the following section, the overall structure of the `mlr3spatiotempcv` package is outlined. Next, various spatial and spatiotemporal partitioning techniques are contrasted and compared, before their application is demonstrated in a machine-learning model assessment in the following section. Finally, recommendations for the selection of suitable resampling techniques are given.

4.2 Spatial and spatiotemporal CV

In CV for predictive model assessment, the following formal setting is considered. The interest is in predicting a numerical or categorical response y of an object or instance using a feature vector $\mathbf{x} = (x^{(1)}, \ldots, x^{(p)})^t \in \mathbb{R}^p$ and a model $\hat{f}_{\mathcal{L}}$ that has been trained on a learning sample $\mathcal{L} = \{(y_i, \mathbf{x}_i),\ i = 1, ..., n\}$. The goal is to estimate the expected value of the performance of $\hat{f}_{\mathcal{L}}$,

$$perf(\hat{f}_{\mathcal{L}}) := E(l(Y, \hat{f}_{\mathcal{L}}(X))),$$

where l is a real-valued loss function, and the expected value is with respect to the probability distribution of X, the features of an instance (Y, X) drawn randomly from the underlying population. This is referred to as the *actual* or *conditional* performance measure, as it is conditional on $\mathcal{L}$ (Hand, 1997). The loss function can take a variety of forms such as the misclassification error $I(Y \neq \hat{f}_{\mathcal{L}}(X))$ in classification, or the squared error $(Y - \hat{f}_{\mathcal{L}}(X))^2$ in regression, among many other possible measures. The choice of the performance measure is equally critical as the choice of the estimation procedure, but it is beyond the scope of this contribution to discuss performance measures for regression and classification (see, e.g., (Hand, 1997) for classification, and (Hyndman & Koehler, 2006) for regression and forecasting tasks).

Since there is only a sample $\mathcal{T}$ of test data drawn from the population, one can only *estimate* the conditional performance of $\hat{f}_{\mathcal{L}}$:

$$\widehat{perf}_{\mathcal{T}}(\hat{f}_{\mathcal{L}}) = \frac{1}{|\mathcal{T}|} \sum_{(Y,X)\in\mathcal{T}} l(Y, \hat{f}_{\mathcal{L}}(X)).$$

This representation as a point estimator of $perf(\hat{f}_{\mathcal{L}})$ underlines the importance of using a random sample for model assessment to avoid estimation bias. Other estimators than the simple mean may be required when $\mathcal{T}$ is not a simple random sample, for instance a stratified random sample (Thompson, 2012). As always, judgment sampling may lead to uncontrollable bias.

Since re-using the learning sample $\mathcal{L}$ for testing, i.e. $\mathcal{T} := \mathcal{L}$, would yield the over-optimistic *resubstitution* or *apparent* performance, CV partitions the sample $\mathcal{L}$ into disjoint training and test sets. Specifically, $\mathcal{L}$ is split into k partitions,

$$\mathcal{L} = \mathcal{L}_1 \cup \ldots \cup \mathcal{L}_k, \qquad \mathcal{L}_i \cap \mathcal{L}_j = \varnothing \quad \text{for all } i \neq j,$$

and a model $\hat{f}_{(i)}$ is fitted on $\mathcal{L}_{(i)} := \mathcal{L} \setminus \mathcal{L}_i$, while $\mathcal{L}_i$ is withheld for testing. This is repeated for $i = 1, \ldots, k$ in order to effectively use the entire sample for testing, while keeping training and test sets disjoint at all times. The k-fold CV estimator can therefore be written as

$$\widehat{perf}_{\mathcal{L},CV}(f) := \frac{1}{k} \sum_{i=1}^{k} \widehat{perf}_{\mathcal{L}_i}(\hat{f}_{\mathcal{L}_{(i)}}),$$

where f is a ML algorithm, i.e. a mapping that trains a model $\hat{f}_{\mathcal{S}}$ using any suitable training sample $\mathcal{S}$. The use of $k = 5$ or $k = 10$ folds is most commonly seen in practice, and these preferences are also supported by theory (Bengio & Grandvalet, 2004; Cawley & Talbot, 2010). The k-fold CV estimator of model performance is a nearly unbiased estimator of the conditional performance measure when the observations were drawn independently (Efron & Gong, 1983). Since $\widehat{perf}_{\mathcal{L},CV}(f)$ still depends on the particular partitioning chosen for $\mathcal{L}$, it is sometimes recommended to repeat the estimation using different random partitionings (r-repeated k-fold cross-validation) to reduce the influence of randomness when creating partitions (Vanwinckelen & Blockeel, 2012).

In traditional CV, the partitioning is based on uniform random resampling, which ignores spatial or temporal autocorrelation or any existing grouping structure as well as the structure of the prediction task, and may result in over-optimistic performance estimates. Several approaches have therefore been proposed in the literature and implemented in software to accommodate a variety of predictive situations (Table 4.1).

Approaches based on *spatial blocking* (or sometimes called *grouping*) require either the construction of spatial zones, or the use of pre-existing spatial structures in the data. Let's refer to these spatial units or blocks as $\mathcal{Z}_i$, $1 \leq i \leq n_z$. These blocks are often constructed to serve as the $k = n_z$ spatial partitions, for example by performing k-means clustering of the sample coordinates (Ruß & Kruse, 2010), which we refer to as *coordinate-based clustering*; or generating the desired number of rectangular blocks as an example of *geometric partitioning*. The blocks may also be defined by a modeler based on an arbitrary partitioning of the study region based on an external data source, which we refer to as *custom resampling*. This often used when the data is grouped. For example, when using to multi-level sampling designs or studying spatial objects, it has been proposed to apply LOO at the site level (Kasurak et al., 2011; Martin et al., 2008) or, in animal movement studies, at the animal level (Anderson et al., 2005). We will broadly refer to such groups of observations as 'blocks' in a generic sense, regardless of the shape or origin of the groups. Also, data can be partitioned in feature space instead of geographic space, which has been referred to as "environmental blocking" (Roberts et al., 2017).

When n_z is much larger than the desired number of folds, k, then a partitioning can be applied to the zones themselves. In this case, the zone indices $1, \ldots, n_z$ are grouped into k equally sized subsets $\mathcal{I}_1, \ldots, \mathcal{I}_k$. This approach has been applied, for example, in spatial CV at the agricultural field level (Peña & Brenning, 2015). We would like to emphasize the conceptual distinction between *CV at the block level*, referring to this scenario, and *leave-one-block-out CV*, where the blocks themselves define the CV partitions. Figure 4.1 gives an overview of the conceptual framework and terminology used in this work.

One variant of CV is leave-one-out (LOO) CV, which has long been established in geostatistics (Cressie, 1993), sometimes with a focus on the spatial distribution of LOO error (Willmott & Matsuura, 2006). Although this is just a special case of non-spatial CV with $k = n$, it is sometimes also referred to as spatial CV (Willmott & Matsuura, 2006).

Spatial variants of CV have been proposed that apply an exclusion *buffer* or guard zone to the test locations to separate them from the training data (Brenning, 2005; Roberts et al., 2017). One approach that has been proposed for defining a separation distance is to use the range of autocorrelation of model residuals to determine the buffer distance, as this seeks to establish independence conditional on the predictors (Brenning, 2005; Roberts et al., 2017).

It should be noted that k-fold CV with a large value of k, and LOO CV in particular ($k = n$), is not only very time-consuming since the model has to be trained k times; these models will also be nearly identical

since only a tiny fraction of the data is withheld, and therefore estimation bias increases. 'Pure' LOO CV is therefore not recommended for machine-learning model assessment.

In the purely temporal domain, a special case is to leave out temporal observational units (or time slices; leave-time-out or LTO CV), as in leave-one-year-out CV (Anderson et al., 2005; Brenning, 2005). CV and hold-out validation strategies for time series have been discussed more extensively in the forecasting literature, considering also the effects of serial autocorrelation (Bergmeir et al., 2018); these methods are not the focus of the implementation presented in this work.

Turning to prediction tasks with spatiotemporal data, various spatial, temporal, or spatiotemporal partitioning strategies are being used, depending on the specific study objectives. While the former two ignore the temporal and spatial dimension of the data, respectively, it has also been proposed to leave out random subsets of locations and time points (Meyer et al., 2018) or spatiotemporal clusters (Y. Zhao & Karypis, 2002). Details of these and other implementations are outlined in the respective subsections of section 4.4.

4.3 mlr3spatiotempcv within the mlr3 ecosystem

With the increased awareness of the importance of spatial and spatiotemporal resampling strategies and the growing popularity of R in environmental modeling and geocomputation, it is important to equip ML frameworks such as `mlr3` with suitable algorithms. In this context, the `mlr3` ecosystem stands out as a unified, object-oriented and extensible framework designed to accommodate numerous ML tasks with a variety of learners, feature and model selection tools, and model assessment capabilities (Becker et al., 2020; Lang et al., 2019). All of these are supported by advanced visualization tools, which are particularly important in a spatial and spatiotemporal setting. Additionally, `mlr3pipelines` (Binder et al., 2021) provides a plethora of preprocessing operators to conveniently build ML pipelines which can be resampled, tuned and benchmarked as regular learners.

With its integrative approach and its aim to provide long-term support, `mlr3` overcomes the challenges of combining multiple specialized packages with poorly standardized interfaces. Issues that practitioners often face include varying argument lists of learners, different return values of `predict()` methods, and support for only specific feature types. These challenges result in substantial overhead and possible reproducibility issues, which are exacerbated by asynchronous development timelines of different components of the used ML pipelines.

Within the `mlr3` ecosystem, partitioning strategies are represented by their own objects of class `Resampling`, most of which are available within `mlr3` itself (e.g., random CV); other specialized strategies are defined in extension packages such as `mlr3spatiotempcv`. In the ML pipeline, these objects define the data splits used for model assessment and selection (hyperparameter tuning) by ML algorithms. Spatial and spatiotemporal partitioning techniques in `mlr3spatiotempcv` are currently mostly imported and interfaced from other packages, in particular `sperrorest`, `blockCV` and `CAST` (Brenning et al., 2012; Meyer, 2020; Valavi et al., 2019), in order to expose them to `mlr3` functionality. To reduce dependencies, some methods were re-implemented instead of importing them from the respective upstream packages.

Resampling objects in `mlr3spatiotmpcv` inherit from class `mlr3::Resampling` and can be created from established object classes for geospatial data in R, including simple features (E. Pebesma, 2018), which facilitates their integration into domain-specific workflows in the geospatial sciences. Support for projected (planar) and unprojected (geographic) coordinate reference systems (CRS) currently varies de-

pending on the partitioning techniques used, since these inherit their behavior from the underlying upstream packages.

Partitioning objects in `mlr3spatiotempcv` are equipped with generic `plot()` and `autoplot()` methods to visualize the created partitions. `autoplot()` is `ggplot2`-based and uses ggplot2 (Wickham, 2009) in two-dimensional geographic space and plotly (Sievert, 2020) in the three dimensional case, i.e., geographic space plus time.

While `mlr3spatiotempcv` solely focuses on spatiotemporal resampling methods and their visualization, other packages such as `mlr3spatial` or `mlr3temporal` are planned in the `mlr3` ecosystem to provide dedicated spatiotemporal learner and prediction methods.

From a user perspective, this package structure results in the following workflow for model assessment with `mlr3spatiotempcv` within `mlr3`: After choosing a ML algorithm that is supported by `mlr3` and setting up a learner object, users need to select hyperparameters that should be tuned and specify these in a `paradox::ParamSet`. Next, a suitable resampling scheme available within `mlr3spatiotempcv` is selected that mimics the spatial and/or temporal structure of the prediction task, such as spatial extrapolation, or forecasting of spatial time series. This information is used to create a `Resampling` object which is used within a (nested) CV to estimate the model performance. When using nested CV, the resampling schemes in the inner (tuning, `mlr3tuning::AutoTuner`) and outer loop (performance estimation, `resample()`) should be identical (Schratz et al., 2019). To evaluate the (nested) resampling, an adequate performance measure with respect to the response variable, such as the misclassification rate (classification) or the root-mean-square error (regression), must be selected and specified within `mlr3tuning::AutoTuner` and `resample$score()`. These choices now allow the user to execute the model assessment via either `resample()` (single model) or `benchmark()` (multiple models), and the results can be summarized visually (via `mlr3viz`) or in tabular form by accessing the respective fields of the returned `ResampleResult` object.

Additional examples and tutorial can be found in the mlr3book or the mlr3gallery.

4.4 Spatiotemporal partitioning methods and their implementation

At the most general level, resampling methods are categorized according to the level at which the data is partitioned and resampled (see Figure 4.1):

- **Spatial leave-one-out resampling**: Each individual observation forms a test set;
- **Leave-one-block-out CV**: Individual blocks are left out as test data, i.e. the number of folds equals the number of blocks;
- **CV at the block level**: Blocks are grouped into k partitions, each of which is used as a test fold.

In this context, a block can refer to an arbitrarily shaped spatial (or spatiotemporal) group of observations, not necessarily a rectangular region. A finer distinction can then be made by looking at how the blocks are derived:

- Using a geometry-based approach (rectangular or circular);
- Using an unsupervised clustering approach;
- Using a custom input, i.e. specifying the blocks with an external grouping variable.

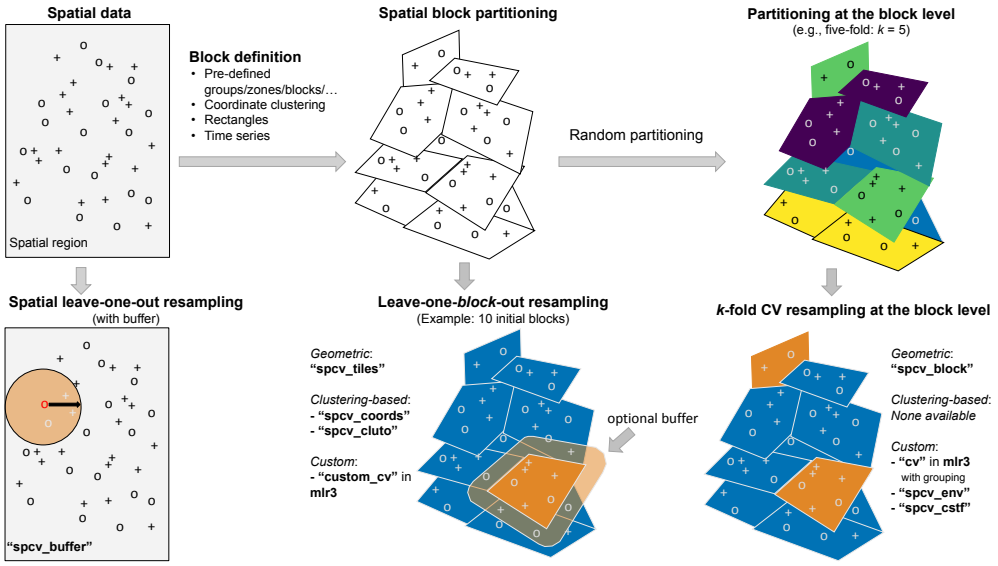


Figure 4.1: Conceptual overview of various spatial partitioning schemas. Starting from unpartioned spatial observations (top left) either a 'spatial block partitioning' or a 'spatial leave-one-out resampling' is applied in the first step. A spatial block partitioning can further be turned into a 'leave-one-block-out resampling' or a 'k-fold CV resampling at the block level'. The use of a buffer is theoretically possible in any scenario but in practice only offered by specific method implementations.

In some resampling schemes, separation buffers or guard zones can be imposed to separate the training and test data.

`mlr3spatiotempcv` currently implements the partitioning methods identified in Table 4.1. Several of the implemented algorithms are themselves versatile toolboxes with multiple options. Comprehensive and up-to-date information can be found in the package's online documentation (https://mlr3spatiotempcv.mlr-org.com). The following sections give an overview of most implemented partitioning strategies and their visualization options. The available methods are further discussed in section 4.6.

Users are encouraged to contribute new or missing spatiotemporal resampling methods directly to `mlr3spatiotempcv`. The already implemented methods can be inspected to get to know the class structure, active bindings and methods.

4.4.1 Spatial leave-one-out

Spatial leave-one-out methods use individual observations in space as test partitions and apply circular buffer or guard zones around around these test points to enforce a minimum prediction distance. Leave-one-disc-out resampling modifies this approach to leave out circular regions centered at observation points.

Spatial leave-one-out with buffer — `"spcv_buffer"`

Leave-one-out CV with buffer and several adaptations for species distribution modeling (Hijmans et al., 2020) are implemented in the `blockCV` package as the so-called "buffering" method and integrated into `mlr3spatiotempcv` under the label `"spcv_buffer"`. In species distribution modeling, the response

Type	Sub-type	Name	R package	References
Spatial leave-one-out	single point, with buffer	`"spcv_buffer"`	`blockCV` (2)	Ploton et al. (2020) Diesing (2020)
	disc, with buffer	`"spcv_disc"`	`sperrorest` (3)	Karasiak et al. (2021) Møller et al. (2021) Endicott et al. (2017)
Leave-one-block-out CV	clustering of coordinates	`"spcv_coords"`	`sperrorest` (6)	Morera et al. (2021) Geiß et al. (2017) T. Wu et al. (2020)
	geometric: rectangular	`"spcv_tiles"`	`sperrorest`	Bebber and Butt (2017) Zurell et al. (2020) Brenning et al. (2015)
	custom	`"custom_cv"`	`mlr3` (0)	-
CV at the block level	geometric: rectangular	`"spcv_block"`	`blockCV` (28)	Jensen et al. (2021) Escobar et al. (2021) Stewart et al. (2021)
	custom	`"cv"` with grouping	`mlr3` (0)	-
	clustering in feature space	`"spcv_env"`	`blockCV` (1)	Morera et al. (2021)
Spatiotemporal CV	custom	`"sptcv_cstf"`	`CAST` (6)	J. Gao et al. (2019) Reitz et al. (2021) Egli and Höpke (2020)
	clustering: custom	`"sptcv_cluto"`	`skmeans` (0)	-

Table 4.1: Article 3: Available spatiotemporal resampling methods in the `mlr3` ecosystem. The "Name" column shows the `mlr3` method name as found in the `mlr3::mlr_resamplings` dictionary. The count in brackets after the package name represents the number of studies that were found having used this resampling technique until May 2021. For each method, up to three randomly selected references were added to the table.

variable can either be recorded as presence/absence data or as presence/background information; both options are supported by this implementation. By default, the dataset contains confirmed presence and confirmed absence observations, i.e. locations where a species was observed and not observed, respectively, and therefore spatial LOO CV in its usual sense can be carried out. Figure 4.2 shows the first test fold generated with this method for presence/absence data with a buffer distance of 1000 m.

```
1
2  library("mlr3")
3  library("mlr3spatiotempcv")
4  task = tsk("ecuador")
5  rsmp_buffer = rsmp("spcv_buffer", theRange = 1000)
6
7  autoplot(rsmp_buffer, size = 0.8, task = task, fold_id = 1)
8
```

```
<ResamplingSpCVBuffewith 0 iterations
* Instantiated: FALSE
* Parameters: theRange=1000

```

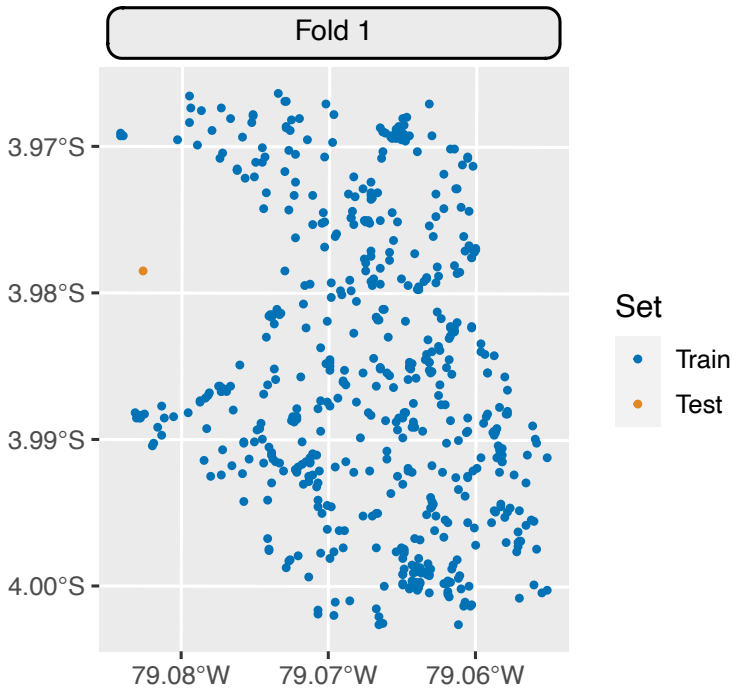


Figure 4.2: Visualization of the spatial buffering method from package blockCV (method "spcv_buffer" in mlr3spatiotempcv). The buffer distance is 1000 m.

In the presence/background (or presence-only) situation, in contrast, only presence observations are recorded, and all other locations within the study area are referred to as background and considered as pseudo-absences. Presence/background modeling can be enabled with the argument `spDataType = "PB"`. In this situation, the method constructs test folds that are centered at the recorded presence locations, offering two different modes of operation. With `addBG = TRUE` (the default), all background points with a distance of `theRange` around a test (presence) point are included in the test fold as absence data; note that in this case, there is no separation buffer between training and test samples. The `addBG = FALSE` setting, in contrast, for which no background data is added to the test fold, then contains only one (presence) observation, and only the data at a distance of `theRange` or greater are included in the training sample, including background data from these areas.

The application of LOO methods can be computationally expensive since the method cycles through the entire dataset and fits one model for each test fold.

Leave-one-disc-out with optional buffer — `"spcv_disc"`

Leave-one-disc-out resampling from package `sperrorest` defines circular test sets that are centered at sample locations, and optionally excludes a buffer zone from the remaining training data. It thus ensures that a minimum separation distance between training and test data is maintained. The number of discs is specified by the `folds` argument, which defaults to the sample size n. Sample locations are selected

randomly when `folds` is smaller than n; it is optionally possible to sample with replacement (`replace = TRUE`). Leave-one-disc-out resampling becomes spatial LOO CV for a radius of 0 m and when each observation is at a unique location.

It should be noted that the resampled discs will potentially overlap. Strictly speaking, this straightforward extension of spatial LOO does therefore not establish a disjoint partitioning as used for CV resampling in the traditional sense.

```
13
14  rsmp_disc = rsmp("spcv_disc", folds = 100, radius = 300L, buffer = 400L)
15  rsmp_disc
16
17  autoplot(rsmp_disc, size = 0.8, task = task, fold_id = 1)
18
19  <ResamplingSpCVDisc> with 100 iterations
20  * Instantiated: FALSE
21  * Parameters: folds=100, radius=300, buffer=400
```

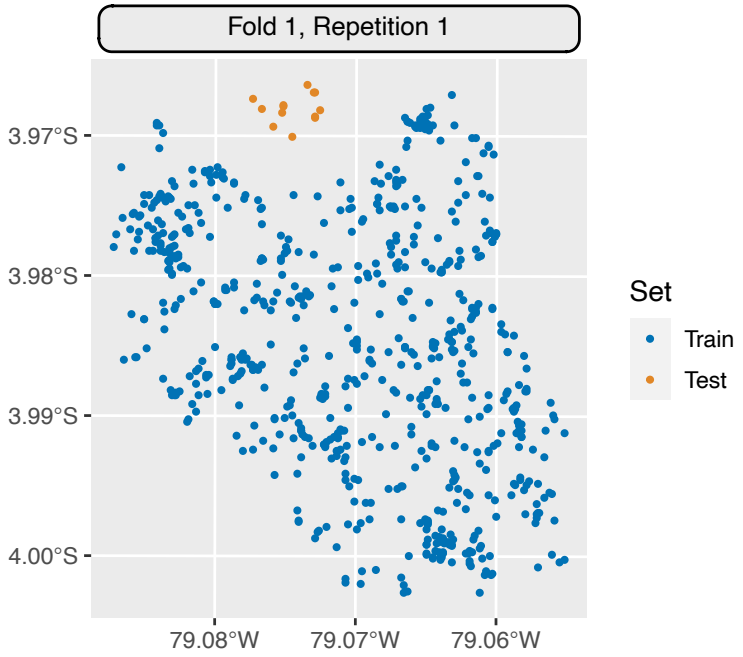


Figure 4.3: Visualization of one training set / test set combination generated with the leave-one-disc-out method from package `sperrorest` (method "spcv_disc" in mlr3spatiotempcv). The disc has a radius of 300 m and is surrounded by a 400-m buffer.

4.4.2 Leave-one-block-out cross-validation

Leave-one-block-out resampling methods partition the dataset spatially in order to use each of the resulting partitions as a CV test fold.

Clustering-based: using coordinates — `"spcv_coords"`

Cluster analysis provides a flexible approach to creating irregularly shaped spatial blocks for spatial resampling. Numerous techniques are available that can potentially be applied to the spatial coordinates of observations, to the features, or to a combination of both. In spatial model assessment, the focus has been on coordinate-based clustering, and specifically on leave-one-block-out resampling with blocks created by k-means clustering of the coordinates (Ruß & Kruse, 2010).

Coordinate-based clustering for spatial CV (Brenning et al., 2012; Ruß & Kruse, 2010) as implemented in package `sperrorest` uses the coordinates of all observations to create clusters in the spatial domain with the help of the k-means clustering algorithm. This can be regarded as a leave-one-block-out resampling method, or as a k-fold CV in which each test set is a spatial cluster. This method is referred to as `"spcv_coords"` in `mlr3spatiotempcv`.

The coordinate-based clustering approach is very versatile as it adapts to irregularly-shaped study areas and ensures that exactly k partitions are created, which are usually of very similar size when the sample locations are spread out evenly. Nevertheless, despite the random selection of initial cluster centers, repeated partitionings may in some cases be nearly identical. Also, k-means clustering may be less suitable for data sets with pre-existing clusters of points and/or with isolated, distant sample locations. When distinct clusters of points are present, as in multi-level sampling, it may be better to define clusters using a factor variable (see method `"custom_cv"` in section 4.4.2).

```
22
23  rsmp_coords = rsmp("spcv_coords", folds = 5)
24
25  autoplot(rsmp_coords, size = 0.8, fold_id = 1, task = task)
26
```

Geometric: using rectangular blocks — `"spcv_tiles"`

Leave-one-tile-out resampling is implemented in the `"spcv_tiles"` method imported from package `sperrorest`. It uses rectangular blocks that can be rotated (argument `rotation`), and a minimum number or fraction of observations per block can optionally be achieved by iteratively merging small blocks into adjacent blocks (argument `reassign` in conjunction with `min_n` or `min_frac`). Block size or number is specified via the argument `dsplit` or `nsplit`, respectively, and square blocks can be obtained with a single (or two identical) `dsplit` value(s).

Note that the actual number of folds obtained may be smaller than `nsplit[1]*nsplit[2]` (or smaller than what would be expected based on `dsplit`) since some blocks may be empty or (optionally) merged into adjacent folds. In the example, there are only eleven folds instead of twelve because the southwestern part of the study area's bounding box does not contain observations (Figure 4.5).

```
27
28  rsmp_tiles = rsmp("spcv_tiles", nsplit = c(3L, 4L))
29
```

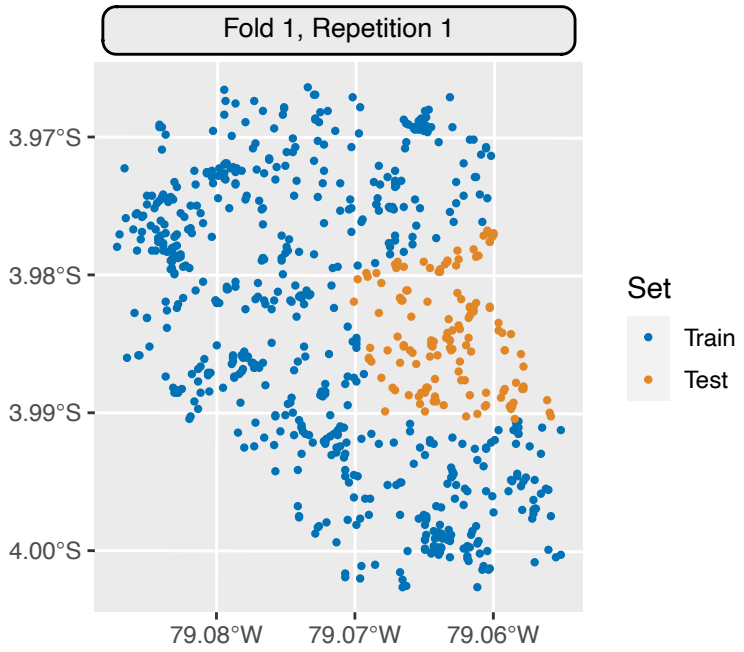


Figure 4.4: Leave-one-block-out CV based on k-means clustering of the coordinates as implemented in package `sperrorest` (method `"spcv_coords"` in `mlr3spatiotempcv`).

```
autoplot(rsmp_tiles, size = 0.8, fold_id = 1, task = task)
```

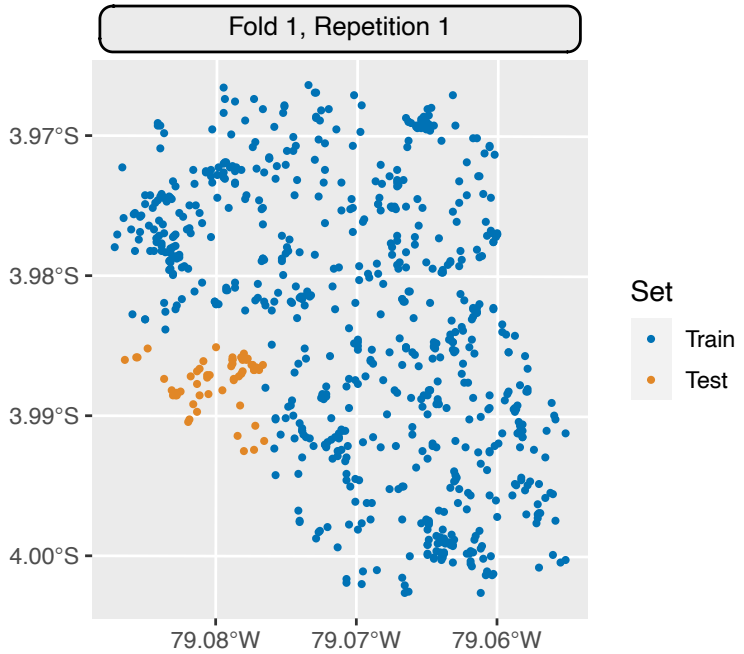


Figure 4.5: Leave-one-block-out resampling from package sperrorest (method "spcv_tiles" in package mlr3spatiotempcv with argument nsplit = c(3,4) indicating the number of rows and columns).

