Supporting Ecological Decision Making Using Feature-Selection and Variable Importance

Supporting Ecological Decision Making Using Feature-Selection and Variable Importance

Supporting Ecological Decision Making Using Feature-Selection and Variable Importance

Patrick Schratz, *Member, IEEE,*
Jannes Muenchow, *Member, IEEE,* Eugenia
Iturritxa, *Member, IEEE,* José
Cortés, *Member, IEEE,*
Bernd Bischl, *Member, IEEE,* and
Alexander Brenning, *Member, IEEE*

***Abstract*—The abstract goes here.**

***Index Terms*—hyperspectral imagery, forest health modeling, machine-learning, feature-selection, model comparison**

P. Schratz, J. Muenchow, J. Cortés and A. Brenning are with the Department of Geography, GIScience group, Friedrich-Schiller-University of Jena, Germany.

B. Bischl is head of the computational statistics group at the Department of Statistics, Ludwig-Maximilian-University Munich.

E. Iturritxa is with NEIKER Tecnalia, Vitoria-Gasteiz, Arab, Spain.

Supporting Ecological Decision Making Using Feature-Selection and Variable Importance

I. Introduction

THE use of machine learning (ML) algorithms for analyzing remote sensing data has seen a huge increase in the last decade [1]. This goes in line 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 open-source efforts. Scientists can nowadays relatively easily process large amounts of (environmental) information using various learning algorithms. This makes it possible to extend the matrix of possible options in a semi-automated way, possibly stumbling across unexpected findings of process settings that would have never been tested otherwise [2].

Machine learning methods in combination with remote sensing data are used in many environmental fields such as vegetation cover analysis or forest carbon storage mapping [3], [4]. The ability of predicting to large unknown areas qualifies these tools as a promising toolset for such tasks. One aspect of this research field is to enhance the understanding of biotic and abiotic triggers, for example by analyzing defoliation at trees [5].

Other approaches for analyzing forest health include temporal change detection [6] or describing the current health status of forests on a stand level [7]. In such studies, the defoliation of trees serves as a proxy for forest health by describing the impact of biotic and abiotic pest triggers [7], [8].

Vegetation indices have shown the potential to provide valuable information when analyzing forest health [9], [10]. Most vegetation indices were developed with the aim of being sensitive to changes of specific wavelength regions, serving as a proxy for underlying plant processes. However, often enough indices developed for different purposes than the one to be analyzed can help explaining complex relationships. This emphasizes the need to extract as much information as possible from the available input data to generate promising features which can help understanding the modeled relationship. A less known index type which can be derived from spectral information is called normalized ratio index (NRI). In contrast to vegetation index (VI), normalized ratio index (NRI)s are not based on an expert-based formulas following environmental heuristics but use a data-driven feature engineering approach by combining (arbitrary) combinations of spectral bands. Especially when working with hyperspectral data, hundreds of NRI features can be derived this way.

Despite its popularity in environmental modeling, there are no studies so far which used machine learning algorithms in combination with remote sensing data to analyze defoliation on a tree level. This study aims to close this gap by analyzing defoliation at trees in northern Spain using airborne hyperspectral data. We make use of the latest state-of-the-art methodology in machine learning by combining feature-selection and hyperparameter tuning across multiple algorithms. Incorporating the idea of creating data-driven NRIs, this study also discusses the practical problems of high-dimensionality in environmental modeling [11], [12].

Even though ML algorithms are capable of handling highly-correlated input variables, the fitting time of models increase substantially and the interpretation of results becomes more complicated. At the same time, one has to deal with the presence of spatial autocorrelation in the data. The dataset used in this study comes with a spatial grouping at the plot level which needs to be accounted for.

The research questions of this study are the following:

- Do different environmental feature sets show differences in performance when modeling defoliation at trees?
- Does combining feature sets have an substantial effect on predictive performance?
- How are feature-selection methods influencing the predictive performance of the models?
- Which features are most important for the models and how can these be interpreted in an ecological context?

random forest (RF), support vector machine (SVM) and extreme gradient boosting were fitted on six feature sets using different feature selection methods. Bayesian Optimization, also known as model-based optimization, was used for hyperparameter tuning and optimization of the number of features. By merging all optimization steps into a single step, multiple optimization stages, which may introduce bias cause long model fitting times, were avoided. A spatial cross-validation (CV) approach on the plot level was used to account for spatial autocorrelation in the data.