Custom: `"custom_cv"` in mlr3

Support for user-defined partitioning strategies is built into `mlr3` directly. In this so-called "Custom CV", users supply a factor variable, each level of which defines a partition. The factor variable can either be specified through a factor vector of the same length as number of observations, or by passing the name of a feature within the task (argument `col`). The following simple example (taken from `sperrorest::partition_factor()`) creates altitudinal zones that define the spatial partitions.

```
32
33  breaks = quantile(task$data()$dem, seq(0, 1, length = 6))
34  zclass = cut(task$data()$dem, breaks, include.lowest = TRUE)
35
36  rsmp_custom = rsmp("custom_cv")
37  rsmp_custom$instantiate(task, f = zclass)
38
39  autoplot(rsmp_custom, size = 0.8, task = task, fold_id = 1)
40
```

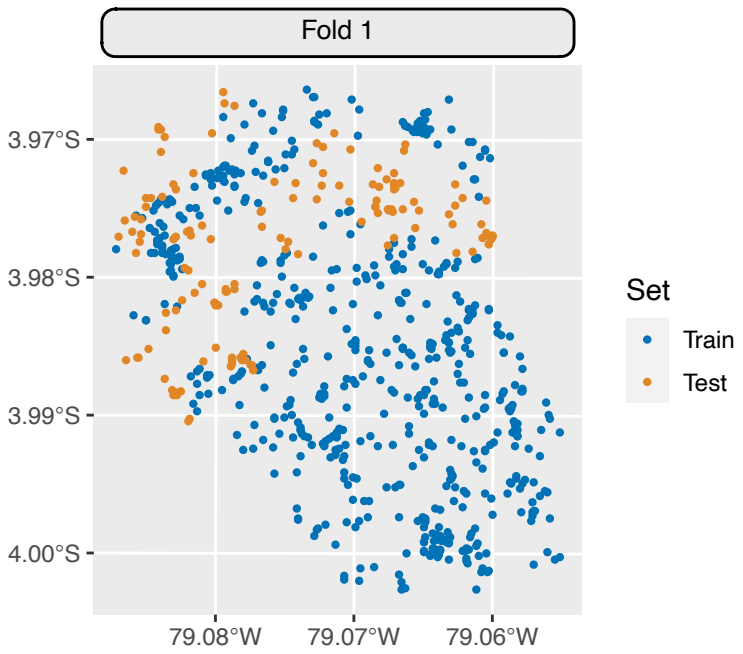


Figure 4.6: Leave-one-level-out (custom) resampling from package mlr3 (method "custom_cv"). A factor variable is used to define all partitions.

4.4.3 Cross-validation at the block level

Methods which operate at the block level first group the observations into blocks and then combine these blocks into CV partitions. In k-fold CV resampling at the block level, there are therefore k partitions, each consisting of $1/k$-th of the blocks. The special case in which k equals the number of blocks, CV at

the block level simply becomes leave-one-block-out CV, for which dedicated implementations exist (see subsection 4.4.2).

Geometric: using rectangular blocks — "spcv_block"

The "spcv_block" method from package blockCV supports both random and systematic resampling of square blocks with argument selection = "random" and "systematic", respectively; (see Figure 4.7 and Figure 4.8). There are additional options for modeling presence-only data, which is a typical use case in species distribution modeling. Users can furthermore supply a user-defined polygon via argument rasterLayer with predefined blocking zones.

The size of the square blocks (in meters) are determined by the range argument. Rectangular blocks can be created by specifying the number of desired rows and columns (arguments rows and cols). Due to the non-trivial specification of argument range package blockCV provides the helper functions spatialAutoRange() and rangeExplorer() to conduct a data-driven estimation of the distance at which the spatial autocorrelation within the data levels off (Valavi et al., 2019). According to the package authors, this estimate should then be used for argument range to have a sensible value for the block sizes created in method "spcv_block".

It should be noted that rectangular partitioning can be problematic in irregularly shaped study areas as shown in Figure 4.7 where some of the resulting partitions may contain substantially fewer observations than others.

```
41
42  rsmp_block_random = rsmp("spcv_block", range = 1000, folds = 5)
43
44  autoplot(rsmp_block_random, size = 0.8, fold_id = 1, task = task,
45    show_blocks = TRUE, show_labels = TRUE)
46
```

In systematic resampling, the blocks are numbered row by row, and blocks $i + j \cdot$ folds are assigned to fold i (see Figure 4.8). This may create undesired patterns when the number of columns is equal to or a multiple of the number of folds.

```
47
48  rsmp_block_systematic = rsmp("spcv_block",
49    range = 1000, folds = 5, selection = "systematic"
50  )
51
52  autoplot(rsmp_block_systematic, size = 0.8, fold_id = 1, task = task,
53    show_blocks = TRUE, show_labels = TRUE)
54
```

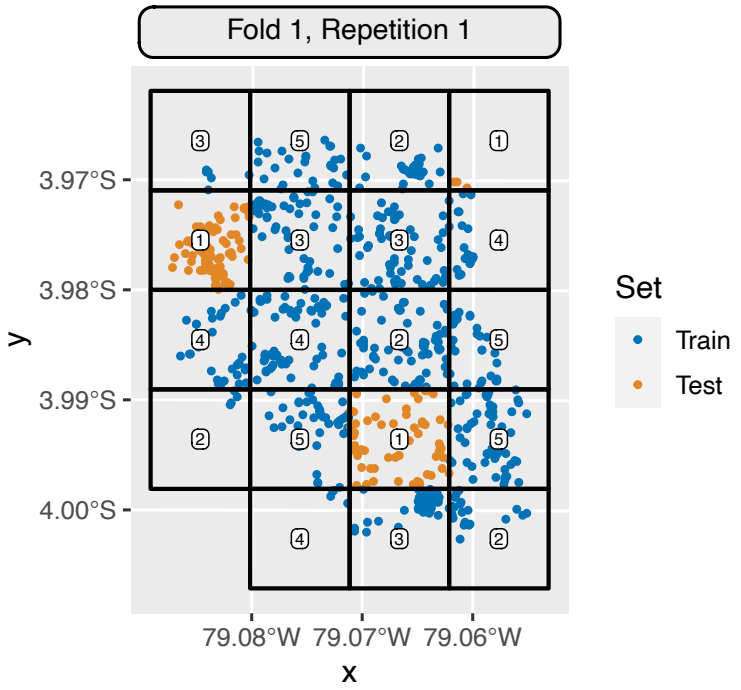


Figure 4.7: Random resampling of square spatial blocks using the implementation in package `blockCV` (method `"spcv_block"` with option `selection = "random"` in `mlr3spatiotempcv`). The size of the squares is 1000 m, and four out of the 19 blocks were assigned to the test partition.

Checkerboard partitioning is a special case of a systematic block partitioning (`selection = "checkerboard"`) which is why we omitted a practical example for this option. It inherently supports only two folds, making it less appealing than the more commonly used five- or ten-fold resampling, which achieve larger training set sizes.

Custom: `"cv"` with grouping in mlr3

Although the `"cv"` resampling strategy in `mlr3` performs random, non-spatial partitioning by default, it can also be used for CV at the block level. This is achieved by specifying the "group" column role in a mlr3 Task object, which uses the factor levels as blocks. A complete group or block of observations is therefore assigned to a specific partition, which consequently honors the grouping structure. In the deprecated `mlr` package this concept was referred to as "blocking".

In contrast to geometric or clustering-based blocks, the spatial or temporal location is not used explicitly, but rather implicitly through the spatial or spatiotemporal footprint of each user-defined block.

The following example uses *k*-means clustering to generate classes that are used as blocks. To underline the honoring of the groups, a number of groups (eight) that is not a multiple of the number of folds (three) was chosen. The test sets in the first and second folds are therefore composed of three groups while the third one holds two groups.

```
55
56  task_cv = tsk("ecuador")
57  group = as.factor(kmeans(task$coordinates(), 8)$cluster)
58  task_cv$cbind(data.frame("group" = group))
```

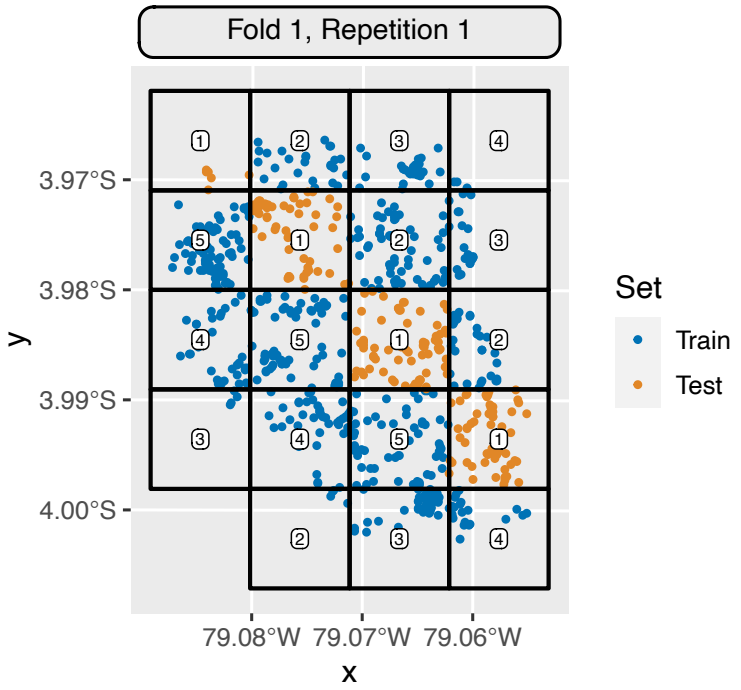


Figure 4.8: Sytematic resampling of square spatial blocks using the implementation in package `blockCV` (method `"spcv_block"` with option `selection = "systematic"` in `mlr3spatiotempcv`). The size of the squares is 1000 m, and four out of the 19 blocks were assigned to this test sample.

```
task_cv$set_col_roles("group", roles = "group")

rsmp_cv_group = rsmp("cv", folds = 3)$instantiate(task_cv)

print(rsmp_cv_group$instance)
```

```
> row_id fold
> 8      1
> 4      1
> 5      1
> 2      2
> 3      2
> 7      2
> 1      3
> 6      3
```

```
autoplot(rsmp_cv_group, size = 0.8, task = task_cv, fold_id = 1)
```

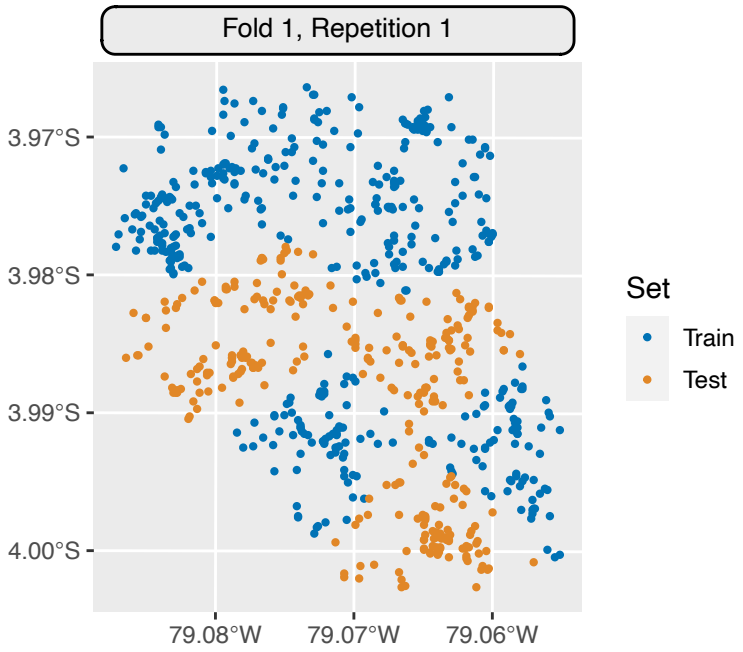


Figure 4.9: Cross-Validation at the block level including predefined groups from package `mlr3` (method `"cv"`). A factor variable is used to define the grouping. Each class is either assigned to the test or training set.

Clustering: using feature-based clustering — `"spcv_env"`

The last method from the `blockCV` package, referred to as "environmental blocking" (Roberts et al., 2017), makes use of *k-means* clustering (Hartigan & Wong, 1979) in a possibly multivariate space to define blocks for resampling at the block level. The user can select one or multiple numeric features via argument `feature` from which the clusters are created. Hereby, *k*-means will use Euclidean distance. To avoid a potential bias introduced by features with high variance when selecting multiple features, all features are standardized by default.

In the following example, the observations are clustered based on the feature "distance to forest" (left sub-figure of Figure 4.10), which results in a distance-based zonification. This method also allows to use multiple features for clustering. The right sub-figure of Figure 4.10 shows the outcome when using "distance to deforestation" and "slope angle".

```
78
79  rsmp_env = rsmp("spcv_env", features = "distdeforest", folds = 5)
80
81  rsmp_env_multi = rsmp("spcv_env", features = c("distdeforest", "slope"), folds = 5)
82
83  plot_env_single = autoplot(rsmp_env, size = 0.3, fold_id = 1, task = task) +
84
85    plot_env_multi = autoplot(rsmp_env_multi, size = 0.3, fold_id = 1, task = task)
86
87  library("patchwork")
```

```
plot_env_single + plot_env_multi
```

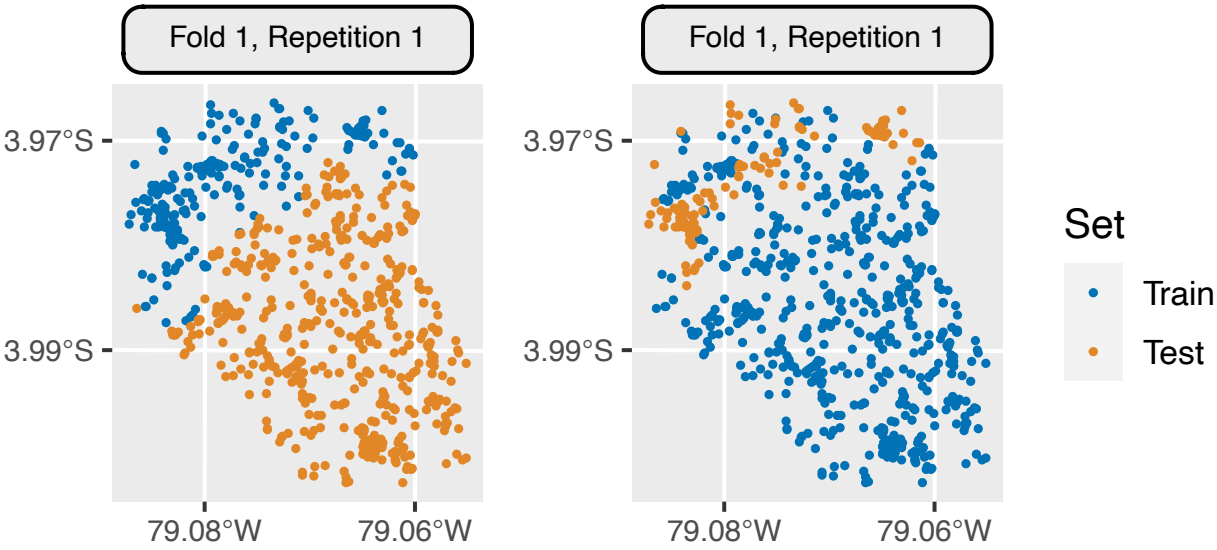


Figure 4.10: Environmental leave-one-block-out CV from package `blockCV` using one (left, "distdeforest") and two (right, "distdeforest" and "slope") predictors to define blocks in the feature space. Due to feature space clustering observations are not (necessarily) grouped in the spatial domain.

4.4.4 Cross-validation for spatiotemporal data

Some of the implemented resampling methods operate in multiple dimensions, i.e. in space, time, or space–time. In this section, only examples of these methods in the spatiotemporal domain will be shown. For their application in lower dimensions, usually only either the space or time coordinates need to be omitted from the user input.

Custom: "Leave-location-and-time-out" and related methods — `"spcv_cstf"`

(Meyer et al., 2018) proposed a spatiotemporal resampling method in which a test set is selected and all observations that correspond to the same location or time point are omitted from the training sample. This method is referred to as "leave-location-and-time-out" (LLTO) in package `CAST`. Additional methods that resample in the temporal and spatial domain only are named "leave-time-out" (LTO) and "leave-location-out" (LLO), respectively. Note that despite their names, LLTO, LTO and LLO are conceptually not leave-*one*-out methods as they place a certain fraction of observations in the test set, as in ordinary CV. Also, LTO and LLO are conceptually similar to `mlr3`'s "cv" method with a custom grouping as they perform a CV at the block level using a grouping structure defined by time points (LTO) and locations (i.e., time series; LLO).

In this section the `cookfarm` dataset is used as an example because it has a temporal dimension identified by the variable "Date".

`mlr3spatiotempcv::autoplot()` supports two visualization types for spatiotemporal methods which can be selected via the logical argument `plot3D`. The heavy lifting of the 3D visualization (i.e. 2D + time) option is done via package `plotly`. Because a dynamic image cannot be included in this manuscript,

static versions, which can be generated by setting `static_image = TRUE`, are shown (see for example Figure 4.11).

CV at the time-point level: "leave-time-out" (LTO) In the LTO method, the time points are resampled into the desired number of folds. In the terminology used in this work, this can be referred to as resampling at the level of time points, which effectively define blocks. Thus, observations from the same time point are jointly sampled into the same test (or training) fold, with no constraints on the temporal distance between the sampled time points. This method does therefore not implement block CV in the sense of the time series literature.

In the `cookfarm` example dataset, the `Date` variable was reduced to five unique levels for better visualization, and then used to create a spatiotemporal regression task in `mlr3spatiotempcv` (Figure 4.11). In `autoplot()`, a stratified sample based on the partitions is taken to reduce the number of points plotted.

```
90
91  data = cookfarm_mlr3
92  set.seed(42)
93  data$Date = sample(rep(c(
94    "2020-01-01", "2020-02-01", "2020-03-01", "2020-04-01",
95    "2020-05-01"), times = 1, each = 35768))
96  task_spt = as_task_regr_st(data,
97    id = "cookfarm", target = "PHIHOX",
98    coordinate_names = c("x", "y"), coords_as_features = FALSE,
99    crs = 26911)
100 task_spt$set_col_roles("Date", roles = "time")
101
102 rsmp_cstf_time = rsmp("sptcv_cstf", folds = 5)
103
104 autoplot(rsmp_cstf_time,
105   fold_id = 5, task = task_spt, plot3D = TRUE,
106   sample_fold_n = 3000L
107 )
108
```

CV at the location level: "leave-location-out" (LLO) In contrast to LTO, the LLO method randomly resamples locations that may, for example, correspond to time series. The sampled locations form the test partition while the temporal information is ignored (Figure 4.12). Unlike spatial CV methods that are based on geometric regions or the clustering of coordinates, the sampled test locations include no particular spatial relationship.

To tell the resampling method to use the 'space' column for partitioning, the 'time' column needs to be unset and the 'space' column defined. Because the temporal variable "Date" is not in use in this scenario, `autoplot()` needs to be instructed explicitily to use it for 3D plotting via argument `plot_time_var`.

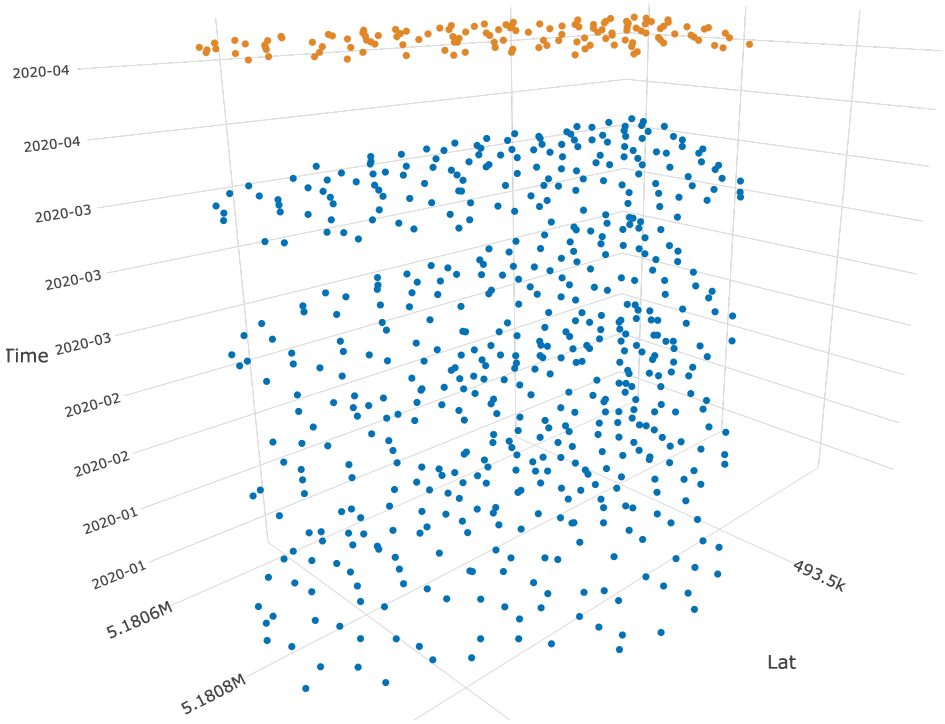


Figure 4.11: Perspective plot of `"leave-time-out"` CV from package `CAST` (method `"sptcv_cstf"` and column role `"time" = "Date"`). Only five folds and five time points were used in this example. Note that the blue dots correspond to five discrete time levels, which appear as a point cloud due to the static viewing angle.

```
109
110  task_spt$col_roles$time = character()
111  task_spt$set_col_roles("SOURCEID", roles = "space")
112
113  rsmp_cstf_loc = rsmp("sptcv_cstf", folds = 5)
114
115  autoplot(rsmp_cstf_loc,
116    fold_id = 5, task = task_spt,
117    plot3D = TRUE, plot_time_var = "Date",
118    sample_fold_n = 3000L)
119
```

"Leave-location-and-time-out" (LLTO) In LLTO, a test set is first randomly sampled from the data set, and then all observations that correspond to the same location or time point are omitted from the training sample (Figure 4.13). LLTO resampling mimics the situation where a model is trained on time series data from a number of locations and time points, and used to predict the time series at other locations and time points that are not included in the training sample.

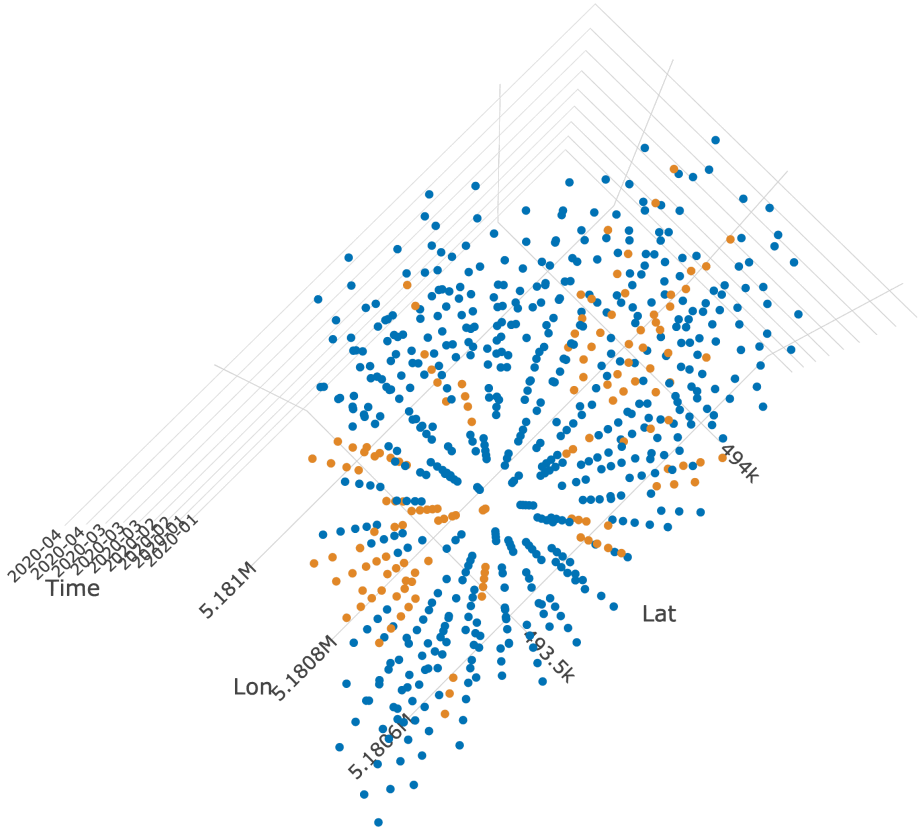


Figure 4.12: Birds-eye view of "leave-location-out" CV from package `CAST` (method `"sptcv_cstf"` and column role `space_var = "SOURCEID"`).

Conceptually, LLTO applies zero-distance buffering in both space and time: The buffer zones consist of all observations whose distance to the test sample in either space or time equals zero. In a mathematical sense, however, this buffering is not based on a valid metric (or distance function) in three-dimensional space (2D + time) as neither the identity of detectability nor the triangle inequality are satisfied by the underlying combined 'distance' measure. Also note that LLTO does not 'combine' LTO with LLO, as neither of these applies a buffer zone.

The `"spcv_cstf"` methods LLO and LTO (with only one of `space_var` or `time_var` set) require a variable in the dataset which should be used for grouping. The specification of the variable(s) which should be used for a spatial, temporal or spatiotemporal grouping is not trivial because the final partitioning should, in the optimal case, ensure that the selected groups inherit substantial autocorrelation within themselves and simultaneously differ substantially from other partitions. Also, if the selected variable contains too many groups, the difference within train/test splits may become undesirably high and tend towards a LOO CV (Meyer et al., 2018).

```
120
121   task_spt$set_col_roles("SOURCEID", roles = "space")
122   task_spt$set_col_roles("Date", roles = "time")
123
124   rsmp_cstf_time_loc = rsmp("sptcv_cstf", folds = 5)
```

```r

  autoplot(rsmp_cstf_time_loc,
    fold_id = 4, task = task_spt, plot3D = TRUE,
    show_omitted = TRUE, sample_fold_n = 3000L)

```

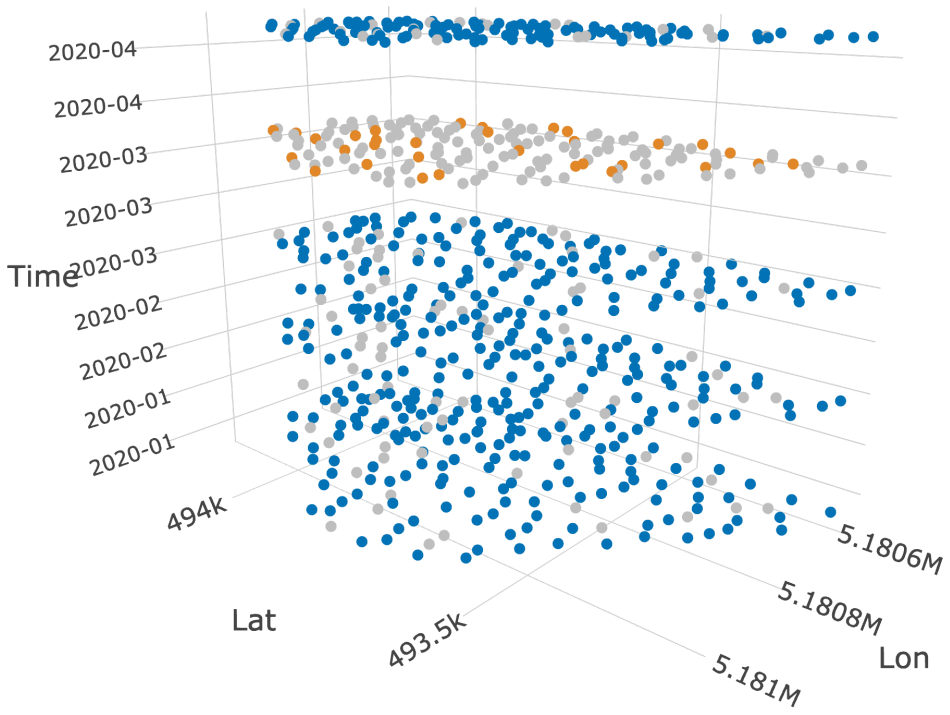


Figure 4.13: Perspective plot of "leave-location-and-time-out" CV from package `CAST` (method `"sptcv_cstf"` and column roles `"time" = "Date"` and `"space" = "SOURCEID"`). The grey points are excluded from both the training and the test set in this example.

Clustering: using CLUTO — `"sptcv_cluto"`

At present, `mlr3spatiotempcv` also supports spatiotemporal partitioning using the versatile CLUTO clustering algorithm (Y. Zhao & Karypis, 2002). CLUTO is available in R through the `skmeans` package, which provides an interface to a downloadable compiled library with a restriction to non-commercial uses (see `help("ResamplingSptCVCluto", package = "mlr3spatiotempcv")` for more information). Due to this restriction and the age of the latest release (14 years at the time of writing) this method is not explained in greater detail.

4.5 Step-by-step example: Comparing spatial and non-spatial CV

A well-known case study is used to demonstrate the application of spatial and non-spatial resampling techniques for model assessment in `mlr3spatiotempcv`. The objective of landslide susceptibility modeling

is to predict how prone to landslide initiation a location is. Models are fitted to historical landslide occurrences, but they need to learn generalizable relationships between predisposing variables and the response as opposed to perfectly reproducing or memorizing the historical distribution. This binary classification task on landslides in Ecuador (Muenchow et al., 2012) is available as a built-in task via `tsk("ecuador")`, but is generated from the learning sample in this example. Random forest is used as a classifier, and the area under the ROC curve (AUROC) as the performance measure.

Spatial CV is implemented in the form of leave-one-block-out CV using coordinate-based k-means clustering to generate irregularly shaped blocks of roughly equal size. This approach is better suited for the irregular shape of the present study area than a rectangular partitioning. Figure 4.14 and Figure 4.15 show the contrasting distributions of training and test samples. For demonstration purposes only four CV folds and two repetitions are used.

4.5.1 Task preparation

In `mlr3`, machine-learning tasks with their respective dataset and response variable are represented by objects of class `Task`. `mlr3spatiotempcv`'s spatial and spatiotemporal machine-learning tasks are also derived from this superclass. Specifically, the `TaskClassifST` and `TaskRegrST` classes for classification and regression tasks require several additional arguments that must be passed as a named list using the `extra_args` argument:

- `coordinate_names`: Names of the features that represent the spatial coordinates. This is automatically inferred when a `sf` object is passed.
- `coords_as_features`: Whether the coordinates should be used as features; by default they are not.
- `crs`: The coordinate reference system of the data as a PROJ string or EPSG code in the format `ESPG:<code>`.

At first all necessary R packages are loaded and a lower verbosity is set to keep the output tidy. A random-number seed is set for reproducibility.

```
130
131 library("mlr3")
132 library("mlr3spatiotempcv")
133
134 # be less verbose
135 lgr::get_logger("bbotk")$set_threshold("warn")
136 lgr::get_logger("mlr3")$set_threshold("warn")
137
138 set.seed(42)
139
```

The task `"ecuador"` is available as an example task in `mlr3spatiotempcv` through `tsk("ecuador")`. To create it manually from a `data.frame` named `ecuador`, one would do:

```
backend = mlr3::as_data_backend(ecuador)
task = TaskClassifST$new(
  id = "ecuador", backend = backend, target = "slides", positive = "TRUE",
  extra_args = list(
    coordinate_names = c("x", "y"), coords_as_features = FALSE,
    crs = "EPSG:32717")
)
```

4.5.2 Model preparation

Next, the random forest learner (`"classif.ranger"`) is initialized with default hyperparameters and the prediction type is set to `"probability"` because the model is used for soft classification. A set of commonly used learners is available in package `mlr3learners` (Lang et al., 2020), including the random forest implementation of (Wright & Ziegler, 2017).

```
library("mlr3learners")

learner = lrn("classif.ranger", predict_type = "prob")
```

4.5.3 Non-spatial cross-validation

To define a resampling strategy, the `rsmp()` function is used to generate a resampling object using four folds and two repetitions following a random sampling logic ("cv").

Next, the created resampling object `rsmp_nsp` is passed to the `resample()` function together with the task and learner objects created earlier to execute the model assessment. This is the actual, potentially time-consuming CV estimation. With the present settings, eight random forest classifiers are fitted and evaluated in this step — one model fitted on each CV training set.

Model performances are calculated from the CV predictions using the AUROC (`"classif.auc"` in `mlr3` notation).

```
rsmp_nsp = rsmp("repeated_cv", folds = 4, repeats = 2)
rsmp_nsp
rr_nsp = resample(
  task = task, learner = learner,
  resampling = rsmp_nsp
)
```

```
rr_nsp$aggregate(measures = msr("classif.auc"))

> classif.auc
> 0.7600664
```

4.5.4 Spatial cross-validation via coordinate-based clustering

The model assessment is now repeated again using spatial CV resampling, for which the only required change is to replace `"repeated_cv"` by `"repeated_spcv_coords"`.

```
rsmp_sp = rsmp("repeated_spcv_coords", folds = 4, repeats = 2)
rsmp_sp
rr_sp = resample(
  task = task, learner = learner,
  resampling = rsmp_sp
)

rr_sp$aggregate(measures = msr("classif.auc"))

> classif.auc
> 0.6100402
```

4.5.5 Visualization of CV partitions

Finally, we visualize (two of) the partitions that were used during performance estimation by making use of the generic `autoplot()` function in package `mlr3spatiotempcv` (Figure 4.14).

```
autoplot(rsmp_sp, task, fold_id = c(1:2), size = 0.8)
autoplot(rsmp_nsp, task, fold_id = c(1:2), size = 0.8)
```

4.5.6 Interpretation

If one takes a closer look at the results, the non-spatial CV estimate of AUC (0.76) is substantially higher compared to the spatial CV estimate of 0.64. Since test points in non-spatial CV may be from the same slopes or even the same landslides as the training data, the non-spatial CV result should/can be considered as an over-optimistic estimate of the model's ability to predict the susceptibility to "new" landslides.

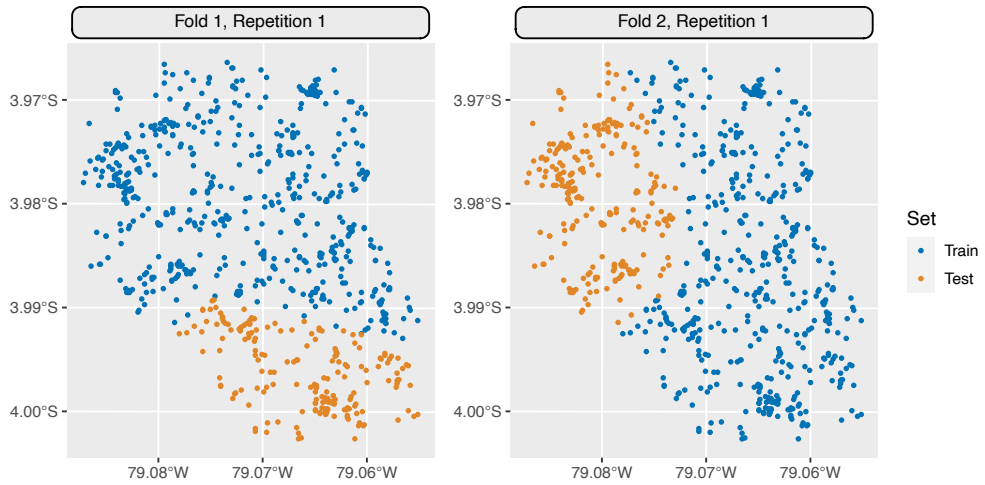


Figure 4.14: Spatial leave-one-block-out partitioning using coordinate-based clustering to create roughly equally sized polygonal blocks. Due to space limitations only the first two folds of the first repetition are shown.

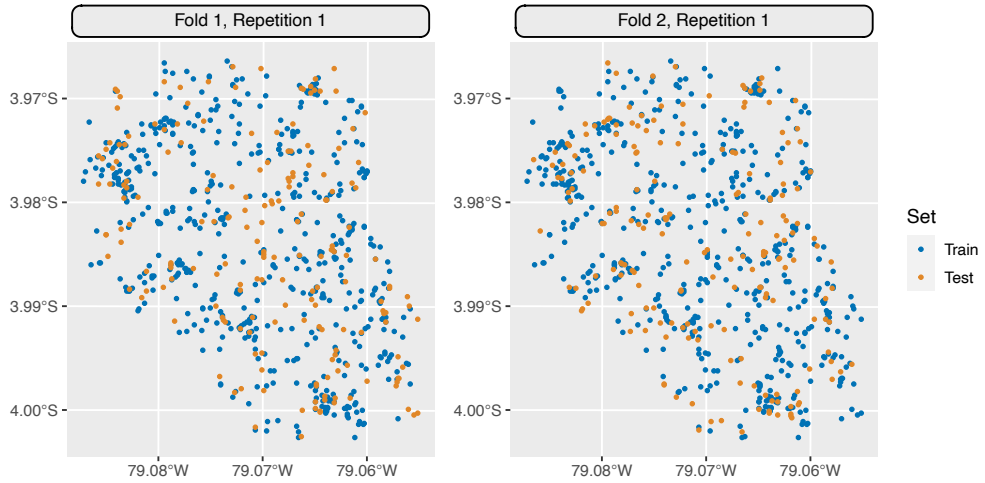


Figure 4.15: Random (non-spatial) four-fold CV partitioning. Only the first two folds of the first repetition are shown.

Spatial CV, in contrast, provides a better/more accurate measure of a model's ability to generalize from the training sample — in this case study, from the specific hillslopes and historical landslides in the training sample. It is also expected that spatial CV results better represent the model's transferability to geologically and topographically similar areas adjacent to the training area. The magnitude of the difference between spatial and non-spatial CV estimates may depend on the dataset, the strength on spatial or spatiotemporal autocorrelation, and the learner itself. Algorithms with a higher tendency to overfit to the training set will tend to have a larger spread in such scenarios.

4.6 Discussion

4.6.1 Choosing a resampling method for model assessment

The question of which resampling method should be chosen for a prediction task and dataset at hand comes up regularly in practice. Even though there is and most likely will be no definitive answer to this

question, we would like to give some guidance in this section to help find an appropriate method. As a general rule, we recommend to use a resampling scheme that (1) mimics the predictive situation in which the model will be applied operationally, and (2) is consistent with the structure of the data. Both aspects are outlined in this section, starting with two concrete modeling scenarios.

Although the case study example in section 4.5 used the `"spcv_coords"` method for coordinate-based clustering, this should not give the impression that this method is the only method suitable for this example task. In this application setting, we want to assess how well the model generalized from the concrete set of historical landslide occurrences, which is why we ensured that training and test sets contain different, "new" hillslopes and landslides. Coordinate-based clustering is particularly appealing in this setting because of its ability to adapt to the irregularly shaped study area of this example. Resampling at the level of sub-catchments could have been a viable alternative approach that can be implemented using custom resampling (`"custom_cv"` method); however, this may result in less balanced sizes of test sets as catchment sizes may vary. When the timing of landslides is known (event-based inventories) or multiple inventories have been compiled for different time points, it can also be recommendable to additionally sample training and test data from different time points, as with the LLTO and LTO (Meyer et al., 2018) or similar methods (Brenning, 2005).

In other application scenarios such as the crop classification example given in section 4.1 (Peña & Brenning, 2015), the objective is to predict the fruit-tree crop type of 'new', unseen fields within an agricultural region. In this use case we are not at all interested in "predicting" the already known crop type of (other) grid cells within the same agricultural field. Also, a model will likely be much better able to predict the crop type within the same field from multitemporal remote-sensing data since all crops within that field are subject to identical management practices (e.g., use of pesticides, pruning of fruit trees, tree spacing), while 'new' fields may be managed differently. As a consequence, grid cells from the same field should be grouped into a block, and resampling should be done at the field level (`"cv"` method with grouping) to receive an honest estimate of the model's performance in a relevant predictive situation. If, in contrast, the objective is to apply the model to an adjacent agricultural region (e.g., adjacent county) where the same crop types are present, it may be advisable to use coordinate-based clustering (`"spcv_coords"` method) to obtain larger, contiguous test regions.

In summary, there are various factors that may be considered in judging the suitability of a resampling method:

- Will the model be applied to predict 'new' outcomes at near or more distant spatial locations?
- Will it be applied to predict into the future, or hindcast gaps between spatiotemporal observations in the past?
- Is it necessary to impose a separation distance or prediction horizon as a spatial or temporal buffer between training and prediction locations?
- How densely are the observations distributed in space and time? Are they more densely distributed than the intended spatial or temporal prediction distance?
- Is the data naturally grouped, e.g. because of the spatial extent of the studied objects, or as a consequence of multi-level (cluster) sampling?
- With an eye on environmental blocking and extrapolation in feature space, is it intended to apply the model to predict 'new' outcomes for unobserved values of predictor variables?

Based on these criteria users may choose a matching resampling method that is either more restrictive (by discarding nearby observations for fold creation) or more liberal (by not removing observations and eventually ignoring natural grouping patterns). The specific publications related to the methods integrated into `mlr3spatiotempcv` may give further advice and provide additional use cases for the application of each respective approach. Users should therefore also refer to publications that are referenced of linked in the help files of this package or its respective upstream packages.

4.6.2 Resampling for hyperparameter tuning

CV is also widely used to assess model performance when tuning the hyperparameters of flexible machine-learning models, and this is also supported by the `mlr3` framework. Using the CV methods introduced here, `mlr3` can therefore be used to optimized models to show an improved performance in specific spatial or spatiotemporal predictive setting (Schratz et al., 2019). Such an optimization may, for example, result in a reduced maximum tree depth or increased minimum node size in the Ecuador case study, since these hyperparameter settings would result in a stronger generalization and reduced overfitting.

We recommend using nested CV for this purpose. In nested CV, an "inner" CV is performed on each CV training set, since hyperparameter tuning is an integral part of model fitting that should not be able to use information from the CV test set. In such scenarios it is recommended to use the same spatial resampling method for the inner CV (hyperparameter tuning) as for the outer CV (model assessment) in order to use the appropriate objective function for optimization. See (Schratz et al., 2019) for more details as well as chapter 11 of Geocomputation with R (Lovelace et al., 2019).

4.6.3 Additional practical issues

Since `mlr3spatiotempcv` harvests already implemented resampling methods from existing R packages, the broader overview presented in this work has highlighted that there are still several gaps that may need to be closed in the future, if specific use cases require those features.

For example, buffering, or the use of a spatial or temporal separation distance between training and test sets, is currently only implemented for some methods (`"spcv_buffer"`, `"spcv_disc"`, and `"sptcv_cstf"` with both `space_var` and `time_var`). Its use should, however, be limited to use cases involving a prediction distance, as a buffer zone reduces the size of the training sample and introduces the risk of geographically biased training data.

CV is often executed repeatedly to reduce the possible influence of random variability on CV estimates. In general, only methods that involve a random mechanism for generating or resampling blocks are suited for this. In leave-one-block-out CV, coordinate-based and environmental clustering (`"spcv_coords"`, `"spcv_env"` and `"sptcv_cluto"`) achieve this as their clusters are generated based on random seeds. However, experience with `"spcv_coords"` shows that clusters from repeated executions may in some situations be nearly identical to each other, resulting in very little variability between CV repetitions. While this effect also depends on the variable used for clustering, similar effects could potentially also apply to `"spcv_env"` and `"sptcv_cluto"` methods. However, such effects are more difficult to quantify because selected features of these methods are always different, in contrast to `"spcv_coords"` which always uses coordinates for clustering. This issue is even more critical in CV at the block level with `"spcv_block"` with options `selection = "systematic"` and `selection = "checkerboard"` be-

cause identical folds are assigned in each repetition. In contrast, `"spcv_block"` with option `selection = "random"` avoids this problem.

4.7 Conclusion and outlook

The `mlr3spatiotempcv` package is the first package to bundle and categorize spatiotemporal resampling methods implemented in multiple other packages in R. The available resampling techniques allow users to vary the scale or granularity of the resampled spatiotemporal units as well as their shape and possible buffer distance between training and test samples. These settings may account for the specific characteristics of spatiotemporal prediction tasks, but modelers now have to make the important decision of choosing a method that is adequate for their situation. They are advised to focus on the spatial or spatiotemporal structure of the model's prediction task, consider the structure of the learning sample at hand, and think about how the autocorrelation between training and test samples might affect their model assessment and selection.

The compilation of resampling techniques in `mlr3spatiotempcv` is by no means complete. Additional methods or parameters may therefore be added in the future as they become available in upstream package or are contributed directly to this package.

Spatiotemporal cross-validation as a paradigm is not yet fully established in scientific workflows, although it has been discussed intensively for more than a decade now. We anticipate that making the existing methods easily accessible to users is an important step to foster the acceptance of spatiotemporal cross-validation in the community and to allow modelers to produce bias-reduced model assessments in environmental and ecological studies.

```
R version 4.2.0 (2022-04-22)
Platform: aarch64-apple-darwin20 (64-bit)
Running under: macOS Monterey 12.4

Matrix products: default
BLAS:   /opt/R/4.2.0-arm64/Resources/lib/libRblas.0.dylib
LAPACK: /opt/R/4.2.0-arm64/Resources/lib/libRlapack.dylib

locale:
[1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8

attached base packages:
[1] stats     graphics  grDevices utils     datasets  methods   base

loaded via a namespace (and not attached):
 [1] pillar_1.7.0                compiler_4.2.0
 [3] mlr3learners_0.5.3          mlr3misc_0.10.0
 [5] rticles_0.23                tools_4.2.0
```

```
 [7] digest_0.6.29               uuid_1.1-0
 [9] tibble_3.1.7                evaluate_0.15
[11] lifecycle_1.0.1             checkmate_2.1.0
[13] gtable_0.3.0                pkgconfig_2.0.3
[15] rlang_1.0.2                 DBI_1.1.2
[17] cli_3.3.0                   yaml_2.3.5
[19] parallel_4.2.0              xfun_0.31
[21] fastmap_1.1.0               dplyr_1.0.9
[23] stringr_1.4.0               knitr_1.39
[25] generics_0.1.2              vctrs_0.4.1
[27] mlr3_0.13.3-9000            globals_0.15.0
[29] tidyselect_1.1.2            grid_4.2.0
[31] glue_1.6.2                  data.table_1.14.2
[33] listenv_0.8.0               R6_2.5.1
[35] fansi_1.0.3                 parallelly_1.32.0
[37] rmarkdown_2.14              purrr_0.3.4
[39] ggplot2_3.3.6               lgr_0.4.3
[41] magrittr_2.0.3              ellipsis_0.3.2
[43] backports_1.4.1             scales_1.2.0
[45] codetools_0.2-18            htmltools_0.5.2
[47] palmerpenguins_0.1.0        assertthat_0.2.1
[49] future_1.26.1               colorspace_2.0-3
[51] mlr3spatiotempcv_2.0.0.9001 utf8_1.2.2
[53] paradox_0.9.0.9000          stringi_1.7.6
[55] munsell_0.5.0               crayon_1.5.1
```

CHAPTER 5

DISCUSSION

The goal of this thesis was to advance methodological processes in the environmental/ecological modeling field. To reach this goal in situ data related to *D. sapinea* (chapter 2) and *F. circinatum* (chapter 3) infections in northern Spain were analyzed using various learning algorithms. State-of-the-art tuning and feature selection methods combined with variable importance analyses were used therefor. Emphasis was put on a bias-reduced assessment of model performance estimates using SCV. All studies were conducted in a reproducible manner and wrapped in individual research compendia. Generalized research software has been created in the form of an `R` package to allow for simplified reusability (chapter 4). The following sections primarily discuss the research questions outlined in section 1.2.

5.1 Analyzing pathogen infections using environmental & remote sensing data

5.1.1 Reflections on data availability and study setup

Data quality played an important role during the analyses. Examples are the in situ measurements of *D. sapinea*-infected trees (case study 1, chapter 2), the observed degree of defoliation caused by *F. circinatum* and the airborne hyperspectral remote sensing data (case study 2, chapter 3). Several components could have possibly influenced data quality in this work: the human error during data collection (MacLean & Lidstone, 1982), the quality of the remote sensing data (geometrical offset, atmospheric correction) and the temporal offset between the acquisitions of the in situ and remote sensing data. The latter is problematic when the data collection time interval overlaps with a seasonal change period, as it was the case in the second case study (chapter 3): here, the in situ data was collected during the first two weeks of September while the hyperspectral data was collected at the end of September/beginning of October (section 3.2.1). Observed trees, both infected and non-infected ones at the time of data collection, might have undergone changes until the remote sensing data was acquired about two and a half weeks later. The potential bias introduced by this offset is hard to quantify and might as well be negligibly small in the best-case scenario. Performing the survey during peak times of the vegetation period might have helped to more easily distinguish healthy from non-healthy trees and hence improved data quality (Morellato et al., 2010). Furthermore it might have been beneficial to analyze data, both in situ and remote sensing, from different seasons in the first place to average out phenology-related changes. This would have also helped to average out potential bias introduced by a time offset between in situ and other datasets (Morellato et al., 2010).

The following paragraph discusses reflections related to predictor selection and possible data source combinations. Both studies used very distinct feature sets to model the respective response variable: when *D. sapinea* was the response variable (chapter 2), only environmental predictors such as temperature, precipitation or soil type were used to model infected trees. In contrast, the second case study (chapter 3) only made use of feature sets derived from hyperspectral remote sensing data. The combined use of both

hyperspectral data and environmental variables in the respective studies would have been challenging. Due to the large study area in the first case study (chapter 2) (the Basque Country), no complete coverage of hyperspectral data would have been available. Multispectral data, which would have been available via the Sentinel satellite fleet, would have lacked the required geometric resolution needed to analyze individual trees adequately. In the second case study (chapter 3), the individual trees were located within plots spanning an area of 0.3 ha - 1.5 ha. For such area sizes there are small to no differences for variables like temperature, precipitation or soil type. Hence the inclusion of these variables would have mostly likely had no effect on the modeling outcome.

Other studies which modeled *D. sapinea* infections found that higher summer temperatures were responsible for increased tree damage while in areas with higher precipitation and lower temperatures the pathogen caused substantially lesser damages (Brodde et al., 2019). Studies analyzing *F. circinatum* also mainly focused on the main climatic variables temperature and precipitation (Möykkynen et al., 2015). In contrast to *D. sapinea*, temperature is less important for *F. circinatum* which prefers more humid regions (Iturritxa et al., 2014). Others also explored the potential of using remote sensing data to analyze pathogens. Poona and Ismail (2019) made use of narrow band indices to analyze *F. circinatum* related stress in *P. radiata* seedlings. These findings back the idea of modeling *D. sapinea* as a function of environmental predictors while aiming to improve model performance by adding yet unused predictors like soil type or PISR (chapter 2) (Iturritxa et al., 2014).

Most research in this work focused on analyzing and detecting the presence of these pathogens using environmental variables, for example temperature, precipitation or soil type. This is in line with various other research, e.g., Drenkhan et al. (2020); Hernandez-Escribano et al. (2018); Iturritxa et al. (2014); Wingfield et al. (2008). The spread of a pathogen, however, might also be initiated and boosted by non-environmental factors. Evira-Recuenco et al. (2015) demonstrated that *F. circinatum*, which is known to be a "seed-borne" pathogen, is able to infest cones on healthy branches in (hygienic) nurseries. These infected cones will then eventually be distributed into pine plots. The potential spread of *F. circinatum* via infected seeds has also been discussed further by Burgess and Wingfield (2002) and Storer et al. (1998). Bosso et al. (2017) found altitude (higher probability of presence at low altitudes) and land cover, i.e., different compositions of tree species in this context, to be important predictors for detecting *D. sapinea* presence in Italy. The authors state that *D. sapinea* was found to be primarily present in areas that are dominated by one of *P. pinea*, *P. pinaster*, *P. halepensis*, *P. sylvestris* or *P. nigra*.

5.1.2 A critical view on model performance

When looking at the model performance results of both studies (RMSE of 28 percentage points for the *F. circinatum* study (case study 2, chapter 3) and a Brier score of around 0.16 when analyzing *D. sapinea* (case study 1, chapter 3)) it becomes likely that (yet unknown) predictors could potentially help explain the existing relationships better and increase model performance. External events such as damages caused by hail storms can lead to entry points for pathogens. It has been shown that the inclusion of such information can improve model performances (Schratz, 2016). Yet data collection for such variables is very costly and often lacking coverage for specific time periods or areas. In addition, such information might be affected by high (human) observer errors.

The performance achieved by the fitted models during this work is rated as fair. Taking the overall effort into account that was put into both data collection and model fitting, it can be concluded that results

stayed below initial expectations. Yet, the overall idea of using learning algorithms to explain pathogen infections stays a promising one. The use of good quality data combined with the potential modeling has to offer (tuning, feature selection, variable importance, benchmarking different algorithms and the potential use of specialized species distribution models (e.g., Bosso et al., 2017)), might result in models that can be of practical help for local decision making, monitoring and early detection of pathogen spread.

5.1.3 The current and future state of *D. sapinea* and *F. circinatum*

Drenkhan et al. (2020) summarized the current state of *F. circinatum* dispersal on a global scale. According to the geodatabase provided by the authors, the fungus is present in 41 countries with the main dissemination regions being Mid-South America and southern Europe. Ganley et al. (2009) used a CLIMEX model, which is a biostatistics software to model the geographical distribution of a species, to visualize suitable living areas of *F. circinatum*. Matching these historical predictions from Ganley et al. (2009) with the current state of the disease distribution mapped by Drenkhan et al. (2020) shows a good fit. Watt et al. (2011) included different climate scenarios into suitable living area projections of *F. circinatum*. The authors found that possible habitat areas in Europe (and New Zealand) grew under all considered climate scenarios while in contrast the global suitable habitat area will most likely decrease between 39% to 58% (Watt et al., 2011). More specifically, Drenkhan et al. (2020) predicted an increase by 24% p.p. to 91% of the total European land up north until Denmark and the Netherlands with the main drivers being the rise in temperature and reduced drought stress. Möykkynen et al. (2015) also used CLIMEX to model the potential spread of *F. circinatum*. While the authors conclude with other studies that central European areas like France and Germany will turn into suitable habitats for the pathogen, they emphasize that the short flight distance of spores will most likely be a limiting factor for pathogen distribution. Human support would be needed to help the pathogen cross natural gaps of unsuitable habitats on its way spreading to Northern Europe.

The risk areas of *D. sapinea* infection, which favors dry and low precipitation areas, will likely increase globally due to ongoing global warming (Bosso et al., 2017; European Environment Agency, 2017). The European Environment Agency (2017) and Lorenzo and Alvarez (2020) predict a substantial decrease in precipitation and an increase in temperature for Spain and the Mediterranean region. The duration of dry periods for Southern Europe, which includes Spain and the Basque Country, will also likely increase (Castaño-Santamaría et al., 2019; Jiménez-Donaire et al., 2020; Lorenzo & Alvarez, 2020). One of the major threats to *D. sapinea* is the minimum temperature during winter time, which needs stay above 8–10 °C to not cause substantial damage to the pathogens population (Desprez-Loustau et al., 2007). Global warming though will lead to more regions not falling below this range, increasing the suitable habitat space of the pathogen.

Bosso et al. (2017) predicted an increase of suitable areas for *D. sapinea* in Italy between 9% to 40% using different climate scenarios. Mountainous areas are predicted to stay unsuitable for the pathogen due to their inconsistent climatic behavior. Yet other studies exist which predict the upcoming climatic changes to have only little impact: Fabre et al. (2011) concluded climatic changes in the next 40 years to be less impactful to the presence of *D. sapinea* than climatic changes which happened within the last 15 years as such already favored the pathogen considerably.

In summary, both pathogens *D. sapinea* and *F. circinatum* have spread substantially in Europe within the last decade and will likely continue to do so in a more rapid way in the upcoming decades. Most pro-

jections and studies conclude that climate change will have a significant positive influence on the increase of high-risk areas in the future. This applies specifically to Europe as suitable areas on a global scale will eventually even decrease (Watt et al., 2011).

5.2 Spatial partitioning methods in cross-validation

The effects of SAC in CV have been discussed intensively in recent years (Brenning, 2012; Geiß et al., 2017; Halvorsen et al., 2016; Mets et al., 2017; Ploton et al., 2020; Roberts et al., 2017; Ruß & Brenning, 2010b; Wadoux et al., 2021). The following sections discuss the contributions of this work to the ongoing debate.

5.2.1 Effects of using spatial cross-validation during hyperparameter tuning

In chapter 2 the effects of using spatial partitioning in different benchmarking scenarios is analyzed using the *k*-means based method after Brenning (2005). The focus was on evaluating differences in predictive performance when using spatial partitioning during the tuning stage of a nested CV setting compared to no hyperparameter tuning or hyperparameter tuning with non-spatial partitioning (Schratz et al., 2019). The motivation was based on existing research which showed substantial differences in performance when comparing spatial and non-spatial partitioning during CV (Brenning, 2012; Goetz et al., 2015; Roberts et al., 2017). So far no study analyzed the effect of spatial partitioning on hyperparameter optimization within the inner loop of a CV.

Hyperparameters differed greatly in many instances between the different partitioning setups (Figure 2.5). Yet the effects on predictive performance were small and non-substantial (Figure 2.6). Few notable model-specific differences were observed, e.g., SVM and BRT showed improvements when using a consistent "spatial/spatial" setup, i.e., using spatial partitioning for both performance estimation and hyperparameter tuning. No similar (positive trending) patterns were observed for unequal resampling pairs, for example "spatial/non-spatial".

The fact that models did not perform worse when using spatial partitioning for hyperparameter optimization should strengthen the motivation to use a consistent partitioning scheme for both stages of a nested CV. Yet, in cases where this is not feasible or technically possible, the use of an inconsistent schema does not seem to have a great (negative) effect on the results if spatial partitioning is at least used in the outer loop of the CV. While the study has seen quite some attraction citation-wise until now, only few follow-up studies used a consistent resampling scheme when conducting CV (e.g., Hengl et al., 2021; Morera et al., 2021; Petermann et al., 2021) until today. The majority of citing studies used the work as a reference to justify hyperparameter tuning in general without referring to nested SCV explicitly. One possible reason for the so far low adaption rate, besides the fact that the influence on predictive performance will likely be negligibly small, could be the lack of (simple) technical solutions to apply spatial partitioning methods in a nested CV scenario. Tackling this potential issue was part of the motivation for the creation of the R package `mlr3spatiotempcv` (chapter 4). Despite the slow adaption rate, Kopczewska (2021) concludes that that the use of (nested) SCV is on the way to become the standard partitioning method when dealing with spatial data.