II. Data and study area

Airborne hyperspectral data with a spatial resolution of one meter and 126 spectral bands was available for four Monterey Pine (*Pinus radiata*) plantations in northern Spain. The trees in the study area plots suffer from infections of invasive pathogens such as *Diplodia sapinea*, *Fusarium circinatum*, *Armillaria mellea* or *Heterobasidion annosum*, leading to a spread of cankers or defoliation [13], [14]. In-situ measurements of defoliation at trees (serving

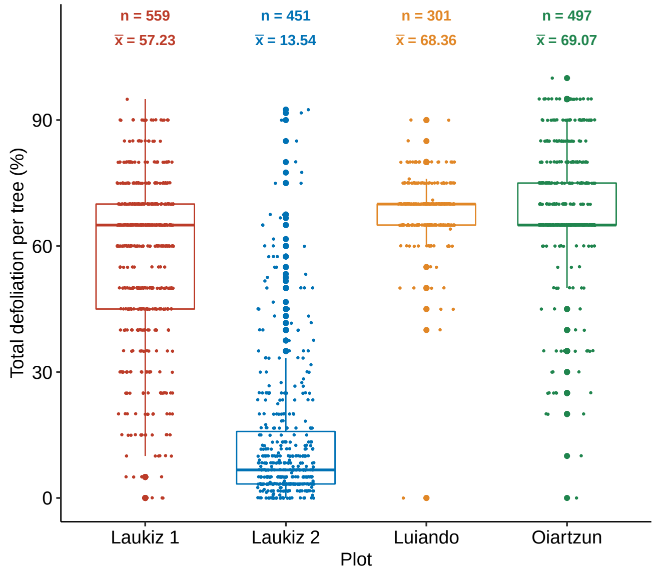

Fig. 1. Response variable *defoliation* at trees for plots *Laukiz 1*, *Laukiz 2*, *Luiando* and *Oiartzun*. n corresponds to the total number of trees in the plot, $\bar{x}$ refers to the mean defoliation.

as a proxy for tree health) were collected to serve as the response variable *defoliation* which ranges from 0% - 100% (Figure 1). The fungi are assumed to infect the trees through open wounds, possibly caused by previous hail damage [15]. The dieback of these trees, which are mainly used as timber, causes high economic damages [16].

A. In-situ data

The *Pinus radiata* plots of this study, namely *Laukiz 1*, *Laukiz 2*, *Luiando* and *Oiartzun*, are located in the northern part of the Basque Country (Figure 2). *Oiartzun* has the most observations (n = 559) while *Laukiz 2* shows the largest area size (1.44 ha). All plots besides *Luiando* are located nearby the coast (Figure 2). In total 1808 observations are available (*Laukiz 1* = 559, *Laukiz 2* = 451, *Luiando* = 301, *Oiartzun* = 497). The data was surveyed in September 2016.

B. Hyperspectral data

The airborne hyperspectral data was acquired during two flight campaigns on September 28th and October 5th 2016, both around 11 am. The images were taken using a AISAEAGLE-II sensor. All preprocessing steps (geometric, radiometric, atmospheric) have been conducted by the Institut Cartografic I Geologic de Catalunya (ICGC). The first four bands are corrupted, leaving 122 bands with valid information. Additional metadata information is available in Table I.

III. Methods

A. Derivation of indices

To use the full potential from 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. The following formula was used for the NRI calculation:

$$NRI_{i,j} = \frac{b_i - b_j}{b_i + b_j} \qquad (1)$$

where i and j are the respective band numbers.

To account for geometric offsets (which were reported with up to 1 m from ICGC) within the hyperspectral data, a buffer of two meters around the centroid of each tree was used during extraction of the reflectance values. Subsequently, the value assigned to a tree observations was the mean of all pixels which were touched by the buffer drawn around each tree. A pixel was considered to fall into a tree's buffer zone if the centroid of the respective pixel was touched by the buffer. This is how the extract() function of the R package {*raster*} handles buffer extraction. In total, $\frac{125*126}{2}$ = 7875 NRIs were calculated. Due to four corrupted bands of the sensor a total of 7471 indices were available for each observation.