5.2.2 (Bias)-differences in performance estimation between spatial and non-spatial CV

The main motivation for using spatial partitioning methods during CV is to create realistic train-prediction scenarios with the data at hand. Realistic in this case means to mimic the actual desired prediction scenario of the model in practice. The desired prediction area is often spatially distant from the region where the training data originated from, i.e., the model is performing an extrapolation task. If this is applied within CV, SAC is actively accounted for in the data because training and test sets become less similar to each other if the spatial distance between their respective observations increases. This separation is actually accounting for the underlying spatial autocorrelation as it results in less overoptimistic and biased-reduced performance estimates. In the last years the acceptance for using SCV to estimate bias-reduced model performances increased with many studies making active use of it, e.g. Meyer et al. (2018); Ploton et al. (2020); Roberts et al. (2017). While the knowledge to make use of spatial partitioning for biased-reduced CV estimates has been around even longer (Brenning, 2005), many researchers might have been unaware of it or might have simply missed the few available tools in commonly used programming languages to account for SAC during CV.

Yet recently, Wadoux et al. (2021) started a scientific debate discussing possible overpessimistic performance results based on the study of Ploton et al. (2020). Wadoux et al. (2021) criticizes the use of SCV in general and uses the study of Ploton et al. (2020) as an example to claim that the performances calculated by the use of SAC during CV are overly pessimistic and that design-based probability sampling should be used, if possible. According to Wadoux et al. (2021), the use of spatial resampling methods leads to misinterpretation of map accuracy, due to overly pessimistic results, and should be replaced by design-based inference instead, which is "model-free" and "design-unbiased." In addition the authors claim that "in probability sampling the validation data [to be] design independent" (Wadoux et al., 2021).

The general claim of SCV possibly being overpessimistic and leading to a higher bias than non-spatial CV is still unaddressed. However, the claim that only the use of design-based probability sampling leads to unbiased performance estimates applies solely to scenarios in which taking a probability sample is a feasible approach and data is not spatially clustered. If data is clustered, SCV does not show more bias than non-spatial CV, as shown by Wadoux et al. (2021) themselves in the "clustered random" example of their study (Fig. 1, setting (d)). Only (very) few environmental/ecological studies evaluate map accuracy on non-clustered datasets. It is agreed on that the use of design-based validation should be encouraged for studies that allow the use of probability samples. Yet the overall claims of Wadoux et al. (2021) of SCV not being the right tool to evaluate model performance is not backed by their study and needs more research.

5.2.3 Availability & application of spatial resampling methods

The term "availability" in this section refers to the existence of code packaged in modules or packages of a programming language. At the start of this thesis in 2016, the availability of spatial resampling implementations across programming languages was sparse. In R specifically, the only project which addressed the topic was the R package `sperrorest` (Brenning, 2012). `sperrorest` provides a flexible solution for applying various spatial resampling methods in combination with any modeling algorithm in R. Yet, no straightforward option to use the implemented resampling strategies with model optimization techniques was available. In addition, parallelization was not supported and (benchmark) results from different models needed to be combined manually. While the addition of parallelized resampling iterations was a byproduct of this thesis, adding support for flexible hyperparameter tuning was found to exceed the

scope of the package. Following the goal of conducting environmental benchmarking analyses as done in chapter 2 and chapter 3, the need for a more generic solution which supports all of the mentioned points was identified.

During this time, machine-learning frameworks such as the R package `caret` or `mlr` started maturing (Bischl et al., 2016; Kuhn, 2017). While their focus was on simplifying the application of machine learning in general, they were lacking support for niche modeling fields and their respective methods, e.g., spatial partitioning methods for environmental modeling applications. It was eventually decided to rather contribute spatial resampling methods to existing frameworks than implementing essential machine-learning building blocks, like hyperparameter tuning, into `sperrorest`. A possible side-motivation was to increase awareness of field-specific methods within the machine learning community due to the large outreach of these frameworks compared to a standalone R package. In addition, the available studies discussing the use non-spatial resampling being possibly suboptimal for spatial data at this time was yet small (Brenning et al., 2012). Last, the application of non-spatial resampling methods on spatial datasets was still widely accepted during peer reviews, often without any hint about the existence of spatial resampling methods.

While in the beginning, only one method from `sperrorest` was contributed into the `mlr` framework, it was quickly realized that a generic interface providing various spatial resampling methods is needed to effectively support the process of selecting and applying spatial resampling methods in a generic way. This finally lead to the creation of the R package `mlr3spatiotempcv` (chapter 4) which attempts to make the application of spatial partitioning methods easier by providing a consistent interface across different partitioning methods while at the same time providing full access to the functionality from the `mlr3` machine learning framework, e.g., the selection of various model optimization techniques. `mlr3spatiotempcv` is an extension of the `mlr3` machine learning framework, which itself is the successor of the `mlr` framework.

During the development of `mlr3spatiotempcv` more R packages providing single or multiple spatial resampling methods were created (`CAST`, `blockCV`) (Meyer, 2020; Valavi et al., 2019). Following the philosophy of building upon existing work, these implementations were bundled within `mlr3spatiotempcv`. Some methods offered by `mlr3spatiotempcv` (e.g., the classes `ResamplingSptCVCstf` and `ResamplingSptCVCluto` in `mlr3spatiotempcv`) also support spatiotemporal data. Yet their application, implementation and discussion are considered to be out of scope for this work. The task to account for two autocorrelation types, during model validation and elsewhere, comes with new issues and challenges (Di Cecco & Gouhier, 2018; Ives et al., 2021).

Similar toolboxes in other languages data-science focused programming languages (Python, Julia) were missing until recently. In Python, the modules `spacv` (Comber, 2020) and `Museo Toolbox` (Karasiak, 2020) provide spatial partitioning methods which can be used within the `scikit-learn` machine learning framework. Both were developed at roughly the same time as `mlr3spatiotempcv`. No similar projects were found for Julia at the time of writing.

The question which spatial resampling method to use comes up frequently in practice. Thus, the topic is addressed in more detail in subsection 4.6.1. While there is no straightforward answer, helpful pointers to simplify the decision-making process are provided. It was observed that the existence of tutorials which show how to use resampling methods together with advanced tuning and feature selection techniques, increased adoption by fellow researchers.

5.3 Using filter-based feature selection for high-dimensional data

The handling of high-dimensional data in modeling is a challenging problem which has been researched for decades (Donoho, 2000; Johnstone & Titterington, 2009). Due to the large increase in available data within the last ten years, this topic has become even more prevalent. While the number of features in environmental modeling is usually low, the availability and use of hyperspectral data changes this. A typical hyperspectral sensor usually consists of hundreds of bands. These bands are often used as the base to generate a large number of features, which are then used for modeling. A dataset which is composed out of more features than observations is commonly referred to as "high-dimensional". Yet this definition is only a rule of thumb (Johnstone & Titterington, 2009).

While wrapper feature selection is a well known and established approach in modeling (El Aboudi & Benhlima, 2016; Kohavi & John, 1997), the study presented in chapter 3 focuses on a less popular feature selection approach: filter methods. Filters have two main advantages compared to other feature selection techniques, such as wrapper methods; (1) merged execution with the hyperparameter tuning stage and (2) the ability to reuse existing ranking calculations due to caching. Essentially, the number of features can be treated as an optimizable hyperparameter of the model. Because filters rank the variable space only once using a specific heuristic, the calculated rankings can be reused in reoccurring optimization steps without the need for a recalculation. This is especially helpful in high-dimensional settings as for example in the second case study of this thesis (chapter 3).

The technical implementations of filter methods in the `R` packages `mlr` and `mlr3` allow to make use of said caching. The implementation of this option and the possibility to use ensemble filters in general was a byproduct of this study. The second case study (chapter 3) then focused on applying different (ensemble) filter methods on different combinations of feature sets derived from hyperspectral data by making use of various algorithms.

Overall models using filter methods did not score a substantially better performance in the case study when compared to models that were fitted without filters (Figure 3.4). Yet specific model/feature set combinations showed notable improvements when making use of filters, e.g., XGBoost on the HR and VI datasets in the second case study (Figure 3.4). Due to their low computational demand and their potential to shrink the predictor space and speed up high-dimensional modeling setups, the use of filters appears to be largely beneficial. A quantification of the speed up is not straightforward and essentially depends on the overall model fitting time of a specific algorithm with a specific amount of features. Once calculated, filter rankings can be used as a pretty much instant subset of the feature set in the respective tuning iteration and thus save time fitting the model in each iteration as the used algorithm does not need to incorporate all available features. The net speedup could be calculated by comparing the one-time calculation of the filter ranking against the added up time saved during each tuning iteration model fit. Though suchlike detailed comparisons are usually not conducted in practice as the exact timings are of low interest and not easily retrievable. PCA achieved similar results with respect to model fitting time in the conducted case study. Because PCA compresses the variable space into few principal components, which are then used to fit the model, PCA exacerbates subsequent interpretation at the single predictor level. This variable squashing and the additional step to disentangle the principal components into single features again might be potential advantages for using filter methods over PCA for feature selection purposes.

Lately the use of filter-based approaches and their development was mainly happening in the bioinformatics field. Many analyses of this research field face high-dimensional modeling challenges (Raj &

Mohanasundaram, 2020; Urbanowicz et al., 2018). No other studies in the environmental/ecological field could be found which made use of filter-based feature selection. It is hoped that the study conducted in chapter 3 can serve as a starting point for the environmental modeling community, both from a study design and programmatic point of view.

5.3.1 Ensemble vs. single filters

The use of ensemble filters showed promising results in recent studies (Drotár, Gazda, & Gazda, 2017; Drotár et al., 2019; Drotár, Šimoňák, et al., 2017; C.-F. Tsai & Sung, 2020) which was one of the drivers to incorporate the BORDA filter. This filter type returns the ranked sum of multiple individual filters and aims to average out the potential weaknesses of single filters, which is also the general idea of any ensemble approach (Pes, 2020). The results of the second case study did not coincide with the promising literature findings as the BORDA filter was not able to score the best result in any learner/feature set/filter setting (Figure 3.5). Yet the use of ensemble filter methods is essentially cost-free in benchmark setups as the calculation of the underlying single filter rankings can be reused in other runs. Overall, more applications of filter methods with other (high-dimensional) environmental datasets are needed to draw generalized conclusions on their performance compared to single filters. Besides solely focusing on filter methods, it might also be worth looking into combining filter and wrapper based approaches as shown by C.-W. Chen et al. (2020) and Pes (2020). Examples includes the creation of hybrid ensembles or the use of multistage approaches, i.e., first applying filters to reduce the feature space substantially and then applying wrapper methods on the remaining subset of the data.

5.3.2 Effects of data-driven feature sets

Multiple feature sets were created from the hyperspectral data used in the second case study (chapter 3) by making use of data-driven feature engineering. The goal was to improve model performances by making use of both arbitrary (NRI feature set) and expert-based indices (VI). Even though the overall best benchmark setting in the study used a combination of the NRI and VI feature sets (Table 3.3), the average performance differences between the feature sets were negligibly small. The HR feature set was thought of as a baseline to compete against as it reflects the reflectance values as seen from the hyperspectral sensor. Overall, the effort of creating additional feature sets from the hyperspectral data did not result in a substantial improvement in predictive performance in this specific case study.

5.3.3 Model interpretation with high-dimensional feature sets

Permutation-based variable importance was used to estimate variable importance of the best performing algorithm (SVM) on the HR and VI feature set, respectively. The so-called red edge (ranging between 680 nm and 720 nm) was estimated to have the highest influence on model performance (Figure 3.6). This finding is in line with other vegetation-related research (Kang et al., 2021; Zhu et al., 2017). The absolute importance of the most important feature(s), which is reflected by the mean decrease in RMSE across permutations when omitting this particular feature, were rather small. This applied especially to the HR dataset for which only one feature exceeded the absolute value of one RMSE.

From a general point of view, the use of a permutation-based method in a scenario with highly-correlated variables must be seen critical as results might be biased (Molnar et al., 2020). A general prob-

lem of permutation-based variable importance is that it creates (highly) unreasonable combinations of feature values. This might result in a high (false-positive) estimated importance of a particular (correlated) feature as the predictive performance of such a model often drops substantially (Molnar et al., 2020). Yet it is unclear if and how much bias this effect introduces into the final performance estimation as it is usually not feasible to assess the amount of unreasonable feature combinations within a dataset beforehand.

This limitation does not apply to local methods like ALE plots or SHAP importance (Apley & Zhu, 2020; Lundberg & Lee, 2017). Local model-agnostic methods like these aim to explain the importance of features of individual predictions rather than estimating the overall model performance (Lundberg & Lee, 2017). However, they also suffer from the problem of shared importance, i.e., if highly-correlated features are present, their individual importance weight is split across the respective correlated features, essentially masking their actual individual importance. A potential solution to this can be a bundled assessment of (highly)-correlated features, also known as grouped feature importance (Au et al., 2021). Here, rather than looking at individual feature importance, the focus is on explaining the importance of a feature group. Brenning (2021) recently came up with a new method which aims to address this problem and builds open existing solutions similar to ALE or (grouped) permutation-based variable importance. The approach builds on the analysis of features in a transformed feature space (e.g. through different means of PCA), and shares ideas earlier proposed by Adebayo and Kagal (2016).

5.4 (Spatial) research software in R & reproducibility

5.4.1 Generalist repositories

To ensure study reproducibility, data must be shared publicly and a license needs to be attributed to it. This paragraph discusses currently available options and their differences.

The selection of a generalist repository to ensure long-term reproducibility of the study data is important. Sharing code and small-sized data on git-based platforms such as GitHub, GitLab, Bitbucket or others is a good start with respect to reproducibility. However, generalist repository providers are needed to ensure longterm conservation of data and code as profit-oriented offerings (e.g., the previously mentioned git providers) are not guaranteed to exist longterm (Bast, 2019). In addition the mentioned git-platforms have a strong focus on code hosting. While the upload of small datasets, around a few megabytes in size, is generally unproblematic, problems might occur for larger datasets. If a certain threshold is exceeded, data storage might incur costs. Hence, platforms are needed which follow the FAIR principles and focus on making (research) data available long-term. Examples are Zenodo (`https://zenodo.org`), the Open Science Framework (`https://osf.io`), Software Heritage (`https://softwareheritage.org`), Dataverse (`https://dataverse.org`) or Figshare (`https://figshare.com`) (Bast, 2019). Other field-specific initiatives like PANGAEA exist which also aim to provide a generic data storage offering to scientists (*PANGAEA: Data Publisher for Earth & Environmental Science*, 2022). Yet many field-specific offerings suffer from their discoverability and are sometimes covered by other, non-science related services that have a higher ranking in web search results (e.g., PANGAEA) and might therefore not be covered here.

Generalist repository providers which are tied to a particular research entity such as journal publishers (e.g., "Mendeley Data" as a partner of Elsevier) or university-specific repository providers must be seen critical. University-driven projects are often hard to discover and might be tied to a limited budget,

which imposes potential maintenance risks. Providers affiliated with journal publishers might also suffer from budget and, in addition, marketing decision issues. Also, due to the convenient integration into a journal's submission process, their use is often not cost-free and comes eventually with the devolution of data permission.

With respect to literature, Choi et al. (2021) also emphasizes the importance of reproducible science in their work and showcases a combination of different online tools to conduct a hydrological case study. The National Academies of Science in the United States published a relatively recent report which includes an exhaustive overview on reproducibility in science (National Academies of Sciences & Medicine, 2019).

For this thesis, Zenodo was chosen as a longterm generalist repository provider given its open source nature, the ability to attach DOIs to datasets, and its backing by the European Commission (within the Horizon 2020 initiative) (European Organization For Nuclear Research & OpenAIRE, 2013).

5.4.2 License attribution

Besides the selection of a generalist repository provider, a license which declares distribution and reusability permissions of the shared content is an important building block for reproducible research. One of the most used licenses which strongly supports reproducibility and allows reusability of data is the "Creative Commons" license family with its specific "CC BY-SA" or "CC BY" types (Y.-H. Lin et al., 2006). Here "BY" stands for the need of copying authors to credit the original authors of the data. The "SA" acronym does not allow new authors to use a more restrictive license in their derived works. More options can be added to a "Creative Commons" license to make it more restrictive, for example the restriction to not allow the reuse for commercial purposes (type "NC"). Other licenses, which encourage reusability, such as MIT or "GNU Lesser General Public License" (LGPL), are mainly used in a software context (Y.-H. Lin et al., 2006). Similar to the "SA" option of the Creative Common license, LGPL enforces the use of the LGPL license for the new product. MIT is the oldest (created in 1988), most permissive and most used license for software projects as of 2020 (Goldstein, 2020). It should be noted that the described licenses from above represent only a subset of licenses which are suitable for the use in research compendia licensing.

5.4.3 Toolboxes for reproducible research workflow management

Software toolboxes can help in the process of executing reproducible research in a well-defined and standardized way. They can be applied at different analysis levels. General ones such as Docker (Merkel, 2014), ReproZip (Chirigati et al., 2016) or "The experiment factory" (Sochat, 2018) can help to bundle analysis on the operating system level, i.e., they define a fixed environment in which the analysis is executed. The "Codecheck" project on the other hand scans an existing code-driven analysis and provides tailored guidelines and best practice recommendations how reproducibility can possibly be improved (Nüst & Eglen, 2021). Those tools can be combined with domain-specific implementations which help to ensure reproducibility on the programming language level. Examples for the `R` programming language are `containerit` (Nüst & Hinz, 2019), `renv` (Ushey, 2021), `drake` (Landau, 2018) or `drakes` successor `targets` (Landau, 2021).

Adhering to these concepts in spatial analysis practice can be complicated given the widespread use of graphical user interface (GUI)-based and/or licensed software in the geoscience field (e.g ArcGIS or ENVI) (Konkol et al., 2019). This view is also shared by Gil et al. (2016) who, in addition, proposes guidelines for the "Geoscience Paper of the Future" in their work which is similar to the concept of a research

compendium. The use of licensed and/or GUI-based software does not completely block study reproducibility but makes it more difficult. Making use of recent online-based services (e.g., Google Earth Engine) for processing data is an additional threat to reproducibility as these services are not guaranteed to be accessible in the future (Nüst & Pebesma, 2021). In general, the use of any non-programmatic approach requires extensive documentation and lowers the probability of a study being reproducible. Studies which involve different software environments (GUI software or different programming languages) either require an overarching toolbox which combines the execution of the respective individual analysis parts or manual connectors between such. The use of a fully code-driven analysis which relies on a single programming language is arguably the desired execution approach with respect to reproducibility (Nüst & Pebesma, 2021; Singleton et al., 2016). This work solely uses the programming language `R` and makes heavy use of the packages `renv` and `drake` as already outlined in more detail in subsection 1.8.1.

While the adaptation of such reproducible toolboxes has slowly increased in recent years, more weight needs to be put on their role and practical use within science. Teaching the theoretical concepts and practical solutions during undergraduate and graduate studies is a key step to establish the reproducible research idea in daily scientific practice. Almost all mentioned concepts also apply to scientific fields outside of the natural sciences and their respective reproducibility toolbox implementations. One example from the life sciences is the "Galaxy" project (Goecks et al., 2010).

CHAPTER 6

CONCLUSIONS

The goal of this thesis was to simplify the application of spatial modeling in R, with a special focus on bias-reduced performance estimation, hyperparameter optimization and feature selection on spatial datasets. This was done by analyzing the forest pathogens *D. sapinea* and *F. circinatum* in the Basque Country in northern Spain following reproducibility best practices. The overall scope of this work is relatively broad and addresses different important subjects in the environmental modeling field; from data analysis over software development to best practices in workflow management.

The main conclusions are presented in the following:

1. The use of spatial modeling to help monitor pathogen infections at trees can be helpful assessing disease spread. This thesis highlighted the importance of data quality, both for in situ and remote sensing data, to achieve model performances which can be of practical help for local decision makers to understand and reduce the spread of such forest diseases. Additional explanatory variables which describe the surrounding environment beyond its climatic state are needed to further improve model performances. Forest pathogen spread is predicted to increase due to climate warming. The use of semi-automated mapping approaches can play an important role in helping to control invasive disease spreading in the future.

2. This thesis investigated the effects of using spatial partitioning during the model optimization stage in a nested CV setting for the first time. Even though the results did not indicate a single strategy to be advantageous, small performance improvements were identified for specific settings. The results encourage the use of consistent resampling strategies for both model optimization and assessment, although the use of a non-spatial resampling strategy during tuning, in contrast to the validation resampling strategy, is unlikely to have a substantial impact on the results.

3. The generalization of software implementations is beneficial for both the work at hand and fellow researchers. By focusing on software documentation and testing, theoretical concepts are solidified and added value beyond the analytical part is created. Numerous contributions to existing free and open source software (FOSS) in R were done throughout this work. The R package `mlr3spatiotempcv`, which bundles spatiotemporal resampling methods in R for simplified application in nested CV scenarios, represents exemplary the software related efforts throughout this thesis.

4. The use of filter-based feature selection in a high-dimensional, collinear data setting did not result in significant improvements of predictive performance. Some specific model setups (i.e., specific combinations of algorithm, feature set and filter method) profited from the use of filters but no consistent pattern could be identified across all benchmark settings. The use of ensemble filters compared to single filters or the use of no filters did not yield an improvement in this work. The integration of filter methods into the tuning step resulted in time savings during the model optimization stage due to reduced model fitting times.

5. Variable importance results of features derived from the (environmental) hyperspectral data showed that variables in the red-edge area of the spectrum are most important to the used models when modeling tree defoliation. While this finding coincides with existing research, results might be biased due to the high correlation among the predictors which classical permutation-based variable importance is not able to account for. More sophisticated approaches like grouped permutation approaches, SHAP or ALE are needed for unbiased interpretation of variable importance.

6. Reproducible research and workflow management tools are able to enhance scientific research by emphasizing transparency and openness of methodologies and concepts used in academic studies.

REFERENCES

Abeel, T., Helleputte, T., Van de Peer, Y., Dupont, P., & Saeys, Y. (2010, February). Robust biomarker identification for cancer diagnosis with ensemble feature selection methods. *Bioinformatics*, *26*(3), 392–398. doi: 10.1093/bioinformatics/btp630

Adamczyk, J., & Osberger, A. (2015, May). Red-edge vegetation indices for detecting and assessing disturbances in Norway spruce dominated mountain forests. *International Journal of Applied Earth Observation and Geoinformation*, *37*, 90–99. doi: 10.1016/j.jag.2014.10.013

Adamson, K., Klavina, D., Drenkhan, R., Gaitnieks, T., & Hanso, M. (2015, October). Diplodia sapinea is colonizing the native Scots pine (Pinus sylvestris) in the northern Baltics. *European Journal of Plant Pathology*, *143*(2), 343–350. doi: 10.1007/s10658-015-0686-8

Adebayo, J., & Kagal, L. (2016, November). *Iterative Orthogonal Feature Projection for Diagnosing Bias in Black-Box Models* (No. arXiv:1611.04967). doi: 10.48550/arXiv.1611.04967

Adelabu, S., Mutanga, O., & Adam, E. (2014, September). Evaluating the impact of red-edge band from Rapideye image for classifying insect defoliation levels. *ISPRS Journal of Photogrammetry and Remote Sensing*, *95*, 34–41. doi: 10.1016/j.isprsjprs.2014.05.013

Adler, P., Falk, C., Friedler, S. A., Nix, T., Rybeck, G., Scheidegger, C., … Venkatasubramanian, S. (2018, January). Auditing black-box models for indirect influence. *Knowledge and Information Systems*, *54*(1), 95–122. doi: 10.1007/s10115-017-1116-3

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*(4), 247–251. doi: 10.1111/cod.12706

Aegerter, B. J., & Gordon, T. R. (2006, November). Rates of pitch canker induced seedling mortality among Pinus radiata families varying in levels of genetic resistance to Gibberella circinata (anamorph Fusarium circinatum). *Forest Ecology and Management*, *235*(1), 14–17. doi: 10.1016/j.foreco.2006.07.011

Anderson, P., Turner, M. G., Forester, J. D., Zhu, J., Boyce, M. S., Beyer, H., & Stowell, L. (2005). Scale-dependent summer resource selection by reintroduced elk in Wisconsin, USA. *The Journal of Wildlife Management*, *69*(1), 298–310. Retrieved 2021-05-08, from https://www.jstor.org/stable/3803606

Apan, A., Held, A., Phinn, S., & Markley, J. (2004, January). Detecting sugarcane 'orange rust' disease using EO-1 Hyperion hyperspectral imagery. *International Journal of Remote Sensing*, *25*(2), 489–498. doi: 10.1080/01431160310001618031

Apley, D. W., & Zhu, J. (2020). Visualizing the effects of predictor variables in black box supervised learning models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *82*(4), 1059–1086. doi: 10.1111/rssb.12377

Arlot, S., & Celisse, A. (2010, January). A survey of cross-validation procedures for model selection. *Statistics Surveys*, *4*, 40–79. doi: 10.1214/09-SS054

Au, Q., Herbinger, J., Stachl, C., Bischl, B., & Casalicchio, G. (2021, April). *Grouped feature importance and combined features effect plot.*

Retrieved 2021-12-28, from https://arxiv.org/abs/2104.11688v1

Ayres, M. P., & Lombardero, M. J. (2000, November). Assessing the consequences of global change for forest disturbance from herbivores and pathogens. *Science of The Total Environment*, *262*(3), 263–286. doi: 10.1016/S0048-9697(00)00528-3

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*(4), 565–574. doi: 10.1016/j.ecolmodel.2009.10.033

Bahn, V., & McGill, B. J. (2012, December). Testing the predictive performance of distribution models. *Oikos*, *122*(3), 321–331. doi: 10.1111/j.1600-0706.2012.00299.x

Bast, R. (2019, August). A FAIRer future. *Nature Physics*, *15*(8), 728–730. doi: 10.1038/s41567-019-0624-3

Bebber, D. P., & Butt, N. (2017, October). Tropical protected areas reduced deforestation carbon emissions by one third from 2000–2012. *Scientific Reports*, *7*(1), 14005. doi: 10.1038/s41598-017-14467-w

Becker, M., Binder, M., Bischl, B., Lang, M., Pfisterer, F., Reich, N. G., … Sonabend, R. (2020, October). *mlr3 book* [manual]. Retrieved from https://mlr3book.mlr-org.com

Belgiu, M., & Drăguţ, L. (2016, April). Random forest in remote sensing: A review of applications and future directions. *ISPRS Journal of Photogrammetry and Remote Sensing*, *114*, 24–31. doi: 10.1016/j.isprsjprs.2016.01.011

Bengio, Y. (2000, August). Gradient-based optimization of hyperparameters. *Neural Computation*, *12*(8), 1889–1900. doi: 10.1162/089976600300015187

Bengio, Y., & Grandvalet, Y. (2004, December). No unbiased estimator of the variance of k-fold cross-validation. *The Journal of Machine Learning Research*, *5*, 1089–1105. Retrieved from https://www.jmlr.org/papers/volume5/grandvalet04a/grandvalet04a.pdf

Bergmeir, C., Hyndman, R. J., & Koo, B. (2018, April). A note on the validity of cross-validation for evaluating autoregressive time series prediction. *Computational Statistics & Data Analysis*, *120*, 70–83. doi: 10.1016/j.csda.2017.11.003

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

Binder, M., Moosbauer, J., Thomas, J., & Bischl, B. (2020, February). Multi-objective hyperparameter tuning and feature selection using filter ensembles. *arXiv:1912.12912 [cs, stat]*. Retrieved 2020-05-29, from http://arxiv.org/abs/1912.12912

Binder, M., Pfisterer, F., Lang, M., Schneider, L., Kotthoff, L., & Bischl, B. (2021). mlr3pipelines - flexible machine learning pipelines in r. *Journal of Machine Learning Research*, *22*(184), 1–7. Retrieved from https://jmlr.org/papers/v22/21-0281.html

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).

Bischl, B., Binder, M., Lang, M., Pielok, T., Richter, J., Coors, S., … Lindauer, M. (2021, November). Hyperparameter Optimization: Foundations, Algorithms, Best Practices and Open Challenges. *arXiv:2107.05847 [cs, stat]*. Retrieved 2022-04-24, from http://arxiv.org/abs/2107.05847

Bischl, B., Lang, M., Kotthoff, L., Schiffner, J., Richter, J., Studerus, E., … Jones, Z. M. (2016). mlr: Machine learning in R. *Journal of Machine Learning Research*, *17*(170),

1–5. Retrieved from http://jmlr.org/papers/v17/15-066.html

Bischl, B., Mersmann, O., Trautmann, H., & Weihs, C. (2012, June). Resampling Methods for Meta-Model Validation with Recommendations for Evolutionary Computation. *Evolutionary Computation*, *20*(2), 249–275. doi: 10.1162/EVCO_a_00069

Bischl, B., Richter, J., Bossek, J., Horn, D., Thomas, J., & Lang, M. (2017, March). *mlrMBO: a modular framework for model-based optimization of expensive black-box functions*. Retrieved from https://arxiv.org/abs/1703.03373

Blackburn, G. A. (1998, December). Quantifying chlorophylls and caroteniods at leaf and canopy scales. *Remote Sensing of Environment*, *66*(3), 273–285. doi: 10.1016/s0034-4257(98)00059-5

Blaschke, M., & Cech, T. (2007). Absterbende Weisskiefern–eine langfristige Folge des Trockenjahres 2003. *Forstschutz aktuell*, *40*, 32–34. Retrieved from https://www.waldwissen.net/assets/waldwirtschaft/schaden/trockenheit/lwf_weisskiefersterben/download/lwf_forstschutzaktuell_072007_weisskiefersterben_2007_4.pdf.pdf

Bolón-Canedo, V., & Alonso-Betanzos, A. (2019, December). Ensembles for feature selection: A review and future trends. *Information Fusion*, *52*, 1–12. doi: 10.1016/j.inffus.2018.11.008

Bommert, A., Sun, X., Bischl, B., Rahnenführer, J., & Lang, M. (2020, March). Benchmark for filter methods for feature selection in high-dimensional classification data. *Computational Statistics & Data Analysis*, *143*, 106839. doi: 10.1016/j.csda.2019.106839

Boochs, F., Kupfer, G., Dockter, K., & Kühbauch, W. (1990, October). Shape of the red edge as vitality indicator for plants. *International Journal of Remote Sensing*, *11*(10), 1741–1753. doi: 10.1080/01431169008955127

Bosso, L., Luchi, N., Maresi, G., Cristinzio, G., Smeraldo, S., & Russo, D. (2017, September). Predicting current and future disease outbreaks of Diplodia sapinea shoot blight in Italy: species distribution models as a tool for forest management planning. *Forest Ecology and Management*, *400*, 655–664. doi: 10.1016/j.foreco.2017.06.044

Breiman, L. (2001). Random Forests. *Machine Learning*, *45*(1), 5–32. doi: 10.1023/a:1010933404324

Brenning, A. (2005, November). Spatial prediction models for landslide hazards: review, comparison and evaluation. *Natural Hazards and Earth System Sciences*, *5*(6), 853–862. doi: 10.5194/nhess-5-853-2005

Brenning, A. (2012, July). 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

Brenning, A. (2021, April). Transforming feature space to interpret machine learning models. *arXiv:2104.04295 [cs, stat]*. Retrieved 2021-11-09, from http://arxiv.org/abs/2104.04295

Brenning, A., & Lausen, B. (2008, September). Estimating error rates in the classification of paired organs. *Statistics in Medicine*, *27*(22), 4515–4531. doi: 10.1002/sim.3310

Brenning, A., Long, S., & Fieguth, P. (2012, October). Detecting rock glacier flow structures using Gabor filters and IKONOS imagery. *Remote Sensing of Environment*, *125*, 227–237. doi: 10.1016/j.rse.2012.07.005

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*(1), 45–57. doi: 10.5194/nhess-15-45-2015

Brett, A., Croucher, M., Haines, R., Hettrick, S., Hetherington, J., Stillwell, M., & Wyatt, C. (2017, April). *Research software engineers: state of the nation report 2017* (Tech. Rep.). Engineering and Physical Sciences Research Council. Retrieved 2021-12-21, from 10.5281/zenodo.495360

Brier, G. W. (1950, January). Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, *78*(1), 1–3. doi: 10.1175/1520-0493(1950)078<0001:vofeit>2.0.co;2

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*. Retrieved from http://arxiv.org/abs/1012.2599

Brodde, L., Adamson, K., Julio Camarero, J., Castaño, C., Drenkhan, R., Lehtijärvi, A., … Oliva, J. (2019). Diplodia tip blight on its way to the north: drivers of disease emergence in northern Europe. *Frontiers in Plant Science*, *9*. doi: 10.3389/fpls.2018.01818

Broge, N., & Leblanc, E. (2001, May). Comparing prediction power and stability of broadband and hyperspectral vegetation indices for estimation of green leaf area index and canopy chlorophyll density. *Remote Sensing of Environment*, *76*(2), 156–172. doi: 10.1016/s0034-4257(00)00197-8

Brungard, C. W., Boettinger, J. L., Duniway, M. C., Wills, S. A., & Edwards, T. C. (2015, February). Machine learning for predicting soil classes in three semi-arid landscapes. *Geoderma*, *239–240*, 68–83. doi: 10.1016/j.geoderma.2014.09.019

Bui, D. T., Tuan, T. A., Klempe, H., Pradhan, B., & Revhaug, I. (2015, January). 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*(2), 361–378. doi: 10.1007/s10346-015-0557-6

Burgess, T., & Wingfield, M. (2002). Quarantine is important in restricting the spread of exotic seed-borne tree pathogens in the southern hemisphere. *The International Forestry Review*, *4*(1), 56–65. Retrieved 2021-12-24, from https://www.jstor.org/stable/43740945

Burman, P., Chow, E., & Nolan, D. (1994). A cross-validatory method for dependent data. *Biometrika*, *81*(2), 351–358. doi: 10.1093/biomet/81.2.351

Byrne, S. (2016). A note on the use of empirical AUC for evaluating probabilistic forecasts. *Electronic Journal of Statistics*, *10*(1), 380–393. doi: 10.1214/16-ejs1109

Cai, J., Luo, J., Wang, S., & Yang, S. (2018, July). Feature selection in machine learning: A new perspective. *Neurocomputing*, *300*, 70–79. doi: 10.1016/j.neucom.2017.11.077

Carter, G. A. (1994, February). Ratios of leaf reflectances in narrow wavebands as indicators of plant stress. *International Journal of Remote Sensing*, *15*(3), 697–703. doi: 10.1080/01431169408954109

Castaño-Santamaría, J., López-Sánchez, C. A., Ramón Obeso, J., & Barrio-Anta, M. (2019, October). Modelling and mapping beech forest distribution and site productivity under different climate change scenarios in the Cantabrian Range (North-western Spain). *Forest Ecology and Management*, *450*, 117488. doi: 10.1016/j.foreco.2019.117488

Cawley, G. C., & Talbot, N. L. C. (2010). On over-fitting in model selection and subsequent selection bias in performance evaluation. *Journal of Machine Learning Research*, *11*(70), 2079–2107. Retrieved 2021-05-22, from http://jmlr.org/papers/v11/cawley10a.html

Chappelle, E. W., Kim, M. S., & McMurtrey, J. E. (1992, March). Ratio analysis of reflectance spectra (RARS): An algorithm for the remote estimation of the concentrations of chlorophyll A, chlorophyll B, and carotenoids in soybean leaves. *Remote Sensing of Environment*, *39*(3), 239–247. doi: 10.1016/0034-4257(92)90089-3

Chehata, N., Orny, C., Boukir, S., Guyon, D., & Wigneron, J. (2014, July). Object-based change detection in wind storm-damaged forest using high-resolution multispectral images. *International Journal of Remote Sensing*, *35*(13), 4758–4777. doi: 10.1080/01431161.2014.930199

Chen, C.-W., Tsai, Y.-H., Chang, F.-R., & Lin, W.-C. (2020). Ensemble feature selection in medical datasets: Combining filter, wrapper, and embedded feature selection results. *Expert Systems*, *37*(5), e12553. doi: 10.1111/exsy.12553

Chen, G., & Meentemeyer, R. K. (2016). Remote sensing of forest damage by diseases and insects. *Remote sensing for sustainability*, 145–162. Retrieved from https://books.google.de/books?id=naSKDQAAQBAJ

Chen, J. M. (1996, September). Evaluation of vegetation indices and a modified simple ratio for boreal applications. *Canadian Journal of Remote Sensing*, *22*(3), 229–242. doi: 10.1080/07038992.1996.10855178

Chen, T., & Guestrin, C. (2016). xgboost: a scalable tree boosting system. In *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785–794). New York, NY, USA: ACM. doi: 10.1145/2939672.2939785

Chirigati, F., Rampin, R., Shasha, D., & Freire, J. (2016). ReproZip: Computational reproducibility with ease. In *Proceedings of the 2016 international conference on management of data* (pp. 2085–2088). San Francisco, California, USA: ACM. doi: 10.1145/2882903.2899401

Cho, M. A., & Skidmore, A. K. (2006, March). A new technique for extracting the red edge position from hyperspectral data: The linear extrapolation method. *Remote Sensing of Environment*, *101*(2), 181–193. doi: 10.1016/j.rse.2005.12.011

Choi, Y.-D., Goodall, J. L., Sadler, J. M., Castronova, A. M., Bennett, A., Li, Z., … Tarboton, D. G. (2021, January). Toward open and reproducible environmental modeling by integrating online data repositories, computational environments, and model Application Programming Interfaces. *Environmental Modelling & Software*, *135*, 104888. doi: 10.1016/j.envsoft.2020.104888

Chou, C. (1982). Susceptibility of Pinus radiata seedlings to infection by Diplodia pinea as affected by pre-inoculation conditions. *New Zealand Journal of Forestry Science*, *12*(3), 438–441. Retrieved from https://www.semanticscholar.org/paper/Susceptibility-of-Pinus-radiata-seedlings-to-by-as-Chou/542273718211eeec12b4977bdec1e7d7872810f3

Cliff, A. D., & Ord, K. (1970, June). Spatial autocorrelation: A review of existing and new measures with applications. *Economic Geography*, *46*, 269. doi: 10.2307/143144

Comber, S. (2020, September). *spacv: Spatial cross-validation in Python*. Retrieved 2021-12-27, from https://github.com/SamComber/spacv

Cook, R. J., & Baker, K. F. (1983). *The nature and practice of biological control of plant pathogens*. American Phytopathological Society. Retrieved from https://books.google.ch/books?id=hvbwAAAAMAAJ

Cressie, N. A. C. (1993). *Statistics for spatial data*. John Wiley & Sons, Inc. doi: 10.1002/9781119115151

Dale, M. R., & Fortin, M.-J. (2002, January). Spatial autocorrelation and statistical tests in

ecology. *Écoscience*, *9*(2), 162–167. doi: 10.1080/11956860.2002.11682702

Das, S. (2001). Filters, wrappers and a boosting-based hybrid for feature selection. In *ICML*. Citeseer. Retrieved from https://www.semanticscholar.org/paper/Filters%2C-Wrappers-and-a-Boosting-Based-Hybrid-for-Das/93b625a0e35b59fa6a3e7dc1cbdb31268d62d69f?p2df

Dash, J., & Curran, P. (2007, January). Evaluation of the MERIS terrestrial chlorophyll index (MTCI). *Advances in Space Research*, *39*(1), 100–104. doi: 10.1016/j.asr.2006.02.034

Dash, J. P., Watt, M. S., Pearse, G. D., Heaphy, M., & Dungey, H. S. (2017, September). Assessing very high resolution UAV imagery for monitoring forest health during a simulated disease outbreak. *ISPRS Journal of Photogrammetry and Remote Sensing*, *131*, 1–14. doi: 10.1016/j.isprsjprs.2017.07.007

Datt, B. (1998, November). Remote Sensing of Chlorophyll a, Chlorophyll b, Chlorophyll a$\mathplus$b, and Total Carotenoid Content in Eucalyptus Leaves. *Remote Sensing of Environment*, *66*(2), 111–121. doi: 10.1016/s0034-4257(98)00046-7

Datt, B. (1999a). Remote sensing of water content in eucalyptus leaves. *Australian Journal of Botany*, *47*(6), 909. doi: 10.1071/bt98042

Datt, B. (1999b, January). Visible/near infrared reflectance and chlorophyll content in Eucalyptus leaves. *International Journal of Remote Sensing*, *20*(14), 2741–2759. doi: 10.1080/014311699211778

Daughtry, C. (2000, November). Estimating corn leaf chlorophyll concentration from leaf and canopy reflectance. *Remote Sensing of Environment*, *74*(2), 229–239. doi: 10.1016/s0034-4257(00)00113-9

de Beurs, K. M., & Townsend, P. A. (2008, October). Estimating the effect of gypsy moth defoliation using MODIS. *Remote Sensing of Environment*, *112*(10), 3983–3990. doi: 10/fpqhrc

De'ath, G. (2007). Boosted trees for ecological modeling and prediction. *Ecology*, *88*(1), 243–251. doi: 10.1890/0012-9658(2007)88[243:btfema]2.0.co;2

Desprez-Loustau, M.-L., Robin, C., Reynaud, G., Déqué, M., Badeau, V., Piou, D., … Marçais, B. (2007, June). Simulating the effects of a climate-change scenario on the geographical range and activity of forest-pathogenic fungi. *Canadian Journal of Plant Pathology*, *29*(2), 101–120. doi: 10.1080/07060660709507447

Devey, M., Matheson, C., & Gordon, T. (1999). Current and potential impacts of pitch canker in radiata pine. In *Proc. IMPACT monterey workshop, monterey, CA, USA* (Vol. 30). Retrieved from https://catalogue.nla.gov.au/Record/179313

Di Cecco, G. J., & Gouhier, T. C. (2018, October). Increased spatial and temporal autocorrelation of temperature under climate change. *Scientific Reports*, *8*(1), 14850. doi: 10.1038/s41598-018-33217-0

Diesing, M. (2020, December). Deep-sea sediments of the global ocean. *Earth System Science Data*, *12*(4), 3367–3381. doi: 10.5194/essd-12-3367-2020

Dietterich, T. G. (2000, June). Ensemble methods in machine learning. In *Proceedings of the First International Workshop on Multiple Classifier Systems* (pp. 1–15). Springer-Verlag. Retrieved 2019-06-23, from http://dl.acm.org/citation.cfm?id=648054.743935

Donoho, D. L. (2000). High-dimensional data analysis: The curses and blessings of dimensionality. *AMS math challenges lecture*, *1*(2000), 32. Retrieved from https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.329.3392&rep=rep1&type=pdf

Dormann, C. F. (2007, March). Effects of incorpo-

rating spatial autocorrelation into the analysis of species distribution data. *Global Ecology and Biogeography*, *16*(2), 129–138. doi: 10.1111/j.1466-8238.2006.00279.x

Dormann, C. F., McPherson, J. M., Araújo, M. B., Bivand, R., Bolliger, J., Carl, G., … Wilson, R. (2007, September). Methods to account for spatial autocorrelation in the analysis of species distributional data: a review. *Ecography*, *30*(5), 609–628. doi: 10.1111/j.2007.0906-7590.05171.x

Doshi-Velez, F., & Kim, B. (2017, March). Towards A Rigorous Science of Interpretable Machine Learning. *arXiv:1702.08608 [cs, stat]*. Retrieved 2022-04-25, from http://arxiv.org/abs/1702.08608

Drenkhan, R., Ganley, B., Martín-García, J., Vahalík, P., Adamson, K., Adamčíková, K., … Mullett, M. S. (2020, July). Global geographic distribution and host range of Fusarium circinatum, the causal agent of pine pitch canker. *Forests*, *11*(7), 724. doi: 10.3390/f11070724

Drotár, P., Gazda, J., & Smékal, Z. (2015, November). An experimental comparison of feature selection methods on two-class biomedical datasets. *Computers in Biology and Medicine*, *66*, 1–10. doi: 10.1016/j.compbiomed.2015.08.010

Drotár, P., Gazda, M., & Gazda, J. (2017, October). Heterogeneous ensemble feature selection based on weighted Borda count. In *2017 9th International Conference on Information Technology and Electrical Engineering (ICITEE)* (pp. 1–4). doi: 10.1109/iciteed.2017.8250495

Drotár, P., Gazda, M., & Vokorokos, L. (2019, April). Ensemble feature selection using election methods and ranker clustering. *Information Sciences*, *480*, 365–380. doi: 10/gf3568

Drotár, P., Šimoňák, S., Pietriková, E., Chovanec, M., Chovancová, E., Ádám, N., … Biňas, M. (2017, June). Comparison of filter techniques for two-step feature selection. *Computing and Informatics*, *36*(3), 597–617. doi: 10.4149/cai_2017_3_597

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

Dudani, S. A. (1976, April). The distance-weighted k-Nearest-Neighbor rule. *IEEE Transactions on Systems, Man, and Cybernetics*, *SMC-6*(4), 325–327. doi: 10.1109/tsmc.1976.5408784

Dwinell, L. D., Barrows-Broaddus, J. B., & Kuhlman, E. G. (1985). Pitch canker: a disease complex. *Plant Disease*, *69*, 270–276.

Efron, B., & Gong, G. (1983, February). A leisurely look at the bootstrap, the jackknife, and cross-validation. *The American Statistician*, *37*(1), 36–48. doi: 10.1080/00031305.1983.10483087

EFSA. (2010). Risk assessment of Gibberella circinata for the EU territory and identification and evaluation of risk management options. *EFSA Journal*, *8*(6), 1620. doi: 10.2903/j.efsa.2010.1620

Egli, S., & Höpke, M. (2020, January). CNN-based tree species classification using high resolution RGB image data from automated UAV observations. *Remote Sensing*, *12*(23), 3892. doi: 10.3390/rs12233892

El Aboudi, N., & Benhlima, L. (2016, September). Review on wrapper feature selection approaches. In *2016 International Conference on Engineering MIS (ICEMIS)* (pp. 1–5). doi: 10.1109/ICEMIS.2016.7745366

Elith, J., & Leathwick, J. R. (2009). Species Distribution Models: Ecological Explanation and Prediction Across Space and Time. *Annual Review of Ecology, Evolution, and Systematics*, *40*(1), 677–697. doi: 10.1146/annurev.ecolsys.110308.120159

Elith, J., Leathwick, J. R., & Hastie, T. (2008). A working guide to boosted regression trees. *Journal of Animal Ecology*, *77*(4), 802–813.

doi: 10.1111/j.1365-2656.2008.01390.x

Elvidge, C. D., & Chen, Z. (1995, October). Comparison of broad-band and narrow-band red and near-infrared vegetation indices. *Remote Sensing of Environment*, *54*(1), 38–48. doi: 10.1016/0034-4257(95)00132-k

Endicott, S., Drescher, M., & Brenning, A. (2017, October). Modelling the spread of European buckthorn in the Region of Waterloo. *Biological Invasions*, *19*(10), 2993–3011. doi: 10.1007/s10530-017-1504-3

Entcheva, P. K., Rock, B. N., Lauten, G. N., & Cibula, W. G. (1996, December). Remote sensing assessment of forest health in the Bohemian forests of central Europe. (CONF-961119-). Retrieved 2022-05-09, from https://www.osti.gov/biblio/478367

Escobar, S., Helmstetter, A. J., Jarvie, S., Montúfar, R., Balslev, H., & Couvreur, T. L. P. (2021). Pleistocene climatic fluctuations promoted alternative evolutionary histories in Phytelephas aequatorialis, an endemic palm from western Ecuador. *Journal of Biogeography*, *48*(5), 1023–1037. doi: 10.1111/jbi.14055

European and Mediterranean Plant Protection Organization. (2021). *Current pest situation evaluated by EPPO on the basis of information dated 2019*. Retrieved 2021-12-13, from https://gd.eppo.int/taxon/GIBBCI/distribution/ES

European Organization For Nuclear Research, & OpenAIRE. (2013). *Zenodo* (Tech. Rep.). CERN. doi: 10.25495/7GXK-RD71

European Commission, J. R. C. (2010). *'Map of soil ph in Europe', land resources management unit, institute for environment & sustainability*. Retrieved from http://esdac.jrc.ec.europa.eu/content/soil-ph-europe

European Environment Agency, E. E. A. (2017). *Climate change, impacts and vulnerability in Europe 2016 : an indicator-based report* (Tech. Rep. No. 1). Publications Office. Retrieved from https://op.europa.eu/s/vWY7

Evans, A. S. (1976). Causation and disease: the Henle-Koch postulates revisited. *The Yale journal of biology and medicine*, *49*(2), 175.

Evira-Recuenco, M., Iturritxa, E., & Raposo, R. (2015, September). Impact of seed transmission on the infection and development of pitch canker disease in Pinus radiata. *Forests*, *6*(9), 3353–3368. doi: 10.3390/f6093353

Fabre, B., Piou, D., Desprez-Loustau, M.-L., & Marçais, B. (2011). Can the emergence of pine Diplodia shoot blight in France be explained by changes in pathogen pressure linked to climate change? *Global Change Biology*, *17*(10), 3218–3227. doi: 10.1111/j.1365-2486.2011.02428.x

Fassnacht, F. E., Neumann, C., Förster, M., Buddenbaum, H., Ghosh, A., Clasen, A., … Koch, B. (2014, June). Comparison of feature reduction algorithms for classifying tree species with hyperspectral data on three central European test sites. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *7*(6), 2547–2561. doi: 10.1109/JSTARS.2014.2329390

Feng, J., Jiao, L., Liu, F., Sun, T., & Zhang, X. (2016, March). Unsupervised feature selection based on maximum information and minimum redundancy for hyperspectral images. *Pattern Recognition*, *51*, 295–309. doi: 10.1016/j.patcog.2015.08.018

Feng, X., Park, D. S., Walker, C., Peterson, A. T., Merow, C., & Papeş, M. (2019, October). A checklist for maximizing reproducibility of ecological niche models. *Nature Ecology & Evolution*, *3*(10), 1382–1395. doi: 10.1038/s41559-019-0972-5

Feurer, M., Klein, A., Eggensperger, K., Springenberg, J., Blum, M., & Hutter, F. (2015). Efficient and robust automated machine learning. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, & R. Garnett (Eds.), *Advances in Neural Information*

Processing Systems 28 (pp. 2962–2970). Curran Associates, Inc. Retrieved 2019-06-10, from http://papers.nips.cc/paper/5872-efficient-and-robust-automated-machine-learning.pdf

Filella, I., & Penuelas, J. (1994, May). The red edge position and shape as indicators of plant chlorophyll content, biomass and hydric status. *International Journal of Remote Sensing*, *15*(7), 1459–1470. doi: 10.1080/01431169408954177

Fleuret, F. (2004, December). Fast binary feature selection with conditional mutual information. *The Journal of Machine Learning Research*, *5*, 1531–1555. Retrieved 2019-06-06, from http://dl.acm.org/citation.cfm?id=1005332.1044711

Friedman, J., Hastie, T., & Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, *33*(1), 1–22. doi: 10.18637/jss.v033.i01

Friedman, J. H. (2001, October). Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, *29*(5), 1189–1232. doi: 10.1214/aos/1013203451

Fundazioa, H. (2016). *El bosque vasco en cifras 2017* (Tech. Rep.). Retrieved from https://www.nasdap.ejgv.euskadi.eus/contenidos/informacion/inventario_forestal_2016/es_agripes/adjuntos/El%20bosque%20vasco%20en%20cifras%202017.pdf

Galvão, L. S., Formaggio, A. R., & Tisot, D. A. (2005, February). Discrimination of sugarcane varieties in Southeastern Brazil with EO-1 Hyperion data. *Remote Sensing of Environment*, *94*(4), 523–534. doi: 10.1016/j.rse.2004.11.012

Gamon, J., Peñuelas, J., & Field, C. (1992, July). A narrow-waveband spectral index that tracks diurnal changes in photosynthetic efficiency. *Remote Sensing of Environment*, *41*(1), 35–44. doi: 10.1016/0034-4257(92)90059-s

Ganley, R. J., Watt, M. S., Manning, L., & Iturritxa, E. (2009, November). A global climatic risk assessment of pitch canker disease. *Canadian Journal of Forest Research*, *39*(11), 2246–2256. 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. doi: 10.1007/s00374-003-0579-4

Gao, B.-C. (1996). NDWI—A normalized difference water index for remote sensing of vegetation liquid water from space. *Remote sensing of environment*, *58*(3), 257–266. doi: 10.1016/S0034-4257(96)00067-3

Gao, J., Liang, T., Yin, J., Ge, J., Feng, Q., Wu, C., … Xie, H. (2019, January). Estimation of alpine grassland forage nitrogen coupled with hyperspectral characteristics during different growth periods on the Tibetan plateau. *Remote Sensing*, *11*(18), 2085. doi: 10.3390/rs11182085

Gardener, B. B. M., & Fravel, D. R. (2002). Biological control of plant pathogens: research, commercialization, and application in the USA. *Plant Health Progress*, *3*(1), 17. doi: 10.1094/PHP-2002-0510-01-RV

Garrity, S. R., Eitel, J. U., & Vierling, L. A. (2011, February). Disentangling the relationships between plant pigments and the photochemical reflectance index reveals a new approach for remote estimation of carotenoid content. *Remote Sensing of Environment*, *115*(2), 628–635. doi: 10.1016/j.rse.2010.10.007

Geiß, C., Pelizari, P. A., Schrade, H., Brenning, A., & Taubenböck, H. (2017, November). 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*(11), 2008–2012. doi: 10.1109/lgrs.2017.2747222

Gentleman, R., & Temple Lang, D. (2007,

March). Statistical Analyses and Reproducible Research. *Journal of Computational and Graphical Statistics*, *16*(1), 1–23. doi: 10.1198/106186007X178663

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

Georganos, S., Grippa, T., Vanhuysse, S., Lennert, M., Shimoni, M., Kalogirou, S., & Wolff, E. (2018, March). Less is more: optimizing classification performance through feature selection in a very-high-resolution remote sensing object-based urban application. *GIScience & Remote Sensing*, *55*(2), 221–242. doi: 10.1080/15481603.2017.1408892

Ghosh, M., Adhikary, S., Ghosh, K. K., Sardar, A., Begum, S., & Sarkar, R. (2019, January). Genetic algorithm based cancerous gene identification from microarray data using ensemble of filter methods. *Medical & Biological Engineering & Computing*, *57*(1), 159–176. doi: 10.1007/s11517-018-1874-4

Gil, Y., David, C. H., Demir, I., Essawy, B. T., Fulweiler, R. W., Goodall, J. L., … Yu, X. (2016). Toward the Geoscience Paper of the Future: Best practices for documenting and sharing research from data to software to provenance. *Earth and Space Science*, *3*(10), 388–415. doi: 10.1002/2015EA000136

Gitelson, A., & Merzlyak, M. N. (1994, March). Quantitative estimation of chlorophyll-a using reflectance spectra: Experiments with autumn chestnut and maple leaves. *Journal of Photochemistry and Photobiology B: Biology*, *22*(3), 247–252. doi: 10.1016/1011-1344(93)06963-4

Gitelson, A. A., Buschmann, C., & Lichtenthaler, H. K. (1999, September). The chlorophyll fluorescence ratio F735/F700 as an accurate measure of the chlorophyll content in plants. *Remote Sensing of Environment*, *69*(3), 296–302. doi: 10.1016/s0034-4257(99)00023-1

Gitelson, A. A., Gritz †, Y., & Merzlyak, M. N. (2003, January). Relationships between leaf chlorophyll content and spectral reflectance and algorithms for non-destructive chlorophyll assessment in higher plant leaves. *Journal of Plant Physiology*, *160*(3), 271–282. doi: 10.1078/0176-1617-00887

Gitelson, A. A., Kaufman, Y. J., & Merzlyak, M. N. (1996, December). Use of a green channel in remote sensing of global vegetation from EOS-MODIS. *Remote Sensing of Environment*, *58*(3), 289–298. doi: 10.1016/s0034-4257(96)00072-7

Gitelson, A. A., & Merzlyak, M. N. (1997, August). Remote estimation of chlorophyll content in higher plant leaves. *International Journal of Remote Sensing*, *18*(12), 2691–2697. doi: 10.1080/014311697217558

Gneiting, T., & Raftery, A. E. (2007, March). Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, *102*(477), 359–378. doi: 10.1198/016214506000001437

Goecks, J., Nekrutenko, A., Taylor, J., & The Galaxy Team. (2010, August). Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences. *Genome Biology*, *11*(8), R86. doi: 10.1186/gb-2010-11-8-r86

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

Goldstein, A. (2020, May). *Open Source Licenses in 2020: Trends and Predictions.* Retrieved 2022-03-28, from https://web.archive.org/web/20200503111426/https://resources.whitesourcesoftware.com/blog-whitesource/top-open-source

-licenses-trends-and-predictions

Goodbody, T. R. H., Coops, N. C., Hermosilla, T., Tompalski, P., McCartney, G., & MacLean, D. A. (2018, August). Digital aerial photogrammetry for assessing cumulative spruce budworm defoliation and enhancing forest inventories at a landscape-level. *ISPRS Journal of Photogrammetry and Remote Sensing*, *142*, 1–11. doi: 10.1016/j.isprsjprs.2018.05.012

Gordon, A. D., Breiman, L., Friedman, J. H., Olshen, R. A., & Stone, C. J. (1984, September). Classification and regression trees. *Biometrics*, *40*(3), 874. doi: 10.2307/2530946

Gordon, T. R., Okamoto, D., Storer, A. J., & Wood, D. L. (1998, August). Susceptibility of five landscape pines to pitch canker disease, caused by fusarium subglutinans F. sp. pini. *HortScience*, *33*(5), 868–871. doi: 10.21273/HORTSCI.33.5.868

Gottardini, E., Cristofolini, F., Cristofori, A., Pollastrini, M., Camin, F., & Ferretti, M. (2020, July). A multi-proxy approach reveals common and species-specific features associated with tree defoliation in broadleaved species. *Forest Ecology and Management*, *467*, 118151. doi: 10.1016/j.foreco.2020.118151

Greenwell, B. M., Boehmke, B. C., & McCarthy, A. J. (2018, May). A simple and effective model-based variable importance measure. *arXiv:1805.04755 [cs, stat]*. Retrieved 2020-05-29, from http://arxiv.org/abs/1805.04755

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

Guanter, L., Alonso, L., & Moreno, J. (2005, December). A method for the surface reflectance retrieval from PROBA/CHRIS data over land: application to ESA SPARC campaigns. *IEEE Transactions on Geoscience and Remote Sensing*, *43*(12), 2908–2917. doi: 10.1109/tgrs.2005.857915

Guisan, A., Edwards, T. C., & Hastie, T. (2002, November). Generalized linear and generalized additive models in studies of species distributions: setting the scene. *Ecological Modelling*, *157*(2), 89–100. doi: 10.1016/S0304-3800(02)00204-1

Guisan, A., Weiss, S. B., & Weiss, A. D. (1999, July). GLM versus CCA spatial modeling of plant species distribution. *Plant Ecology*, *143*(1), 107–122. doi: 10.1023/A:1009841519580

Gundersen, O. E., Gil, Y., & Aha, D. W. (2018, September). On reproducible ai: towards reproducible research, open science, and digital scholarship in ai publications. *AI Magazine*, *39*(3), 56–68. doi: 10.1609/aimag.v39i3.2816

Guo, Y., Chung, F.-L., Li, G., & Zhang, L. (2019). Multi-label bioinformatics data classification with ensemble embedded feature selection. *IEEE Access*, *7*, 103863–103875. doi: 10.1109/ACCESS.2019.2931035

Guyon, I., & Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of Machine Learning Research*, *3*(Mar), 1157–1182. Retrieved 2018-08-08, from http://www.jmlr.org/papers/v3/guyon03a.html

Guyot, G., & Baret, F. (1988). Utilisation de la haute resolution spectrale pour suivre l'etat des couverts vegetaux. In T. D. Guyenne & J. J. Hunt (Eds.), *Spectral signatures of objects in remote sensing* (Vol. 287, p. 279). Aussois, France.