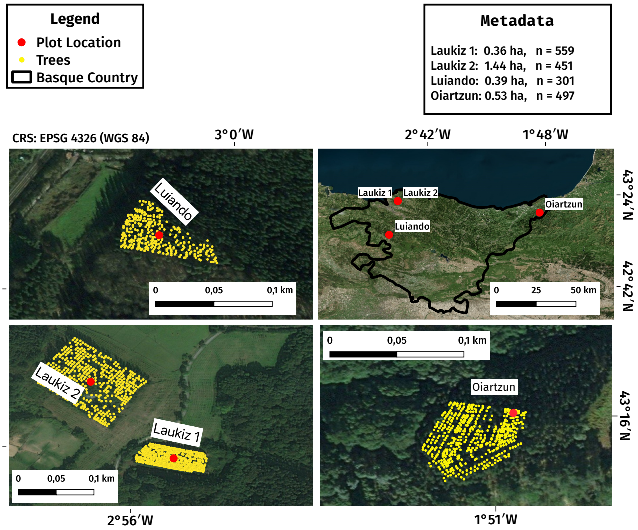

Fig. 2. Information about location, size and spatial distribution of trees for all plots used in this study.

TABLE I
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

B. Dimension reduction

We want to stress the differences between 'dimension reduction' and 'high-dimensionality'. The former refers to the general idea of reducing features from a dataset [17]. In modeling this means finding the best subset of covariates which provide the most predictive power to the model or extracting the main components of the features using a principal component analysis (PCA). 'High-dimensionality' in contrast is a dataset attribute and applies when $p > n$, where p is the number of covariates and n the number of observations [18]. There is no absolute value at which the term applies, both for n and p. Hence for this study the word 'high-dimensionality' only applies to the experiments using the 'NRI' feature set (> 7000 features for 1808 observations).

The case of a feature-rich dataset comes with several challenges for both model fitting and evaluation.

- Model fitting times increase.
- Noise is possibly introduced into models by highly-correlated variables [19].
- Model interpretation and prediction tasks become more challenging [19].

In the following sections a brief overview about sub-categories of feature selection approaches is given. Due to the focus of this study on the use of filter methods, which are a sub-group of feature selection methods, other approaches were grouped into a single section.

1) Filter methods: The concept of filters originates from the idea of ranking features using certain heuristics of an algorithm [20]. Some filter methods are restricted towards specific types of variables (numeric or nominal). Filters do only rank features, they do not choose which covariates to drop or keep [21]. The selection of features to keep for the model fitting step is usually done within the optimization phase of the model fitting, along with the hyperparameter tuning. Essentially, the number of covariates in the model is treated as a hyperparameter of the model. The goal is to optimize the number of features (using the ranked covariates) at which the model achieves the best performance. In well-implemented software solutions the filter calculation is only done once and then cached, saving computational resources [22].

a) Ensemble filter methods: Besides the concept of choosing a specific filter method to rank variables, studies showed that combining several filters using statistical operations such as 'minimum', 'mean' and 'sum' are able to enhance the predictive performance of the resulting models [23], [24]. This approach is referred to as 'ensemble filter' [25]. Ensemble filters align with the recent rise of the 'ensemble' approach in machine learning which uses stacking to combine the predictions of multiple models, aiming to enhance predictive performance [26], [27]. In this work the 'Borda' ensemble filter was applied [24]. For this filter, the final order is the sum of all single filters.

b) Ensuring a fair weighting in the ensemble: Filter methods can be grouped into classes: Correlation based, entropy based, linear and non-linear methods. It is important to not give certain classes too much weight in the ensemble as otherwise the final result will be biased towards these. 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. In case pairs of filters with a correlation above 0.9 were discovered, only one of these was included into the ensemble filter. By this we ensured that the ensemble filter is not biased towards a certain group of methods yielding highly similar rankings.

c) Description of used filter methods: Filter methods can be classified as follows:

- univariate/multivariate (scoring based on a single variable / multiple variables)
- linear/non-linear (calculation of linear/non-linear interaction terms)
- entropy/correlation (scoring based on derivations of entropy or correlation based approaches)

Filter 'Information Gain' is only defined for nominal response variables:

$$H(Class) + H(Attribute) - H(Class, Attribute) \quad (2)$$

where H is the conditional entropy of the response variable (class) or the feature (attribute), respectively. To be able to use this method with a numeric response (defoliation in our case), the variable is discretized into equal bins and treated as a class variable. While the number of bins can be treated as a hyperparameter of the filter method, we decided to use $n_{bin} = 10$ after rank correlations of > 0.9 for different bin sizes were observed in a side analysis.