Haboudane, D., Miller, J. R., Tremblay, N., Zarco-Tejada, P. J., & Dextraze, L. (2002, August). Integrated narrow-band vegetation indices for prediction of crop chlorophyll content for application to precision agriculture. *Remote Sensing of Environment*, *81*(2-3), 416–426. doi: 10.1016/s0034-4257(02)00018-4

Hais, M., Hellebrandová, K. N., & Šrámek, V. (2019, March). Potential of Landsat spectral indices in regard to the detection of forest health changes due to drought effects. *Journal of Forest Science, 65 (2019)*(No. 2), 70–78. doi:

10.17221/137/2018-JFS

Hall, C. a. S., & Day, J. (1977). *Ecosystem modeling in theory and practice: an introduction with case histories*. John Wiley and Sons,New York, NY. Retrieved 2022-04-24, from https://www.osti.gov/biblio/5835861

Halvorsen, R., Mazzoni, S., Dirksen, J. W., Næsset, E., Gobakken, T., & Ohlson, M. (2016, May). 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: 10.1016/j.ecolmodel.2016.02.021

Hand, D. (1997). *Construction and assessment of classification rules*. New York: Wiley. Retrieved from https://www.wiley.com/en-sg/Construction+and+Assessment+of+Classification+Rules-p-9780471965831

Hartigan, J., & Wong, M. (1979). A k-means clustering algorithm.. doi: 10.2307/2346830

Hartman, J. R., Vaillancourt, L. J., Flowers, J. L., & Bateman, A. M. (2009). Managing Diplodia tip blight of landscape Austrian pines. *Journal of Arboriculture*, *35*(1), 27. doi: 10.48044/jauf.2009.007

Hastie, T., Friedman, J., & Tibshirani, R. (2001). *The elements of statistical learning*. Springer New York. doi: 10.1007/978-0-387-21606-5

Hawryło, P., Bednarz, B., Wezyk, P., & Szostak, M. (2018, January). Estimating defoliation of Scots pine stands using machine learning methods and vegetation indices of Sentinel-2. *European Journal of Remote Sensing*, *51*(1), 194–204. doi: 10.1080/22797254.2017.1417745

Heil, B. J., Hoffman, M. M., Markowetz, F., Lee, S.-I., Greene, C. S., & Hicks, S. C. (2021, October). Reproducibility standards for machine learning in the life sciences. *Nature Methods*, *18*(10), 1132–1135. doi: 10.1038/s41592-021-01256-7

Heim, R. H. J., Wright, I. J., Chang, H.-C., Carnegie, A. J., Pegg, G. S., Lancaster, E. K., … Oldeland, J. (2018). Detecting myrtle rust (Austropuccinia psidii) on lemon myrtle trees using spectral signatures and machine learning. *Plant Pathology*, *67*(5), 1114–1121. doi: 10.1111/ppa.12830

Henelius, A., Puolamäki, K., Boström, H., Asker, L., & Papapetrou, P. (2014, September). A peek into the black box: exploring classifiers by randomization. *Data Mining and Knowledge Discovery*, *28*(5), 1503–1529. doi: 10.1007/s10618-014-0368-8

Hengl, T., de Jesus, J. M., Heuvelink, G. B. M., Gonzalez, M. R., Kilibarda, M., Blagotić, A., … Kempen, B. (2017, February). SoilGrids250m: Global gridded soil information based on machine learning. *PLOS ONE*, *12*(2), e0169748. doi: 10.1371/journal.pone.0169748

Hengl, T., Miller, M. A. E., Križan, J., Shepherd, K. D., Sila, A., Kilibarda, M., … Crouch, J. (2021, March). African soil properties and nutrients mapped at 30 m spatial resolution using two-scale ensemble machine learning. *Scientific Reports*, *11*(1), 6130. doi: 10.1038/s41598-021-85639-y

Hepting, G. H., & Roth, E. R. (1946). Pitch canker, a new disease of some southern pines. *Journal of Forestry*, *44*(10), 742–4. Retrieved 2021-05-23, from https://www.cabdirect.org/cabdirect/abstract/19460602721

Hernández-Clemente, R., Navarro-Cerrillo, R. M., Suárez, L., Morales, F., & Zarco-Tejada, P. J. (2011, September). Assessing structural effects on PRI for stress detection in conifer forests. *Remote Sensing of Environment*, *115*(9), 2360–2375. doi: 10.1016/j.rse.2011.04.036

Hernández-Clemente, R., Navarro-Cerrillo, R. M., & Zarco-Tejada, P. J. (2012, December). Carotenoid content estimation in a heterogeneous conifer forest using narrow-band indices and PROSPECTDART simulations. *Remote Sensing of Environment*, *127*, 298–315.

doi: 10.1016/j.rse.2012.09.014

Hernandez-Escribano, L., Iturritxa, E., Aragonés, A., Mesanza, N., Berbegal, M., Raposo, R., & Elvira-Recuenco, M. (2018, March). Root infection of canker pathogens, Fusarium circinatum and Diplodia sapinea, in asymptomatic trees in Pinus radiata and Pinus pinaster plantations. *Forests*, *9*(3), 128. doi: 10.3390/f9030128

Hijmans, R. J., Phillips, S., Leathwick, J., & Elith, J. (2020). *dismo: Species distribution modeling* [manual]. Retrieved from https://CRAN.R-project.org/package=dismo

Hlebarska, S., & Georgieva, M. (2018). Distribution of the invasive pathogen Diplodia sapinea on pinus spp. in Bulgaria. *Conference proceedings, 90 years forest research institute - for the society and nature, Sofia, Bulgaria, 24-26 October 2018*, 61–70. Retrieved 2021-06-24, from https://www.cabdirect.org/cabdirect/abstract/20183376613

Hoffmann, F., Bertram, T., Mikut, R., Reischl, M., & Nelles, O. (2019). Benchmarking in classification and regression. *WIREs Data Mining and Knowledge Discovery*, *9*(5), e1318. doi: 10.1002/widm.1318

Holmes, T. P., Aukema, J. E., Von Holle, B., Liebhold, A., & Sills, E. (2009). Economic impacts of invasive species in forest past, present, and future. *In: The Year In Ecology and Conservation Biology, 2009. Ann. NY Acad. Sci. 1162: 18-38.*, *1162*, 18–38. doi: 10.1111/j.1749-6632.2009.04446.x

Horler, D. N. H., Dockray, M., & Barber, J. (1983, January). The red edge of plant leaf reflectance. *International Journal of Remote Sensing*, *4*(2), 273–288. doi: 10.1080/01431168308948546

Huete, A. (1988, August). A soil-adjusted vegetation index (SAVI). *Remote Sensing of Environment*, *25*(3), 295–309. doi: 10.1016/0034-4257(88)90106-x

Huete, A. R. (2012). Vegetation indices, remote sensing and forest monitoring. *Geography Compass*, *6*(9), 513–532. doi: 10.1111/j.1749-8198.2012.00507.x

Huete, A. R., Liu, H. Q., Batchily, K., & van Leeuwen, W. (1997, March). A comparison of vegetation indices over a global set of TM images for EOS-MODIS. *Remote Sensing of Environment*, *59*(3), 440–451. doi: 10.1016/s0034-4257(96)00112-5

Hunt, E., & Rock, B. (1989, October). Detection of changes in leaf water content using Near- and Middle-Infrared reflectances. *Remote Sensing of Environment*, *30*(1), 43–54. doi: 10.1016/0034-4257(89)90046-1

Hunt, E. R., Doraiswamy, P. C., McMurtrey, J. E., Daughtry, C. S., Perry, E. M., & Akhmedov, B. (2013, April). A visible band index for remote sensing leaf chlorophyll content at the canopy scale. *International Journal of Applied Earth Observation and Geoinformation*, *21*, 103–112. doi: 10.1016/j.jag.2012.07.020

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

Hyndman, R. J., & Koehler, A. B. (2006). Another look at measures of forecast accuracy - ScienceDirect. *International Journal of Forecasting*, *22*(4), 679–688. doi: 10.1016/j.ijforecast.2006.03.001

Innes, J. (1993). Methods to estimate forest health. *Silva Fennica*, *27*(2). doi: 10.14214/sf.a15668

IPCC. (2013). Summary for policymakers [Book Section]. In T. Stocker et al. (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* (pp. 1–30). Cambridge, United Kingdom and New York, NY, USA: Cambridge University Press. doi: 10.1017/CBO9781107415324.004

Iturritxa, E., Ganley, R. J., Raposo, R., García-Serna, I., Mesanza, N., Kirkpatrick, S. C., & Gordon, T. R. (2013). Resistance levels of Spanish conifers against Fusarium circinatum and Diplodia pinea. *Forest Pathology*, *43*(6), 488–495. doi: 10.1111/efp.12061

Iturritxa, E., Ganley, R. J., Wright, J., Heppe, E., Steenkamp, E. T., Gordon, T. R., & Wingfield, M. J. (2011, March). A genetically homogenous population of Fusarium circinatum causes pitch canker of Pinus radiata in the Basque Country, Spain. *Fungal Biology*, *115*(3), 288–295. doi: 10.1016/j.funbio.2010.12.014

Iturritxa, E., Mesanza, N., & Brenning, A. (2014). Spatial analysis of the risk of major forest diseases in Monterey pine plantations. *Plant Pathology*, *64*(4), 880–889. doi: 10.1111/ppa.12328

Iturritxa, E., Trask, T., Mesanza, N., Raposo, R., Elvira-Recuenco, M., & Patten, C. L. (2017, January). Biocontrol of Fusarium circinatum infection of young Pinus radiata trees. *Forests*, *8*(2), 32. doi: 10.3390/f8020032

Ives, A. R., Zhu, L., Wang, F., Zhu, J., Morrow, C. J., & Radeloff, V. C. (2021, December). Statistical inference for trends in spatiotemporal data. *Remote Sensing of Environment*, *266*, 112678. doi: 10.1016/j.rse.2021.112678

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

Jarnevich, C. S., Talbert, M., Morisette, J., Aldridge, C., Brown, C. S., Kumar, S., … Holcombe, T. (2017, November). Minimizing effects of methodological decisions on interpretation and prediction in species distribution studies: An example with background selection. *Ecological Modelling*, *363*, 48–56. doi: 10.1016/j.ecolmodel.2017.08.017

Jensen, D. A., Rao, M., Zhang, J., Grøn, M., Tian, S., Ma, K., & Svenning, J.-C. (2021, February). The potential for using rare, native species in reforestation– A case study of yews (Taxaceae) in China. *Forest Ecology and Management*, *482*, 118816. doi: 10.1016/j.foreco.2020.118816

Jiang, Y., Wang, T., de Bie, C. A. J. M., Skidmore, A. K., Liu, X., Song, S., … Shao, X. (2014, March). Satellite-derived vegetation indices contribute significantly to the prediction of epiphyllous liverworts. *Ecological Indicators*, *38*, 72–80. doi: 10.1016/j.ecolind.2013.10.024

Jiménez-Donaire, M. d. P., Giráldez, J. V., & Vanwalleghem, T. (2020, November). Impact of Climate Change on Agricultural Droughts in Spain. *Water*, *12*(11), 3214. doi: 10.3390/w12113214

Johnson, J. B., & Omland, K. S. (2004, February). Model selection in ecology and evolution. *Trends in Ecology & Evolution*, *19*(2), 101–108. doi: 10.1016/j.tree.2003.10.013

Johnstone, I. M., & Titterington, D. M. (2009, November). Statistical challenges of high-dimensional data. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *367*(1906), 4237–4253. doi: 10.1098/rsta.2009.0159

Jolliffe, I., & Cadima, J. (2016, April). Principal component analysis: a review and recent developments. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *374*(2065), 20150202. doi: 10.1098/rsta.2015.0202

Jones, D. R., Schonlau, M., & Welch, W. J. (1998, December 01). Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, *13*(4), 455–492. doi: 10.1023/a:1008306431147

Jordan, C. F. (1969, July). Derivation of leaf-area index from quality of light on the forest floor. *Ecology*, *50*(4), 663–666. doi: 10.2307/1936256

Kälin, U., Lang, N., Hug, C., Gessler, A., & Wegner, J. D. (2019, March). Defoliation estima-

tion of forest trees from ground-level images. *Remote Sensing of Environment*, *223*, 143–153. doi: 10.1016/j.rse.2018.12.021

Kang, Y., Meng, Q., Liu, M., Zou, Y., & Wang, X. (2021, January). Crop classification based on red edge features analysis of GF-6 WFV data. *Sensors*, *21*(13), 4328. doi: 10.3390/s21134328

Kanzler, A., Nel, A., & Ford, C. (2014, May). Development and commercialisation of the Pinus patula × P. tecunumanii hybrid in response to the threat of Fusarium circinatum. *New Forests*, *45*(3), 417–437. doi: 10.1007/s11056-014-9412-1

Karasiak, N. (2020). Museo ToolBox: A Python library for remote sensing including a new way to handle rasters. *Journal of Open Source Software*, *5*(48), 1978. doi: 10.21105/joss.01978

Karasiak, N., Dejoux, J.-F., Monteil, C., & Sheeren, D. (2021, April). Spatial dependence between training and test sets: another pitfall of classification accuracy assessment in remote sensing. *Machine Learning*. doi: 10.1007/s10994-021-05972-1

Karatzoglou, A., Smola, A., Hornik, K., & Zeileis, A. (2004). kernlab – An S4 Package for Kernel Methods in R. *Journal of Statistical Software*, *11*(9), 1–20. doi: 10.18637/jss.v011.i09

Kasurak, A., Kelly, R., & Brenning, A. (2011). Linear mixed modelling of snow distribution in the central Yukon. *Hydrological Processes*, *25*(21), 3332–3346. doi: 10.1002/hyp.8168

Kaya, A. G. A., Yeltekin, Ş., Lehtijarvi, T. D., Lehtijarvi, A., & Woodward, S. (2019). Severity of Diplodia shoot blight (caused by Diplodia sapinea) was greatest on Pinus sylvestris and Pinus nigra in a plantation containing five pine species. *Phytopathologia Mediterranea*, *58*(2), 249–259. Retrieved from https://doaj.org/article/3d9b8162586c480caea4dd3e04753868

Kayet, N., Pathak, K., Chakrabarty, A., Singh, C. P., Chowdary, V. M., Kumar, S., & Sahoo, S. (2019, November). Forest health assessment for geo-environmental planning and management in hilltop mining areas using Hyperion and Landsat data. *Ecological Indicators*, *106*, 105471. doi: 10.1016/j.ecolind.2019.105471

Keller, S., Braun, A. C., Hinz, S., & Weinmann, M. (2016, August). Investigation of the impact of dimensionality reduction and feature selection on the classification of hyperspectral EnMAP data. In *2016 8th Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)* (pp. 1–5). doi: 10.1109/WHISPERS.2016.8071759

Kira, K., & Rendell, L. A. (1992, July). The feature selection problem: traditional methods and a new algorithm. In *Proceedings of the tenth national conference on Artificial intelligence* (pp. 129–134). AAAI Press. Retrieved 2019-06-23, from http://dl.acm.org/citation.cfm?id=1867135.1867155

Koenig, W. D. (1999, January). Spatial autocorrelation of ecological phenomena. *Trends in Ecology & Evolution*, *14*(1), 22–26. doi: 10.1016/S0169-5347(98)01533-X

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

Kohavi, R., & John, G. H. (1997, December). Wrappers for feature subset selection. *Artificial Intelligence*, *97*(1), 273–324. doi: 10/fngm9z

Konkol, M., Kray, C., & Pfeiffer, M. (2019, February). Computational reproducibility in geoscientific papers: Insights from a series of studies with geoscientists and a reproduction study. *International Journal of Geographical Information Science*, *33*(2), 408–429. doi: 10.1080/13658816.2018.1508687

Kopczewska, K. (2021, December). Spatial machine learning: new opportunities for regional science. *The Annals of Regional Science*. doi: 10.1007/s00168-021-01101-x

Kuhn, M. (2017). *caret: Classification and Regression Training*. Retrieved from https://CRAN.R

-project.org/package=caret

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

Kuhn, M., & Wickham, H. (2020). *tidymodels: a collection of packages for modeling and machine learning using tidyverse principles.* [manual]. Retrieved from https://www.tidymodels.org

Kursa, M. B. (2018). *praznik: collection of information-based feature selection filters.* Retrieved from https://CRAN.R-project.org/package=praznik

Landau, W. M. (2018). The drake R package: a pipeline toolkit for reproducibility and high-performance computing. *Journal of Open Source Software*, *3*(21). Retrieved from https://doi.org/10.21105/joss.00550

Landau, W. M. (2021). The targets R package: a dynamic Make-like function-oriented pipeline toolkit for reproducibility and high-performance computing. *Journal of Open Source Software*, *6*(57), 2959. Retrieved from https://doi.org/10.21105/joss.02959

Lang, M., Au, Q., Coors, S., & Schratz, P. (2020). *mlr3learners: Recommended Learners for 'mlr3'* [manual]. Retrieved from https://CRAN.R-project.org/package=mlr3learners

Lang, M., Binder, M., Richter, J., Schratz, P., Pfisterer, F., Coors, S., … Bischl, B. (2019, December). mlr3: A modern object-oriented machine learning framework in R. *Journal of Open Source Software*. doi: 10.21105/joss.01903

Lantz, B. (2019). *Machine Learning with R: Expert techniques for predictive modeling* (Third ed.). Packt Publishing Ltd. Retrieved from https://www.scribd.com/book/498883274

La Porta, N., Capretti, P., Thomsen, I. M., Kasanen, R., Hietala, A. M., & Von Weissenberg, K. (2008, April). Forest pathogens with higher damage potential due to climate change in Europe. *Canadian Journal of Plant Pathology*, *30*(2), 177–195. doi: 10.1080/07060661.2008.10540534

Lary, D. J., Alavi, A. H., Gandomi, A. H., & Walker, A. L. (2016, January). Machine learning in geosciences and remote sensing. *Geoscience Frontiers*, *7*(1), 3–10. doi: 10.1016/j.gsf.2015.07.003

Legendre, P. (1993, September). Spatial autocorrelation: Trouble or new paradigm? *Ecology*, *74*(6), 1659–1673. doi: 10.2307/1939924

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

Lehnert, L. W., Meyer, H., & Bendix, J. (2018). *hsdar: Manage, analyse and simulate hyperspectral data in R.* Retrieved from https://CRAN.R-project.org/package=hsdar

le Maire, G., François, C., & Dufrêne, E. (2004, January). Towards universal broad leaf chlorophyll indices using PROSPECT simulated database and hyperspectral reflectance measurements. *Remote Sensing of Environment*, *89*(1), 1–28. doi: 10.1016/j.rse.2003.09.004

Lemaire, G., Francois, C., Soudani, K., Berveiller, D., Pontailler, J., Breda, N., … Dufrene, E. (2008, October). Calibration and validation of hyperspectral indices for the estimation of broadleaved forest leaf chlorophyll content, leaf mass per area, leaf area index and leaf canopy biomass. *Remote Sensing of Environment*, *112*(10), 3846–3864. doi: 10.1016/j.rse.2008.06.005

Levin, N., Kidron, G. J., & Ben-Dor, E. (2007, August). Surface properties of stabilizing coastal dunes: combining spectral and field analyses. *Sedimentology*, *54*(4), 771–788. doi: 10.1111/j.1365-3091.2007.00859.x

Lichstein, J. W., Simons, T. R., Shriner, S. A., & Franzreb, K. E. (2002). Spatial Au-

tocorrelation and Autoregressive Models in Ecology. *Ecological Monographs*, *72*(3), 445–463. doi: 10.1890/0012-9615(2002)072[0445:SAAAMI]2.0.CO;2

Lichtenthaler, H., Lang, M., Sowinska, M., Heisel, F., & Miehé, J. (1996, January). Detection of vegetation stress via a new high resolution fluorescence imaging system. *Journal of Plant Physiology*, *148*(5), 599–612. doi: 10.1016/s0176-1617(96)80081-2

Liebhold, A., & Sharov, A. A. (1998). Testing for correlation in the presence of spatial autocorrelation in insect count data. In *Population and community ecology for insect management and conservation* (pp. 111–118). CRC Press. Retrieved from https://doi.org/10.1201%2F9780429333422-10

Lin, H., Yan, E., Wang, G., & Song, R. (2014, June). Analysis of hyperspectral bands for the health diagnosis of tree species. In *2014 Third International Workshop on Earth Observation and Remote Sensing Applications (EORSA)* (pp. 448–451). doi: 10.1109/EORSA.2014.6927931

Lin, Y.-H., Ko, T.-M., Chuang, T.-R., & Lin, K.-J. (2006). Open source licenses and the creative commons framework: License selection and comparison. *Journal of information science and engineering*, *22*(1), 1–17.

Lobell, D. B., Asner, G. P., Law, B. E., & Treuhaft, R. N. (2001, March). Subpixel canopy cover estimation of coniferous forests in Oregon using SWIR imaging spectrometry. *Journal of Geophysical Research: Atmospheres*, *106*(D6), 5151–5160. doi: 10.1029/2000jd900739

Loehle, C. (2018, September). Disequilibrium and relaxation times for species responses to climate change. *Ecological Modelling*, *384*, 23–29. doi: 10.1016/j.ecolmodel.2018.06.004

Loo, J. A. (2009). Ecological impacts of non-indigenous invasive fungi as forest pathogens. In D. W. Langor & J. Sweeney (Eds.), *Ecological Impacts of Non-Native Invertebrates and Fungi on Terrestrial Ecosystems* (pp. 81–96). Dordrecht: Springer Netherlands. doi: 10.1007/978-1-4020-9680-8_6

Lorenzo, M. N., & Alvarez, I. (2020, June). Climate change patterns in precipitation over Spain using CORDEX projections for 2021–2050. *Science of The Total Environment*, *723*, 138024. doi: 10.1016/j.scitotenv.2020.138024

Lovelace, R., Nowosad, J., & Muenchow, J. (2019). *Geocomputation with R*. CRC Press. Retrieved from https://geocompr.robinlovelace.net/

Lovett, G. M., Weiss, M., Liebhold, A. M., Holmes, T. P., Leung, B., Lambert, K. F., … Weldy, T. (2016). Nonnative forest insects and pathogens in the United States: Impacts and policy options. *Ecological Applications*, *26*(5), 1437–1455. doi: 10.1890/15-1176

Luchi, N., Longa, C. M. O., Danti, R., Capretti, P., & Maresi, G. (2014). Diplodia sapinea: the main fungal species involved in the colonization of pine shoots in Italy. *Forest Pathology*, *44*(5), 372–381. doi: 10.1111/efp.12109

Ludwig, M., Morgenthal, T., Detsch, F., Higginbottom, T. P., Lezama Valdes, M., Nauß, T., & Meyer, H. (2019, March). Machine learning and multi-sensor based modelling of woody vegetation in the Molopo area, South Africa. *Remote Sensing of Environment*, *222*, 195–203. doi: 10.1016/j.rse.2018.12.019

Lundberg, S. M., & Lee, S.-I. (2017). A Unified Approach to Interpreting Model Predictions. In *Advances in Neural Information Processing Systems* (Vol. 30). Curran Associates, Inc. Retrieved 2022-05-14, from https://proceedings.neurips.cc/paper/2017/hash/8a20a8621978632d76c43dfd28b67767-Abstract.html

Ma, Y., Wu, H., Wang, L., Huang, B., Ranjan, R., Zomaya, A., & Jie, W. (2015, October). Remote sensing big data computing: Challenges and opportunities. *Future Gen-*

eration Computer Systems, *51*, 47–60. doi: 10.1016/j.future.2014.10.029

Maccioni, A., Agati, G., & Mazzinghi, P. (2001, August). New vegetation indices for remote measurement of chlorophylls based on leaf directional reflectance spectra. *Journal of Photochemistry and Photobiology B: Biology*, *61*(1-2), 52–61. doi: 10.1016/s1011-1344(01)00145-2

MacLean, D. A., & Lidstone, R. G. (1982, September). Defoliation by spruce budworm: estimation by ocular and shoot-count methods and variability among branches, trees, and stands. *Canadian Journal of Forest Research*, *12*(3), 582–594. doi: 10.1139/x82-090

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. Retrieved from http://proceedings.mlr.press/v64/malkomes_bayesian_2016.html

Manzanos, T., Aragonés, A., & Iturritxa, E. (2019). Genotypic diversity and distribution of Sphaeropsis sapinea within Pinus radiata trees from northern Spain. *Forest Pathology*, *49*(6), e12550. doi: 10.1111/efp.12550

Manzanos, T., Stanosz, G. R., Smith, D. R., Muenchow, J., Schratz, P., Brenning, A., … Iturritxa, E. (2019, February). Mating type ratios and pathogenicity in Diplodia shoot blight fungi populations: Comparative analysis. *Forest Pathology*, *49*(1), e12475. doi: 10.1111/efp.12475

Martin, M. E., Plourde, L. C., Ollinger, S. V., Smith, M. L., & McNeil, B. E. (2008, September). A generalizable method for remote sensing of canopy nitrogen across a wide range of forest ecosystems. *Remote Sensing of Environment*, *112*(9), 3511–3519. doi: 10.1016/j.rse.2008.04.008

Martín-García, J., Zas, R., Solla, A., Woodward, S., Hantula, J., Vainio, E. J., … Diez, J. J. (2019). Environmentally friendly methods for controlling pine pitch canker. *Plant Pathology*, *68*(5), 843–860. doi: 10.1111/ppa.13009

Mascaro, J., Asner, G. P., Knapp, D. E., Kennedy-Bowdoin, T., Martin, R. E., Anderson, C., … Chadwick, K. D. (2014, January). A tale of two "forests": random forest machine learning aids tropical forest carbon mapping. *PLOS ONE*, *9*(1), e85993. doi: 10/f5vmrb

McMurtrey, J., Chappelle, E., Kim, M., Meisinger, J., & Corp, L. (1994, January). Distinguishing nitrogen fertilization levels in field corn (Zea mays L.) with actively induced fluorescence and passive reflectance measurements. *Remote Sensing of Environment*, *47*(1), 36–44. doi: 10.1016/0034-4257(94)90125-2

Mead, D. J. (2013). *Sustainable management of Pinus radiata plantations*. Food and agriculture organization of the United nations (FAO). Retrieved from https://www.fao.org/3/i3274e/i3274e01.pdf

Meng, J., Li, S., Wang, W., Liu, Q., Xie, S., & Ma, W. (2016, September). Mapping forest health using spectral and textural information extracted from SPOT-5 satellite images. *Remote Sensing*, *8*(9), 719. doi: 10.3390/rs8090719

Meng, R., Dennison, P. E., Zhao, F., Shendryk, I., Rickert, A., Hanavan, R. P., … Serbin, S. P. (2018, September). Mapping canopy defoliation by herbivorous insects at the individual tree level using bi-temporal airborne imaging spectroscopy and LiDAR measurements. *Remote Sensing of Environment*, *215*, 170–183. doi: 10.1016/j.rse.2018.06.008

Merkel, D. (2014). Docker: lightweight linux containers for consistent development and deployment. *Linux journal*, *2014*(239), 2.

Merzlyak, M. N., Gitelson, A. A., Chivkunova, O. B., & Rakitin, V. Y. (1999, May). Non-destructive optical detection of pigment changes during leaf senescence and fruit

ripening. *Physiologia Plantarum*, *106*(1), 135–141. doi: 10.1034/j.1399-3054.1999.106119.x

Mesanza, N., Hernández, M., Raposo, R., & Iturritxa, E. (2021, January). First Report of Mycosphaerella dearnessii, Teleomorph of Lecanosticta acicola, in Europe. *Plant Health Progress*, *22*(4), 565–566. doi: 10.1094/PHP-03-21-0060-BR

Mesanza, N., Iturritxa, E., & Patten, C. L. (2016, October). Native rhizobacteria as biocontrol agents of Heterobasidion annosum s.s. and Armillaria mellea infection of Pinus radiata. *Biological Control*, *101*, 8–16. doi: 10.1016/j.biocontrol.2016.06.003

Mets, K. D., Armenteras, D., & Dávalos, L. M. (2017, May). Spatial autocorrelation reduces model precision and predictive power in deforestation analyses. *Ecosphere*, *8*(5), e01824. doi: 10.1002/ecs2.1824

Meyer, H. (2020). *CAST: 'caret' applications for spatial-temporal models* [manual]. Retrieved from https://CRAN.R-project.org/package=CAST

Meyer, H., & Pebesma, E. (2021). Predicting into unknown space? Estimating the area of applicability of spatial prediction models. *Methods in Ecology and Evolution*, *12*(9), 1620–1633. doi: 10.1111/2041-210X.13650

Meyer, H., Reudenbach, C., Hengl, T., Katurji, M., & Nauss, T. (2018, March). 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

Michel, M. (2003). *El Pino radiata (Pinus radiata D. Don) en la historia forestal de la comunidad autónoma de Euskadi: análisis de un proceso de forestalismo intensivo* (Doctoral dissertation, Universidad Politécnica de Madrid). Retrieved from https://oa.upm.es/404/

Micheletti, N., Foresti, L., Robert, S., Leuenberger, M., Pedrazzini, A., Jaboyedoff, M., & Kanevski, M. (2013, December). Machine learning feature selection methods for landslide susceptibility mapping. *Mathematical Geosciences*, *46*(1), 33–57. doi: 10.1007/s11004-013-9511-0

Møller, A. B., Mulder, V. L., Heuvelink, G. B. M., Jacobsen, N. M., & Greve, M. H. (2021, April). Can we use machine learning for agricultural land suitability assessment? *Agronomy*, *11*(4), 703. doi: 10.3390/agronomy11040703

Molnar, C. (2019). *Interpretable machine learning - a guide for making black box models explainable*. Munich, Germany: self-published. Retrieved from https://christophm.github.io/interpretable-ml-book/

Molnar, C., König, G., Bischl, B., & Casalicchio, G. (2020, June). *Model-agnostic feature importance and effects with dependent features – a conditional subgroup approach.* Retrieved 2021-12-28, from https://arxiv.org/abs/2006.04628v2

Molnar, C., König, G., Herbinger, J., Freiesleben, T., Dandl, S., Scholbeck, C. A., … Bischl, B. (2022). General Pitfalls of Model-Agnostic Interpretation Methods for Machine Learning Models. In A. Holzinger, R. Goebel, R. Fong, T. Moon, K.-R. Müller, & W. Samek (Eds.), *xxAI - Beyond Explainable AI: International Workshop, Held in Conjunction with ICML 2020, July 18, 2020, Vienna, Austria, Revised and Extended Papers* (pp. 39–68). Cham: Springer International Publishing. doi: 10.1007/978-3-031-04083-2_4

Morellato, L. P. C., Camargo, M. G. G., D'Eça Neves, F. F., Luize, B. G., Mantovani, A., & Hudson, I. L. (2010). The influence of sampling method, sample size, and frequency of observations on plant phenological patterns and interpretation in tropical forest trees. In I. L. Hudson & M. R. Keatley (Eds.), *Phenological Research: Methods for Environmen-*

tal and Climate Change Analysis (pp. 99–121). Dordrecht: Springer Netherlands. doi: 10.1007/978-90-481-3335-2_5

Morera, A., Martínez de Aragón, J., Bonet, J. A., Liang, J., & de-Miguel, S. (2021, March). Performance of statistical and machine learning-based methods for predicting biogeographical patterns of fungal productivity in forest ecosystems. *Forest Ecosystems*, *8*(1), 21. doi: 10.1186/s40663-021-00297-w

Möykkynen, T., Capretti, P., & Pukkala, T. (2015, March). Modelling the potential spread of Fusarium circinatum, the causal agent of pitch canker in Europe. *Annals of Forest Science*, *72*(2), 169–181. doi: 10.1007/s13595-014-0412-2

Muenchow, J., Brenning, A., & Richter, M. (2012). Geomorphic process rates of landslides along a humidity gradient in the tropical Andes. *Geomorphology*, *139–140*, 271–284. doi: 10.1016/j.geomorph.2011.10.029

Muenchow, J., Dieker, P., Kluge, J., Kessler, M., & von Wehrden, H. (2018, February). A review of ecological gradient research in the Tropics: identifying research gaps, future directions, and conservation priorities. *Biodiversity and Conservation*, *27*(2), 273–285. doi: 10.1007/s10531-017-1465-y

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*(6), 1154–1166. doi: 10.1111/jvs.12038

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* (E. J. A. Z. Nagusia, Ed.). Eusko Jaurlaritzaren Argitalpen Zerbitzu Nagusia, Servicio Central de Publicaciones del Gobierno VAsco, C/ Donostia-San Sebastián 1, 01010 Vitoria-Gasteiz. Retrieved from https://dialnet.unirioja.es/servlet/libro?codigo=770126

Müller, M. M., Hantula, J., Wingfield, M., & Drenkhan, R. (2019). Diplodia sapinea found on Scots pine in Finland. *Forest Pathology*, *49*(1), e12483. doi: 10.1111/efp.12483

Murase, H., Nagashima, H., Yonezaki, S., Matsukura, R., & Kitakado, T. (2009, July). 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*(6), 1417–1424. doi: 10.1093/icesjms/fsp105

Muscarella, R., Galante, P. J., Soley-Guardia, M., Boria, R. A., Kass, J. M., Uriarte, M., & Anderson, R. P. (2014). ENMeval: An R package for conducting spatially independent evaluations and estimating optimal model complexity for Maxent ecological niche models. *Methods in Ecology and Evolution*, *5*(11), 1198–1205. doi: 10.1111/2041-210X.12261

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*(1), 44. doi: 10.1007/s10661-015-5049-6

Nagler, P. L., Inoue, Y., Glenn, E. P., Russ, A. L., & Daughtry, C. S. T. (2003, October). Cellulose absorption index (CAI) to quantify mixed soil–plant litter scenes. *Remote Sensing of Environment*, *87*(2-3), 310–325. doi: 10.1016/j.rse.2003.06.001

National Academies of Sciences, & Medicine. (2019). *Reproducibility and replicability in science*. Washington, DC: The National Academies Press. doi: 10.17226/25303

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. Retrieved from https://www.miteco.gob.es/es/ceneam/recursos/pag-web/conservacion/atlas.aspx

Nirenberg, H. I., & O'Donnell, K. (2018, August). New Fusarium species and combinations within the Gibberella fujikuroi species complex. *Mycologia*. Retrieved 2021-05-25, from https://www.tandfonline.com/doi/abs/10.1080/00275514.1998.12026929

Nüst, D., & Eglen, S. J. (2021, July). CODECHECK: an Open Science initiative for the independent execution of computations underlying research articles during peer review to improve reproducibility. *F1000Research*, *10*, 253. doi: 10.12688/f1000research.51738.2

Nüst, D., & Hinz, M. (2019, August). containerit: Generating Dockerfiles for reproducible research with R. *Journal of Open Source Software*, *4*(40), 1603. doi: 10.21105/joss.01603

Nüst, D., & Pebesma, E. (2021, July). Practical Reproducibility in Geography and Geosciences. *Annals of the American Association of Geographers*, *111*(5), 1300–1310. doi: 10.1080/24694452.2020.1806028

O'Laughlin, J., Livingston, R. L., Thier, R., Thornton, J. P., Toweill, D. E., & Morelan, L. (1994). Defining and Measuring Forest Health. *Journal of Sustainable Forestry*. doi: 10.1300/J091v02n01_03

Oliva, J., Stenlid, J., Grönkvist-Wichmann, L., Wahlström, K., Jonsson, M., Drobyshev, I., & Stenström, E. (2016, November). Pathogen-induced defoliation of Pinus sylvestris leads to tree decline and death from secondary biotic factors. *Forest Ecology and Management*, *379*, 273–280. doi: 10.1016/j.foreco.2016.08.011

Olson, R. S., La Cava, W., Orzechowski, P., Urbanowicz, R. J., & Moore, J. H. (2017, December). PMLB: a large benchmark suite for machine learning evaluation and comparison. *BioData Mining*, *10*. doi: 10.1186/s13040-017-0154-4

Oppelt, N., & Mauser, W. (2004, January). Hyperspectral monitoring of physiological parameters of wheat during a vegetation period using AVIS data. *International Journal of Remote Sensing*, *25*(1), 145–159. doi: 10.1080/0143116031000115300

Osimo, D., & Switters, J. (2019). *Recognising the importance of software in research: Research Software Engineers (RSEs), a UK example*. LU: Publications Office of the European Union. Retrieved 2021-12-21, from https://data.europa.eu/doi/10.2777/787013

O'Sullivan, D., & Unwin, D. (2003). *Geographic information analysis*. John Wiley & Sons. Retrieved from https://books.google.ch/books/about/Geographic_Information_Analysis.html?id=HvlMAQAAIAAJ&redir_esc=y

Pal, M., & Foody, G. M. (2010, May). Feature selection for classification of hyperspectral data by SVM. *IEEE Transactions on Geoscience and Remote Sensing*, *48*(5), 2297–2307. doi: 10.1109/TGRS.2009.2039484

Palmer, M., Stewart, E., & Wingfield, M. (1987). Variation among isolates of Sphaeropsis sapinea in the north central United States. *Phytopathology*, *77*(6), 944–948. Retrieved from https://www.apsnet.org/publications/phytopathology/backissues/Documents/1987Articles/Phyto77n06_944.PDF

PANGAEA: Data Publisher for Earth & Environmental Science. (2022, March). Retrieved 2022-03-27, from https://pangaea.de/

Paulpietersburg, N. (1980). Association of Diplodia pinea with a root disease of

pines in South Africa. *Plant Disease*, *64*(2), 221. Retrieved from https://www.apsnet.org/publications/plantdisease/backissues/Documents/1980Articles/PlantDisease64n02_221.pdf

Pearce, J., & Derrick, B. (2019, October). Preliminary Testing: The Devil of Statistics? *Reinvention: an International Journal of Undergraduate Research*, *12*(2). doi: 10.31273/reinvention.v12i2.339

Pearson, K. (1901, November). LIII. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, *2*(11), 559–572. doi: 10/dd63n4

Pebesma, E. (2018). Simple features for R: standardized support for spatial vector data. *The R Journal*, *10*(1), 439–446. Retrieved 2021-05-08, from https://journal.r-project.org/archive/2018/RJ-2018-009/index.html

Pebesma, E. J., & Bivand, R. S. (2005, November). Classes and methods for spatial data in R. *R News*, *5*(2), 9–13. Retrieved from https://CRAN.R-project.org/doc/Rnews/

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., … Duchesnay, É. (2011). Scikit-learn: machine learning in python. *Journal of Machine Learning Research*, *12*(85), 2825–2830. Retrieved 2021-05-25, from http://jmlr.org/papers/v12/pedregosa11a.html

Peña, M., & Brenning, A. (2015, December). Assessing fruit-tree crop classification from Landsat-8 time series for the Maipo Valley, Chile. *Remote Sensing of Environment*, *171*, 234–244. doi: 10.1016/j.rse.2015.10.029

Peña, M., Liao, R., & Brenning, A. (2017). Using spectrotemporal indices to improve the fruit-tree crop classification accuracy. *ISPRS Journal of Photogrammetry and Remote Sensing*, *128*, 158–169. doi: 10.1016/j.isprsjprs.2017.03.019

Penuelas, J., Filella, I., Lloret, P., Munoz, F., & Vilaleliu, M. (1995, September). Reflectance assessment of mite effects on apple trees. *International Journal of Remote Sensing*, *16*(14), 2727–2733. doi: 10.1080/01431169508954588

Peñuelas, J., Gamon, J. A., Fredeen, A. L., Merino, J., & Field, C. B. (1994, May). Reflectance indices associated with physiological changes in nitrogen- and water-limited sunflower leaves. *Remote Sensing of Environment*, *48*(2), 135–146. doi: 10.1016/0034-4257(94)90136-8

Penuelas, J., Pinol, J., Ogaya, R., & Filella, I. (1997, September). Estimation of plant water concentration by the reflectance Water Index WI (R900/R970). *International Journal of Remote Sensing*, *18*(13), 2869–2875. doi: 10.1080/014311697217396

Pérez-Sierra, A., Landeras, E., León, M., Berbegal, M., García-Jiménez, J., & Armengol, J. (2007, July). Characterization of Fusarium circinatum from Pinus spp. in northern Spain. *Mycological Research*, *111*(7), 832–839. doi: 10.1016/j.mycres.2007.05.009

Pes, B. (2020, May). Ensemble feature selection for high-dimensional data: a stability analysis across multiple domains. *Neural Computing and Applications*, *32*(10), 5951–5973. doi: 10.1007/s00521-019-04082-3

Petermann, E., Meyer, H., Nussbaum, M., & Bossew, P. (2021, February). Mapping the geogenic radon potential for Germany by machine learning. *Science of The Total Environment*, *754*, 142291. doi: 10.1016/j.scitotenv.2020.142291

Pfenning, L. H., Costa, S. d. S., de Melo, M. P., Costa, H., Ventura, J. A., Auer, C. G., & dos Santos, Á. F. (2014, June). First report and characterization of Fusarium circinatum, the causal agent of pitch canker in Brazil. *Tropical Plant Pathology*, *39*, 210–216. doi: 10.1590/S1982-56762014000300004

Pinto, J., Powell, S., Peterson, R., Rosalen, D., & Fernandes, O. (2020, January). Detection of defoliation injury in peanut with hyperspectral proximal remote sensing. *Remote Sensing*, *12*(22), 3828. doi: 10.3390/rs12223828

Plesser, H. E. (2018). Reproducibility vs. replicability: a brief history of a confused terminology. *Frontiers in Neuroinformatics*, *11*, 76. doi: 10.3389/fninf.2017.00076

Ploton, P., Mortier, F., Réjou-Méchain, M., Barbier, N., Picard, N., Rossi, V., … Pélissier, R. (2020, September). Spatial validation reveals poor predictive performance of large-scale ecological mapping models. *Nature Communications*, *11*(1), 4540. doi: 10.1038/s41467-020-18321-y

Pohjankukka, J., Pahikkala, T., Nevalainen, P., & Heikkonen, J. (2017, July). Estimating the prediction performance of spatial models via spatial k-fold cross validation. *International Journal of Geographical Information Science*, *31*(10), 2001–2019. doi: 10.1080/13658816.2017.1346255

Polikar, R. (2012). Ensemble learning. In C. Zhang & Y. Ma (Eds.), *Ensemble Machine Learning: Methods and Applications* (pp. 1–34). Boston, MA: Springer US. doi: 10.1007/978-1-4419-9326-7_1

Pollastrini, M., Feducci, M., Bonal, D., Fotelli, M., Gessler, A., Grossiord, C., … Bussotti, F. (2016, February). Physiological significance of forest tree defoliation: Results from a survey in a mixed forest in Tuscany (central Italy). *Forest Ecology and Management*, *361*, 170–178. doi: 10.1016/j.foreco.2015.11.018

Poona, N. K., & Ismail, R. (2019, August). Developing optimized spectral indices using machine learning to model Fusarium circinatum stress in Pinus radiata seedlings. *Journal of Applied Remote Sensing*, *13*(3), 034515. doi: 10.1117/1.JRS.13.034515

Probst, P., Boulesteix, A.-L., & Bischl, B. (2019). Tunability: importance of hyperparameters of machine learning algorithms. *Journal of Machine Learning Research*, *20*(53), 1–32. Retrieved 2019-06-11, from http://jmlr.org/papers/v20/18-444.html

Probst, P., Wright, M. N., & Boulesteix, A.-L. (2019). Hyperparameters and tuning strategies for random forest. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *9*(3), e1301. doi: 10.1002/widm.1301

Qi, J., Chehbouni, A., Huete, A. R., Kerr, Y. H., & Sorooshian, S. (1994, May). A modified soil adjusted vegetation index. *Remote Sensing of Environment*, *48*(2), 119–126. doi: 10.1016/0034-4257(94)90134-1

Quillfeldt, P., Engler, J. O., Silk, J. R., & Phillips, R. A. (2017, April). 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: 10.1111/jav.01238

Quinlan, J. R. (1986, March). Induction of decision trees. *Machine Learning*, *1*(1), 81–106. doi: 10/ctd6mv

R Core Team. (2019). *R: a language and environment for statistical computing*. Vienna, Austria. Retrieved from https://www.R-project.org/

Racine, J. (2000, November). Consistent cross-validatory model-selection for dependent data: HV-block cross-validation. *Journal of Econometrics*, *99*(1), 39–61. doi: 10.1016/s0304-4076(00)00030-0

Radovic, M., Ghalwash, M., Filipovic, N., & Obradovic, Z. (2017, January). Minimum redundancy maximum relevance feature selection approach for temporal gene expression data. *BMC Bioinformatics*, *18*(1), 9. doi: 10.1186/s12859-016-1423-9

Raj, D. M. D., & Mohanasundaram, R. (2020, April). An efficient filter-based feature selection model to identify significant features from high-dimensional microarray data. *Arabian Journal for Science and Engineering*,

45(4), 2619–2630. doi: 10.1007/s13369-020-04380-2

Reitz, O., Graf, A., Schmidt, M., Ketzler, G., & Leuchner, M. (2021). Upscaling net ecosystem exchange over heterogeneous landscapes with machine learning. *Journal of Geophysical Research: Biogeosciences*, *126*(2), e2020JG005814. doi: 10.1029/2020JG005814

Rengarajan, R., & Schott, J. R. (2016, September). Modeling forest defoliation using simulated BRDF and assessing its effect on reflectance and sensor reaching radiance. In *Remote Sensing and Modeling of Ecosystems for Sustainability XIII* (Vol. 9975, p. 997503). San Diego, California, United States: International Society for Optics and Photonics. doi: 10.1117/12.2235391

Rest, K. L., Pinaud, D., Monestiez, P., Chadoeuf, J., & Bretagnolle, V. (2014). Spatial leave-one-out cross-validation for variable selection in the presence of spatial autocorrelation. *Global Ecology and Biogeography*, *23*(7), 811–820. doi: 10.1111/geb.12161

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016, June). Model-Agnostic Interpretability of Machine Learning. *arXiv:1606.05386 [cs, stat]*. Retrieved 2022-04-25, from http://arxiv.org/abs/1606.05386

Richter, J. (2017). *mlrHyperopt: Easy hyperparameter optimization with mlr and mlrMBO*. Retrieved from http://doi.org/10.5281/zenodo.896269

Ridgeway, G. (2017). *gbm: Generalized boosted regression models*. Retrieved from https://CRAN.R-project.org/package=gbm

Roberts, D. R., Bahn, V., Ciuti, S., Boyce, M. S., Elith, J., Guillera-Arroita, G., … Dormann, C. F. (2017, March). Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure. *Ecography*, *40*(8), 913–929. doi: 10.1111/ecog.02881

Rochac, J. F. R., & Zhang, N. (2016, February). Feature extraction in hyperspectral imaging using adaptive feature selection approach. In *2016 Eighth International Conference on Advanced Computational Intelligence (ICACI)* (pp. 36–40). doi: 10.1109/ICACI.2016.7449799

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: 10.1109/access.2017.2779794

Rondeaux, G., Steven, M., & Baret, F. (1996, February). Optimization of soil-adjusted vegetation indices. *Remote Sensing of Environment*, *55*(2), 95–107. doi: 10.1016/0034-4257(95)00186-7

Rougier, N. P., Hinsen, K., Alexandre, F., Arildsen, T., Barba, L., Benureau, F. C. Y., … Zito, T. (2017, December). Sustainable computational science: the ReScience initiative. *PeerJ Computer Science*, *3*, e142. doi: 10.7717/peerj-cs.142

Roujean, J.-L., & Breon, F.-M. (1995, March). Estimating PAR absorbed by vegetation from bidirectional reflectance measurements. *Remote Sensing of Environment*, *51*(3), 375–384. doi: 10.1016/0034-4257(94)00114-3

Royle, D. D., & Lathrop, R. G. (1997, August). Monitoring Hemlock Forest Health in New Jersey Using Landsat TM Data and Change Detection Techniques. *Forest Science*, *43*(3), 327–335. doi: 10.1093/forestscience/43.3.327

Rullán-Silva, C., Olthoff, A. E., Pando, V., Pajares, J. A., & Delgado, J. A. (2015, July). Remote monitoring of defoliation by the beech leaf-mining weevil Rhynchaenus fagi in northern Spain. *Forest Ecology and Management*, *347*, 200–208. doi: 10.1016/j.foreco.2015.03.005

Ruß, G., & Brenning, A. (2010a). Data mining in precision agriculture: management of spatial information. In E. Hüllermeier, R. Kruse, & F. Hoffmann (Eds.), *Computational Intelligence for Knowledge-Based Systems Design*

(pp. 350–359). Berlin, Heidelberg: Springer. doi: 10.1007/978-3-642-14049-5_36

Ruß, G., & Brenning, A. (2010b, May). Spatial variable importance assessment for yield prediction in precision agriculture. In *Advances in Intelligent Data Analysis IX* (pp. 184–195). Springer, Berlin, Heidelberg. 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. doi: 10.1007/978-3-642-14400-4_35

Sandino, J., Pegg, G., Gonzalez, F., & Smith, G. (2018, April). Aerial mapping of forests affected by pathogens using UAVs, hyperspectral sensors, and artificial intelligence. *Sensors*, *18*(4), 944. doi: 10.3390/s18040944

Schliep, K., & Hechenbichler, K. (2016). *kknn: Weighted k-nearest neighbors*. Retrieved from `https://CRAN.R-project.org/package=kknn`

Schratz, P. (2016). *Modeling the spatial distribution of hail damage in pine plantations of northern Spain as a major risk factor for forest disease.* doi: 10.5281/zenodo.814262

Schratz, P., Becker, M., Lang, M., & Brenning, A. (2021, October). mlr3spatiotempcv: Spatiotemporal resampling methods for machine learning in R. *arXiv:2110.12674 [cs, stat]*. Retrieved 2021-12-20, from `http://arxiv.org/abs/2110.12674`

Schratz, P., Muenchow, J., Iturritxa, E., Cortés, J., Bischl, B., & Brenning, A. (2021, January). Monitoring forest health using hyperspectral imagery: does feature selection improve the performance of machine-learning techniques? *Remote Sensing*, *13*(23), 4832. doi: 10.3390/rs13234832

Schratz, P., Muenchow, J., Iturritxa, E., Richter, J., & Brenning, A. (2019, August). Hyperparameter tuning and performance assessment of statistical and machine-learning algorithms using spatial data. *Ecological Modelling*, *406*, 109–120. doi: 10.1016/j.ecolmodel.2019.06.002

Serrano, L., Peñuelas, J., & Ustin, S. L. (2002, August). Remote sensing of nitrogen and lignin in Mediterranean vegetation from AVIRIS data. *Remote Sensing of Environment*, *81*(2-3), 355–364. doi: 10.1016/s0034-4257(02)00011-1

Shannon, C. E. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, *27*(3), 379–423. doi: 10.1002/j.1538-7305.1948.tb01338.x

Shao, J. (1993, June). Linear model selection by cross-validation. *Journal of the American Statistical Association*, *88*(422), 486. doi: 10.2307/2290328

Shendryk, I., Broich, M., Tulbure, M. G., McGrath, A., Keith, D., & Alexandrov, S. V. (2016, December). Mapping individual tree health using full-waveform airborne laser scans and imaging spectroscopy: A case study for a floodplain eucalypt forest. *Remote Sensing of Environment*, *187*, 202–217. doi: 10.1016/j.rse.2016.10.014

Showstack, R. (2014). Sentinel satellites initiate new era in earth observation. *Eos, Transactions American Geophysical Union*, *95*(26), 239–240. doi: 10.1002/2014EO260003

Sievert, C. (2020). *Interactive web-based data visualization with R, plotly, and shiny*. Chapman and Hall/CRC. Retrieved from `https://plotly-r.com`

Šimić Milas, A., Rupasinghe, P., Balenović, I., & Grosevski, P. (2015, December). Assessment of Forest Damage in Croatia using Landsat-8 OLI Images. *South-east European forestry : SEEFOR*, *6*(2), 159–169. doi: 10.15177/seefor.15-14

Sims, D. A., & Gamon, J. A. (2002, August). Relationships between leaf pigment content and spectral reflectance across a wide range of species, leaf structures and developmental

stages. *Remote Sensing of Environment*, *81*(2), 337–354. doi: 10.1016/s0034-4257(02)00010-x

Sinclair, W. A., & Lyon, H. H. (2005). *Diseases of trees and shrubs*. (No. Ed. 2). Comstock Publishing Associates. Retrieved from https://books.google.ch/books?id=-eucAQAACAAJ&dq=Diseases+of+trees+and+shrubs&hl=de&sa=X&redir_esc=y

Singleton, A. D., Spielman, S., & Brunsdon, C. (2016, August). Establishing a framework for Open Geographic Information science. *International Journal of Geographical Information Science*, *30*(8), 1507–1521. doi: 10.1080/13658816.2015.1137579

Smith, R. C. G., Adams, J., Stephens, D. J., & Hick, P. T. (1995). Forecasting wheat yield in a Mediterranean-type environment from the NOAA satellite. *Australian Journal of Agricultural Research*, *46*(1), 113. doi: 10.1071/ar9950113

Smoliński, S., & Radtke, K. (2016, August). 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*, fsw136. doi: 10.1093/icesjms/fsw136

Sochat, V. (2018). The experiment factory: Reproducible experiment containers. *Journal of Open Source Software*, *3*(22), 521. Retrieved from https://joss.theoj.org/papers/10.21105/joss.00521.pdf

Sprent, P., & Smeeton, N. C. (2016). *Applied Nonparametric Statistical Methods*. CRC Press.

Srivastava, V., Griess, V. C., & Padalia, H. (2018, October). Mapping invasion potential using ensemble modelling. A case study on Yushania maling in the Darjeeling Himalayas. *Ecological Modelling*, *385*, 35–44. doi: 10.1016/j.ecolmodel.2018.07.001

Stanosz, G. R., Blodgett, J. T., Smith, D. R., & Kruger, E. L. (2001). Water stress and Sphaeropsissapinea as a latent pathogen of red pine seedlings. *New Phytologist*, *149*(3), 531–538. doi: 10.1046/j.1469-8137.2001.00052.x

Steger, S., Brenning, A., Bell, R., Petschko, H., & Glade, T. (2016, June). Exploring discrepancies between quantitative validation results and the geomorphic plausibility of statistical landslide susceptibility maps. *Geomorphology*, *262*, 8–23. doi: 10.1016/j.geomorph.2016.03.015

Stelmaszczuk-Górska, M., Thiel, C., & Schmullius, C. (2017, March). 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

Stewart, S. B., Elith, J., Fedrigo, M., Kasel, S., Roxburgh, S. H., Bennett, L. T., … Nitschke, C. R. (2021). Climate extreme variables generated using monthly time-series data improve predicted distributions of plant species. *Ecography*, *44*(4), 626–639. doi: 10.1111/ecog.05253

Stone, M. (1974). Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society: Series B (Methodological)*, *36*(2), 111–133. doi: 10.1111/j.2517-6161.1974.tb00994.x

Storer, A., Gordon, T., & Clark, S. (1998). Association of the pitch canker fungus, Fusarium subglutinans f. sp. pini, with Monterey pine seeds and seedlings in California. *Plant pathology*, *47*(5), 649–656. doi: 10.1046/j.1365-3059.1998.00288.x

Swart, W. J., Wingfield, M. J., & Knox-Davies, P. S. (1988). Relative susceptibilities to Sphaeropsis sapinea of six Pinus spp. cultivated in South Africa. *European Journal of Forest Pathology*, *18*(3-4), 184–189. doi: 10.1111/j.1439-0329.1988.tb00917.x

Telford, R., & Birks, H. (2005, November). The secret assumption of transfer functions: problems with spatial autocorrelation in evalu-

ating model performance. *Quaternary Science Reviews*, *24*(20-21), 2173–2179. doi: 10.1016/j.quascirev.2005.05.001

Telford, R., & Birks, H. (2009, June). Evaluation of transfer functions in spatially structured environments. *Quaternary Science Reviews*, *28*(13-14), 1309–1316. doi: 10.1016/j.quascirev.2008.12.020

Thenkabail, P. S., Lyon, J. G., & Huete, A. (Eds.). (2018). *Hyperspectral indices and image classifications for agriculture and vegetation*. CRC Press. doi: 10.1201/9781315159331

Thenkabail, P. S., Smith, R. B., & De Pauw, E. (2000, February). Hyperspectral vegetation indices and their relationships with agricultural crop characteristics. *Remote Sensing of Environment*, *71*(2), 158–182. doi: 10.1016/S0034-4257(99)00067-X

Thompson, S. K. (2012). Sampling. In *Sampling* (Third ed., p. i-xxi). John Wiley & Sons, Ltd. doi: 10.1002/9781118162934.fmatter

Tognelli, M. F., & Kelt, D. A. (2004). Analysis of determinants of mammalian species richness in South America using spatial autoregressive models. *Ecography*, *27*(4), 427–436. doi: 10.1111/j.0906-7590.2004.03732.x

Townsend, P. A., Singh, A., Foster, J. R., Rehberg, N. J., Kingdon, C. C., Eshleman, K. N., & Seagle, S. W. (2012, April). A general Landsat model to predict canopy defoliation in broadleaf deciduous forests. *Remote Sensing of Environment*, *119*, 255–265. doi: 10.1016/j.rse.2011.12.023

Treitz, P. M., & Howarth, P. J. (1999, September). Hyperspectral remote sensing for estimating biophysical parameters of forest ecosystems. *Progress in Physical Geography: Earth and Environment*, *23*(3), 359–390. doi: 10.1177/030913339902300303

Trunk, G. V. (1979, July). A problem of dimensionality: a simple example. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *PAMI-1*(3), 306–307. doi: 10.1109/TPAMI.1979.4766926

Tsai, C.-F., & Sung, Y.-T. (2020, September). Ensemble feature selection in high dimension, low sample size datasets: Parallel and serial combination approaches. *Knowledge-Based Systems*, *203*, 106097. doi: 10.1016/j.knosys.2020.106097

Tsai, Y.-T. (2018, December). An overview of machine learning and HPC in open sources for bioinformatics. In *2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)* (pp. 1338–1342). doi: 10.1109/BIBM.2018.8621078

Tucker, C. J. (1979, May). Red and photographic infrared linear combinations for monitoring vegetation. *Remote Sensing of Environment*, *8*(2), 127–150. doi: 10.1016/0034-4257(79)90013-0

Urban, M., Berger, C., Mudau, T. E., Heckel, K., Truckenbrodt, J., Onyango Odipo, V., … Schmullius, C. (2018, September). Surface moisture and vegetation cover analysis for drought monitoring in the southern Kruger national park using Sentinel-1, Sentinel-2, and Landsat-8. *Remote Sensing*, *10*(9), 1482. doi: 10.3390/rs10091482

Urbanowicz, R. J., Olson, R. S., Schmitt, P., Meeker, M., & Moore, J. H. (2018). Benchmarking relief-based feature selection methods for bioinformatics data mining. *Journal of Biomedical Informatics*, *85*, 168–188. doi: 10.1016/j.jbi.2018.07.015

Ushey, K. (2021). *renv: Project environments* [manual]. Retrieved from `https://CRAN.R-project.org/package=renv`

Valavi, R., Elith, J., Lahoz-Monfort, J. J., & Guillera-Arroita, G. (2019). blockCV: An r package for generating spatially or environmentally separated folds for k-fold cross-validation of species distribution models. *Methods in Ecology and Evolution*, *10*(2), 225–232. doi: 10.1111/2041-210X.13107

Vanschoren, J., van Rijn, J. N., Bischl, B., & Torgo,

L. (2014, June). OpenML: networked science in machine learning. *ACM SIGKDD Explorations Newsletter*, *15*(2), 49–60. doi: 10.1145/2641190.2641198

Vanwinckelen, G., & Blockeel, H. (2012, January). On estimating model accuracy with repeated cross-validation. In *BeneLearn 2012: Proceedings of the 21st Belgian-Dutch Conference on Machine Learning* (pp. 39–44). Retrieved 2021-05-22, from https://lirias.kuleuven.be/1655861

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

Verbesselt, J., Robinson, A., Stone, C., & Culvenor, D. (2009, September). Forecasting tree mortality using change metrics derived from MODIS satellite data. *Forest Ecology and Management*, *258*(7), 1166–1173. doi: 10.1016/j.foreco.2009.06.011

Vincini, M., Frazzi, E., & D'Alessio, P. (2006). Angular dependence of maize and sugar beet VIs from directional CHRIS/Proba data. In *Proc. 4th ESA CHRIS PROBA workshop* (Vol. 2006, pp. 19–21).

Vogelmann, J. E., Rock, B. N., & Moss, D. M. (1993, May). Red edge spectral measurements from sugar maple leaves. *International Journal of Remote Sensing*, *14*(8), 1563–1575. doi: 10.1080/01431169308953986

Vorpahl, P., Elsenbeer, H., Märker, M., & Schröder, B. (2012, July). How can statistical models help to determine driving factors of landslides? *Ecological Modelling*, *239*, 27–39. 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. doi: 10.1016/j.renene.2016.12.095

Wadoux, A. M. J. C., Heuvelink, G. B. M., de Bruin, S., & Brus, D. J. (2021, October). Spatial cross-validation is not the right way to evaluate map accuracy. *Ecological Modelling*, *457*, 109692. doi: 10.1016/j.ecolmodel.2021.109692

Walthall, C. L., Daughtry, C. S. T., Chappelle, E. W., Mcmurtrey, J. E., & Kim, M. S. (1994). The use of high spectral resolution bands for estimating absorbed photosynthetically active radiation (a par). *NASA Gov*. Retrieved 2020-11-30, from https://core.ac.uk/display/10462460

Wang, M.-R., Cui, Z.-H., Li, J.-W., Hao, X.-Y., Zhao, L., & Wang, Q.-C. (2018, October). In vitro thermotherapy-based methods for plant virus eradication. *Plant Methods*, *14*(1), 87. doi: 10.1186/s13007-018-0355-y

Watanabe, M. D. B., & Ortega, E. (2014, January). 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: 10.1016/j.ecolmodel.2013.03.006

Watt, M. S., Ganley, R. J., Kriticos, D. J., & Manning, L. K. (2011, February). Dothistroma needle blight and pitch canker: the current and future potential distribution of two important diseases of Pinus species. *Canadian Journal of Forest Research*, *41*(2), 412–424. doi: 10.1139/X10-204

Watt, M. S., Kriticos, D. J., Alcaraz, S., Brown, A. V., & Leriche, A. (2009, March). The hosts and potential geographic range of Dothistroma needle blight. *Forest Ecology and Management*, *257*(6), 1505–1519. doi: 10.1016/j.foreco.2008.12.026

Wenger, S. J., & Olden, J. D. (2012, January). Assessing transferability of ecological models: an underappreciated aspect of statistical validation. *Methods in Ecology and Evolution*, *3*(2), 260–267. doi: 10.1111/j.2041-210X.2011.00170.x

Whitehill, J. G. A., Lehman, J. S., & Bonello, P. (2007, February). Ips pini (Curculionidae: Scolytinae) Is a vector of the

fungal pathogen, Sphaeropsis sapinea (Coelomycetes), to Austrian Pines, Pinus nigra (Pinaceae). *Environmental Entomology*, *36*(1), 114–120. doi: 10.1603/0046-225X(2007)36[114:IPCSIA]2.0.CO;2

Wickham, H. (2009). *ggplot2: elegant graphics for data analysis*. Springer-Verlag New York. Retrieved from http://ggplot2.org

Wieland, R., Kerkow, A., Früh, L., Kampen, H., & Walther, D. (2017, May). Automated feature selection for a machine learning approach toward modeling a mosquito distribution. *Ecological Modelling*, *352*, 108–112. doi: 10.1016/j.ecolmodel.2017.02.029

Wilkinson, M. D., Dumontier, M., Aalbersberg, I. J., Appleton, G., Axton, M., Baak, A., … Mons, B. (2016, March). The FAIR Guiding Principles for scientific data management and stewardship. *Scientific Data*, *3*(1), 160018. doi: 10.1038/sdata.2016.18

Willmott, C. J., & Matsuura, K. (2006, January). On the use of dimensioned measures of error to evaluate the performance of spatial interpolators. *International Journal of Geographical Information Science*, *20*(1), 89–102. doi: 10.1080/13658810500286976

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*(4), 319. doi: 10.1071/ap08036

Wollan, A. K., Bakkestuen, V., Kauserud, H., Gulden, G., & Halvorsen, R. (2008, December). Modelling and predicting fungal distribution patterns using herbarium data. *Journal of Biogeography*, *35*(12), 2298–2310. doi: 10.1111/j.1365-2699.2008.01965.x

Wood, S. (2017). *Generalized additive models: an introduction with R*. Chapman and Hall/CRC. Retrieved from https://cran.r-project.org/package=mgcv

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), 1–17. doi: 10.18637/jss.v077.i01

Wu, C., Niu, Z., Tang, Q., & Huang, W. (2008, July). Estimating chlorophyll content from hyperspectral vegetation indices: Modeling and validation. *Agricultural and Forest Meteorology*, *148*(8-9), 1230–1241. doi: 10.1016/j.agrformet.2008.03.005

Wu, T., Luo, J., Dong, W., Gao, L., Hu, X., Wu, Z., … Liu, J. (2020). Disaggregating county-level census data for population mapping using residential geo-objects with multisource geo-spatial data. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *13*, 1189–1205. doi: 10.1109/JSTARS.2020.2974896

Wu, W. (2014, January). The generalized difference vegetation index (GDVI) for dryland characterization. *Remote Sensing*, *6*(2), 1211–1233. doi: 10.3390/rs6021211

Xia, J., Liao, W., Chanussot, J., Du, P., Song, G., & Philips, W. (2015, July). Improving random forest with ensemble of features and semisupervised feature extraction. *IEEE Geoscience and Remote Sensing Letters*, *12*(7), 1471–1475. doi: 10.1109/LGRS.2015.2409112

Xu, H., Caramanis, C., & Mannor, S. (2016, July). Statistical optimization in high dimensions. *Operations Research*, *64*(4), 958–979. doi: 10.1287/opre.2016.1504

Xu, S., Zhao, Q., Yin, K., Zhang, F., Liu, D., & Yang, G. (2019, February). Combining random forest and support vector machines for object-based rural-land-cover classification using high spatial resolution imagery. *Journal of Applied Remote Sensing*, *13*(1), 014521. doi: 10.1117/1.JRS.13.014521

Yang, E.-S., Kim, J. D., Park, C.-Y., Song, H.-J., & Kim, Y.-S. (2017, August). Hyperparameter tuning for hidden unit conditional ran-

dom fields. *Engineering Computations*, *34*(6), 2054–2062. doi: 10.1108/ec-11-2015-0350

Youssef, A. M., Pourghasemi, H. R., Pourtaghi, Z. S., & Al-Katheeri, M. M. (2015, December). 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*(5), 1315–1318. doi: 10.1007/s10346-015-0667-1

Yu, R., Ren, L., & Luo, Y. (2021, July). Early detection of pine wilt disease in Pinus tabuliformis in north China using a field portable spectrometer and UAV-based hyperspectral imagery. *Forest Ecosystems*, *8*(1), 44. doi: 10.1186/s40663-021-00328-6

Zandler, H., Brenning, A., & Samimi, C. (2015, March). Quantifying dwarf shrub biomass in an arid environment: comparing empirical methods in a high dimensional setting. *Remote Sensing of Environment*, *158*, 140–155. doi: 10.1016/j.rse.2014.11.007

Zarco-Tejada, P. J., González-Dugo, V., Williams, L. E., Suárez, L., Berni, J. A. J., Goldhamer, D., & Fereres, E. (2013, November). A PRI-based water stress index combining structural and chlorophyll effects: Assessment using diurnal narrow-band airborne imagery and the CWSI thermal index. *Remote Sensing of Environment*, *138*, 38–50. doi: 10.1016/j.rse.2013.07.024

Zarco-Tejada, P. J., & Miller, J. R. (1999, November). Land cover mapping at BOREAS using red edge spectral parameters from CASI imagery. *Journal of Geophysical Research: Atmospheres*, *104*(D22), 27921–27933. doi: 10.1029/1999jd900161

Zarco-Tejada, P. J., Pushnik, J. C., Dobrowski, S., & Ustin, S. L. (2003, February). Steady-state chlorophyll a fluorescence detection from canopy derivative reflectance and double-peak red-edge effects. *Remote Sensing of Environment*, *84*(2), 283–294. doi: 10.1016/s0034-4257(02)00113-x

Zawadzki, Z., & Kosinski, M. (2019). *FSelectorRcpp: 'Rcpp' Implementation of 'FSelector' Entropy-Based Feature Selection Algorithms with a Sparse Matrix Support*. Retrieved from `https://CRAN.R-project.org/package=FSelectorRcpp`

Zhang, K., Thapa, B., Ross, M., & Gann, D. (2016). Remote sensing of seasonal changes and disturbances in mangrove forest: a case study from South Florida. *Ecosphere*, e01366. doi: 10.1002/ecs2.1366

Zhao, X.-M. (2013). Maximum Relevance/Minimum Redundancy (MRMR). In W. Dubitzky, O. Wolkenhauer, K.-H. Cho, & H. Yokota (Eds.), *Encyclopedia of Systems Biology* (pp. 1191–1192). New York, NY: Springer New York. doi: 10.1007/978-1-4419-9863-7_432

Zhao, Y., & Karypis, G. (2002). Evaluation of hierarchical clustering algorithms for document datasets. In *Proceedings of the eleventh international conference on Information and knowledge management* (pp. 515–524). doi: 10.1145/584792.584877

Zhu, Y., Liu, K., Liu, L., Myint, S. W., Wang, S., Liu, H., & He, Z. (2017, October). Exploring the potential of Worldview-2 red-edge band-based vegetation indices for estimation of mangrove leaf area index with machine learning algorithms. *Remote Sensing*, *9*(10), 1060. doi: 10.3390/rs9101060

Zuber, V., & Strimmer, K. (2011). High-dimensional regression and variable selection using car scores. *Statistical Applications in Genetics and Molecular Biology*, *10*(1), 1–27. doi: 10/c62xhp

Zurell, D., Zimmermann, N. E., Gross, H., Baltensweiler, A., Sattler, T., & Wüest, R. O. (2020). Testing species assemblage predictions from stacked and joint species distribution models. *Journal of Biogeography*, *47*(1), 101–113. doi: 10.1111/jbi.13608

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. doi: 10.1007/978-0-387-87458-6

Appendices

Appendix 1-1: Descriptive summary of numerical and nominal predictor variables

Variable	n	Min	q_1	$\tilde{x}$	$\bar{x}$	q_3	Max	IQR	#NA
temp	922	12.6	14.6	15.2	15.1	15.7	16.8	1.0	0
precip	922	88.1	181.2	224.6	234.1	252.2	496.6	71.0	0
hail_probability	922	0.0	0.2	0.6	0.5	0.7	1.0	0.5	0
ph	922	4.0	4.4	4.6	4.6	4.8	6.0	0.4	0
slope_degrees	922	0.1	12.3	19.3	19.8	27.0	55.1	14.7	0
pisr	922	-0.1	0.0	0.0	0.0	0.0	0.1	0.1	0
age	922	2.0	13.0	20.0	19.0	24.0	40.0	11.0	0

Table A1: Article 1: Summary of numerical predictor variables. Precipitation (precip) in mm/m², temperature (temp) in °C, solar radiation (pisr) in kW/m², tree age (age) in years. Statistics show sample size (**n**), minimum (**Min**), 25th percentile ($\mathbf{q_1}$), median ($\tilde{\mathbf{x}}$), mean ($\bar{\mathbf{x}}$), 75th percentile ($\mathbf{q_3}$), maximum (**Max**), inner-quartile range (**IQR**) and NA Count (**#NA**).

Variable	Levels	#	%
diploo1	0	700	75.9
	1	222	24.1
	all	922	100.0
soil	soils with clay-enriched subsoil	215	23.3
	soils with little or no profile differentiation	705	76.5
	pronounced accumulation of organic matter in the mineral topsoil	1	0.1
	soils with limitations to root growth	1	0.1
	all	922	100.0
lithology	surface deposits	31	3.4
	clastic sedimentary rock	600	65.1
	biological sedimentary rock	136	14.8
	chemical sedimentary rock	142	15.4
	magmatic rock	13	1.4
	all	922	100.0
year	2009	399	43.3
	2010	260	28.2
	2011	102	11.1
	2012	161	17.5
	all	922	100.0

Table A2: Article 1: Summary of nominal predictor variables

Appendix 1-2: Hyperparameter ranges and types for each model

Table A3: Article 1: Hyperparameter ranges and types for each model. Hyperparameter notations from the respective R packages are shown.

Model (package)	Hyperparameter	Type	Start	End	Default
RF (ranger)	x_{try}	dbl	0	0.5	-
	`min.node.size`	int	1	10	1
	`sample.fraction`	dbl	0.2	0.9	1
SVM (kernlab)	`C`	dbl	2^{-10}	2^{10}	1
	σ	dbl	2^{-5}	2^{5}	1
XGBoost (xgboost)	`nrounds`	int	10	70	-
	`colsample_bytree`	dbl	0.6	1	1
	`subsample`	dbl	0.6	1	1
	`max_depth`	int	3	15	6
	`gamma`	int	0.05	10	0
	`eta`	dbl	0.1	1	0.3
	`min_child_weight`	int	1	7	1

Appendix 2-1: Spearman correlations of NRI feature rankings obtained with different filters

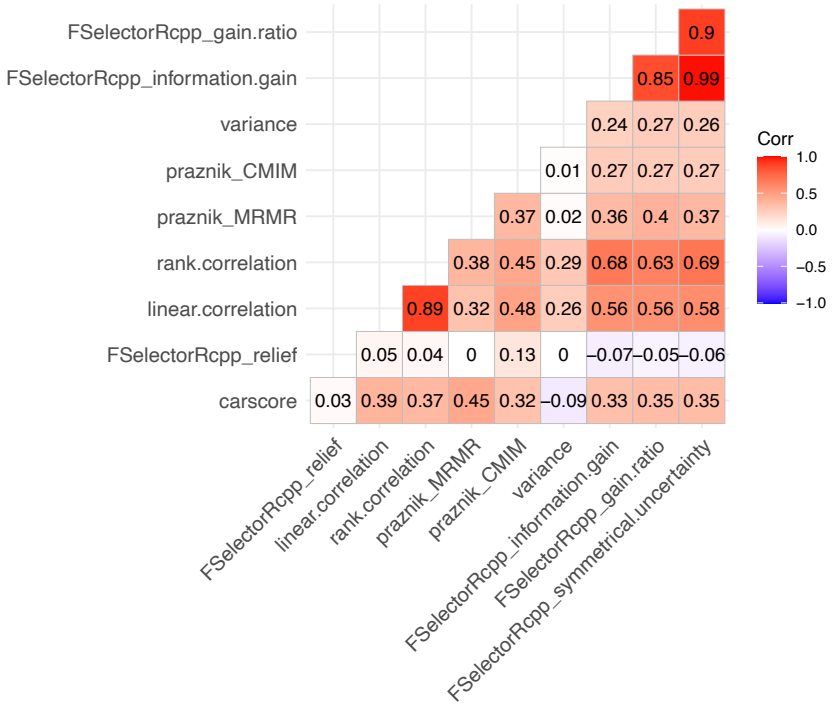


Figure A1: Article 2: Spearman correlations of NRI feature rankings obtained with different filters. Filter names refer to the nomenclature used by the `mlr` R package. Underscores in names divide the terminology into their upstream `R` package and the actual filter name.

Appendix 2-2: List of available vegetation indices in the `hsdar` package

Table A4: Article 2: List of available vegetation indices in the `hsdar` package.

Name	Formula	Reference*
Boochs	D_{703}	Boochs et al. (1990)
Boochs2	D_{720}	Boochs et al. (1990)
CAI	$0.5 \times (R_{2000} + R_{2200}) - R_{2100}$	Nagler et al. (2003)
CARI	$a = (R_{700} - R_{550})/150$ $b = R_{550} - (a \times 550)$ $\frac{R_{700} \times \lvert(a \times 670 + R_{670} + b)}{R_{670} \times (a^2+1)\rvert^{0.5}}$	Walthall et al. (1994)
Carter	R_{695}/R_{420}	Carter (1994)
Carter2	R_{695}/R_{760}	Carter (1994)
Carter3	R_{605}/R_{760}	Carter (1994)
Carter4	R_{710}/R_{760}	Carter (1994)
Carter5	R_{695}/R_{670}	Carter (1994)
Carter6	R_{550}	Carter (1994)
CI	$R_{675} \times R_{690}/R_{683}^2$	Zarco-Tejada et al. (2003)
CI 2	$R_{760}/R_{700} - 1$	A. A. Gitelson et al. (2003)
ClAInt	$\int_{600nm}^{735nm} R$	Oppelt and Mauser (2004)
CRI1	$1/R_{515} - 1/R_{550}$	A. A. Gitelson et al. (2003)
CRI2	$1/R_{515} - 1/R_{770}$	A. A. Gitelson et al. (2003)
CRI3	$1/R_{515} - 1/R_{550} \times R_{770}$	A. A. Gitelson et al. (2003)
CRI4	$1/R_{515} - 1/R_{700} \times R_{770}$	A. A. Gitelson et al. (2003)
D1	D_{730}/D_{706}	Zarco-Tejada et al. (2003)
D2	D_{705}/D_{722}	Zarco-Tejada et al. (2003)

Datt	$(R_{850} - R_{710})/(R_{850} - R_{680})$	Datt (1999b)
Datt2	R_{850}/R_{710}	Datt (1999b)
Datt3	D_{754}/D_{704}	Datt (1999b)
Datt4	$R_{672}/(R_{550} \times R_{708})$	Datt (1998)
Datt5	R_{672}/R_{550}	Datt (1998)
Datt6	$(R_{860})/(R_{550} \times R_{708})$	Datt (1998)
Datt7	$(R_{860} - R_{2218})/(R_{860} - R_{1928})$	Datt (1999a)
Datt8	$(R_{860} - R_{1788})/(R_{860} - R_{1928})$	Datt (1999a)
DD	$(R_{749} - R_{720}) - (R_{701} - R_{672})$	le Maire et al. (2004)
DDn	$2 \times (R_{710} - R_{660} - R_{760})$	Lemaire et al. (2008)
DPI	$(D_{688} * D_{710})/D_{697}^2$	Zarco-Tejada et al. (2003)
DWSI1	R_{80}/R_{1660}	Apan et al. (2004)
DWSI2	R_{1660}/R_{550}	Apan et al. (2004)
DWSI3	R_{1660}/R_{680}	Apan et al. (2004)
DWSI4	R_{550}/R_{680}	Apan et al. (2004)
DWSI5	$(R_{800} + R_{550})/(R_{1660} + R_{680})$	Apan et al. (2004)
EGFN	$\frac{(\max(D_{650:750}) - \max(D_{500:550}))}{(\max(D_{650:750}) + \max(D_{500:550}))}$	Peñuelas et al. (1994)
EGFR	$\max(D_{650:750})/\max(D_{500:550})$	Peñuelas et al. (1994)
EVI	$\frac{2.5 \times ((R_{800} - R_{670})}{(R_{800} - (6 \times R_{670}) - (7.5 \times R_{475}) + 1)}$	A. R. Huete et al. (1997)
GDVI	$(R^n_{800} - R^n_{680})/(R^n_{800} + R^n_{680})^{**}$	W. Wu (2014)
GI	R_{554}/R_{677}	Smith et al. (1995)
Gitelson	$1/R_{700}$	A. A. Gitelson et al. (1999)
Gitelson2	$(R_{750} - R_{800}/R_{695} - R_{740}) - 1$	A. A. Gitelson et al. (2003)

GMI1	R_{750}/R_{550}	A. A. Gitelson et al. (2003)
GMI2	R_{750}/R_{700}	A. A. Gitelson et al. (2003)
Green NDVI	$\frac{R_{800}-R_{550}}{R_{800}+R_{550}}$	A. A. Gitelson et al. (1996)
LWVI_1	$\frac{(R_{1094}-R_{983})}{(R_{1094}+R_{983})}$	Galvão et al. (2005)
LWVI_2	$\frac{R_{1094}-R_{1205}}{R_{1094}+R_{1205}}$	Galvão et al. (2005)
Maccioni	$\frac{R_{780}-R_{710})}{R_{780}-R_{680}}$	Maccioni et al. (2001)
MCARI	$((R_{700}-R_{670})-0.2\times(R_{700}-R_{550}))\times(R_{700}/R_{670})$	Daughtry (2000)
MCARI2	$((R_{750}-R_{705})-0.2\times(R_{750}-R_{550}))\times(R_{750}/R_{705})$	C. Wu et al. (2008)
mND705	$\frac{(R_{750}-R_{705})}{R_{750}+R_{705}-2\times R_{445}}$	Sims and Gamon (2002)
mNDVI	$\frac{(R_{800}-R_{680})}{R_{800}+R_{680}-2\times R_{445}}$	Sims and Gamon (2002)
MPRI	$\frac{R_{515}-R_{530}}{R_{515}+R_{530}}$	Hernández-Clemente et al. (2011)
MSAVI	$0.5\times((2\times R_{800}+1)^2-8\times(R_{800}-R_{670}))^{0.5}$	Qi et al. (1994)
MSI	$\frac{R_{1600}}{R_{817}}$	E. Hunt and Rock (1989)
mSR	$\frac{R_{800}-R_{445}}{R_{680}-R_{445}}$	Sims and Gamon (2002)
mSR2	$\frac{(R_{750}/R_{705})-1}{R_{750}/R_{705}+1)^{0.5}}$	J. M. Chen (1996)
mSR705	$\frac{R_{750}-R_{445}}{R_{705}-R_{445}}$	Sims and Gamon (2002)
MTCI	$\frac{R_{754}-R_{709}}{R_{709}-R_{681}}$	J. Dash and Curran (2007)
MTVI	$1.2\times(1.2\times(R_{800}-R_{550})-2.5\times(R_{670}-R_{550}))$	Haboudane et al. (2002)
NDLI	$\frac{log(1/R_{1754})-log(1/R_{1680})}{log(1/R_{1754})+log(1/R_{1680})}$	Serrano et al. (2002)
NDNI	$\frac{log(1/R_{1510})-log(1/R_{1680})}{log(1/R_{1510})+log(1/R_{1680})}$	Serrano et al. (2002)
NDVI	$\frac{R_{800}-R_{680}}{R_{800}+R_{680}}$	Tucker (1979)
NDVI2	$\frac{R_{750}-R_{705}}{R_{750}+R_{705}}$	A. Gitelson and Merzlyak (1994)
NDVI3	$\frac{R_{682}-R_{553}}{R_{682}+R_{553}}$	Guanter et al. (2005)

NDWI	$\frac{R_{860}-R_{1240}}{R_{860}+R_{1240}}$	B.-C. Gao (1996)
NPCI	$\frac{R_{680}-R_{430}}{R_{680}+R_{430}}$	Peñuelas et al. (1994)
OSAVI	$\frac{(1+0.16)\times(R_{800}-R_{670})}{R_{800}+R_{670}+0.16}$	Rondeaux et al. (1996)
OSAVI2	$\frac{(1+0.16)\times(R_{750}-R_{705})}{R_{750}+R_{705}+0.16)}$	C. Wu et al. (2008)
PARS	$\frac{R_{746}}{R_{513}}$	Chappelle et al. (1992)
PRI	$\frac{R_{531}-R_{570}}{R_{531}+R_{570}}$	Gamon et al. (1992)
PRI_norm	$\frac{PRI\times(-1)}{RDVI\times R_{700}/R_{670}}$	Zarco-Tejada et al. (2013)
PRI*CI2	$PRI * CI2$	Garrity et al. (2011)
PSRI	$\frac{R_{678}-R_{500}}{R_{750}}$	Merzlyak et al. (1999)
PSSR	$\frac{R_{800}}{R_{635}}$	Blackburn (1998)
PSND	$\frac{R_{800}-R_{470}}{R_{800}-R_{470})}$	Blackburn (1998)
PWI	$\frac{R_{900}}{R_{970}}$	Penuelas et al. (1997)
RDVI	$\frac{R_{800}-R_{670}}{\sqrt{R_{800}+R_{670}}}$	Roujean and Breon (1995)
REP_LE	Red-edge position through linear extrapolation	Cho and Skidmore (2006)
REP_Li	$R_{re} = \frac{R_{670}+R_{780})}{2}$ $\frac{700+40\times((R_{re}-R_{700})}{(R_{740}-R_{700}))}$	Guyot and Baret (1988)
SAVI	$\frac{(1+L)\times(R_{800}-R_{670})}{(R_{800}+R_{670}+L)}$	A. Huete (1988)
SIPI	$\frac{R_{800}-R_{445}}{R_{800}-R_{680}}$	Penuelas et al. (1995)
SPVI	$0.4 \times 3.7 \times (R_{800} - R_{670}) - 1.2 \times ((R_{530} - R_{670})^2)^{0.5}$	Vincini et al. (2006)
SR	$\frac{R_{800}}{R_{680}}$	Jordan (1969)
SR1	$\frac{R_{750}}{R_{700}}$	A. A. Gitelson and Merzlyak (1997)
SR2	$\frac{R_{752}}{R_{690}}$	A. A. Gitelson and Merzlyak (1997)
SR3	$\frac{R_{750}}{R_{550}}$	A. A. Gitelson and Merzlyak (1997)

SR4	$\frac{R_{700}}{R_{670}}$	McMurtrey et al. (1994)
SR5	$\frac{R_{675}}{R_{700}}$	Chappelle et al. (1992)
SR6	$\frac{R_{750}}{R_{710}}$	Zarco-Tejada and Miller (1999)
SR7	$\frac{R_{440}}{R_{690}}$	Lichtenthaler et al. (1996)
SR8	$\frac{R_{515}}{R_{550}}$	Hernández-Clemente et al. (2012)
SRPI	$\frac{R_{430}}{R_{680}}$	Penuelas et al. (1995)
SRWI	$\frac{R_{850}}{R_{1240}}$	Zarco-Tejada et al. (2003)
Sum_Dr1	$\sum_{i=626}^{795} D1_i$	Elvidge and Chen (1995)
Sum_Dr2	$\sum_{i=680}^{780} D1_i$	Filella and Penuelas (1994)
SWIR FI	$\frac{R^2_{2133}}{R_{2225} \times R^3_{2209}}$	Levin et al. (2007)
SWIR LI	$3.87 \times (R_{2210} - R_{2090}) - 27.51 \times (R_{2280} - R_{2090}) - 0.2$	Lobell et al. (2001)
SWIR SI	$-41.59 \times (R_{2210} - R_{2090}) + 1.24 \times (R_{2280} - R_{2090}) + 0.64$	Lobell et al. (2001)
SWIR VI	$37.72 \times (R_{2210} - R_{2090}) + 6.27 \times (R_{2280} - R_{2090}) + 0.57$	Lobell et al. (2001)
TCARI	$3 * ((R_{700} - R_{670}) - 0.2 \times R_{700} - R_{550}) \times (R_{700}/R_{670}))$	Haboudane et al. (2002)
TCARI/OSAVI	TCARI/OSAVI	Haboudane et al. (2002)
TCARI2	$3 \times ((R_{750} - R_{705}) - 0.2 \times (R_{750} - R_{550}) \times (R_{750}/R_{705}))$	C. Wu et al. (2008)
TCARI2/OSAVI2	TCARI2/OSAVI2	C. Wu et al. (2008)
TGI	$-0.5(190(R_{670} - R_{550}) - 120(R_{670} - R_{480}))$	E. R. Hunt et al. (2013)
TVI	$0.5 \times (120 \times (R_{750} - R_{550}) - 200 \times (R_{670} - R_{550}))$	Broge and Leblanc (2001)
Vogelmann	$\frac{R_{740}}{R_{720}}$	Vogelmann et al. (1993)
Vogelmann2	$\frac{R_{734} - R_{747}}{R_{715} + R_{726})}$	Vogelmann et al. (1993)
Vogelmann3	$\frac{D_{715}}{D_{705}}$	Vogelmann et al. (1993)
Vogelmann4	$\frac{R_{734} - R_{747}}{R_{715} + R_{720}}$	Vogelmann et al. (1993)

Appendix 2-3: Translation of the original hyperspectral data preprocessing description

The following information was provided by the Institut Carogràfic i Geològic de Catalunya, which was in charge of image acquisition and data preprocessing.

The AISA EAGLE-II sensor was used for airborne image acquisition with a field of view of 37.7 °. Its spectral resolution is 2.4 nm and ranges from 400 nm to 1000 nm.

The conversion of digital numbers (DN) to spectral radiance was made using software designed for the instrument. Images were originally scaled in 12 bits but were radiometrically calibrated to 16 bits, reserving the highest value (65,535) for null values. The procedure was applied to the 23 previously selected images. Finally, the geometric and atmospheric corrections were applied to the images.

The aim of this procedure was to reduce the positional errors of the images. The cartographic reference system in use was EPSG 25830. Positioning was achieved by coupling an Applanix POS AV 410 system to the sensor, integrating GPS and IMU systems. The system provides geographic coordinates of the terrain and relative coordinates of the aircraft (attitude) at each scanned line. Additionally a DSM from GeoEuskadi with a spatial resolution of 1 m was used. The orthorectified hyperspectral images were compared to orthoimages (1:5000) from GeoEuskadi. This comparison was used as the base to calculate RMSE, which was below the ground sampling distance in the across and along track directions.

The radiance measured by an instrument depends on the illumination geometry and the reflective properties of the observed surface. Radiation may be absorbed or scattered (Rayleigh and Mie scattering). Scattering is responsible for the adjacency effect, i.e., radiation coming from neighbors' areas to the target pixel. The MODTRAN algorithm was used to model the effect of the atmosphere on the radiation. To represent the aerosols of the study area, the rural model was used. In addition, optical thickness was estimated on pixels with a high vegetation cover. Columnar water vapor was estimated by a linear regression ratio where the spectral radiance of each pixel at the band of the maximum water absorption (906 nm) is compared to its theoretical value in the absence of absorption. Nonetheless, this technique is unreliable in the presence of a spectral resolution as in this case. To resolve this, the water vapor parameter was selected manually according to the smoothness observed on the reflectance peak at 960 nm. This was combined with a mid-latitude summer atmosphere model. The output of this procedure was reflectance from the target pixel scaled between 0 and 10,000.

The image acquisitions were originally attempted during one day (29 October 2016). Due to the variable meteorological conditions, some stands had to be imaged one day later.