2) Wrapper methods and PCA: Other approaches to assess feature importance are so called 'wrapper methods' and the PCA [34], [35].

Wrappers [20], [36] apply algorithms that are also used for hyperparameter optimization such as 'Random Search' or 'Generic Simulated Annealing'. First a (random) subset of features is chosen based on the selected algorithm. In comparison to filters, no ranking is done in this step. Now the model is fitted on the data and the performance is evaluated. This is done multiple times, depending on the defined stopping criteria set by the user (performance, runtime, evaluations). A disadvantage of this approach is that hyperparameter tuning can only be applied after the feature-selection optimization has finished. Hence, the 'wrapper approach' is an expensive optimization method because two stages need to be run in sequential order. Due to their extensive runtimes, wrappers were not considered in this work.

A method with a completely different approach compared to filters and wrappers is the PCA [28], [35]. Here, the main components of the feature space are extracted and combined. Most often the first two extracted main components are used since these contain the major information of the covariates. By using the (automatically

TABLE II
List of filter methods used in this work

Name	Group	Ref.
Linear correlation (Pearson)	univariate, linear, correlation	[28]
Information gain	univariate, non-linear, entropy	[29]
Minimum redundancy, maximum relevance	multivariate, non-linear, entropy	[30]
Carscore	multivariate, linear, correlation	[31]
Relief	multivariate, linear, entropy,	[32]
Conditional minimal information maximization	multivariate, linear, entropy	[33]

estimated) explained variance of the main components, the model can rely on a few features containing the majority of information available in the data. This enables cheap model fitting with balanced loss of predictor information. The disadvantage of this methodology is the lack of interpretability because the main components cannot be related back to the original covariates.

C. Benchmarking design

1) Algorithms: The benchmarking matrix of this study consists of the following algorithms:

- Extreme Gradient Boosting (XGBOOST)
- Random Forest (RF)
- Penalized Regression (both L1 (Lasso) and L2 (Ridge))
- Support Vector Machine (SVM)

RF and SVM are well established algorithms that are widely used in environmental modeling. extreme gradient boosting (XGBOOST) (commonly abbreviated as XGBOOST) showed promising results in benchmark competitions in recent years. Penalized regression is a statistical modeling technique capable of dealing with highly-correlated covariates by applying a penalization term which shrinks the coefficients of the model [18]. Common penalties are 'lasso' (L1) and 'ridge' (L2). The former does not allow the full removal of variables from the model (penalization to zero) while the latter does. Both penalties can also be combined. The combined approach is called 'elastic net' but was not used in this work.

2) Feature sets: Three feature sets were used in this study with each representing a different way of feature engineering:

- The raw hyperspectral band information (HR): No feature engineering)
- Vegetation Indices (VI): Expert-based feature engineering)
- Normalized Ratio Indices (NRI): Data-driven feature engineering)

The idea of splitting the features into different sets originated from the question whether feature engineered indices from reflectance values have a positive effect on model performance. Benchmarking all models on these distinct groups of features makes it easier to draw conclusions on their impact when keeping all other variables such as model type, tuning strategy and partitioning method constant. However, rather than only looking at these three groups we decided to also take their combinations into account for the overall comparison:

- HR + VI
- HR + NRI
- HR + VI + NRI

Even though the feature selection task was handed over to the filter methods in this study, we ensured to not include features with a pair-wise correlation of 1. Having such can cause undesired effects during model fitting and feature importance calculation. To account for such, all pair-wise correlations between features were calculated. For pairs which exceeded the threshold of 0.9999999999 the one with the largest mean absolute correlation was removed. This process was repeated p times, each time calculating a new correlation matrix.

This preprocessing step resulted in the following final number of predictors: HR (122), VI (86), NRI (7467).

3) Hyperparameter Optimization: An exhaustive hyperparameter tuning was applied during nested spatial CV for all algorithms. Model-based Optimization [37] (MBO) was used for hyperparameter optimization. This approach first composes n randomly chosen hyperparameter settings out of a user defined search space. After these n tries have been evaluated, a new hyperparameter setting, going to be evaluated next, is proposed by a fitted surrogate model (by default a kriging method). This strategy continues until a termination criterion, defined by the user, is reached [38], [39].

An initial design of 30 randomly composed hyperparameter settings and a termination 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 the substantial reduction of the tuning budget required to find a setting which is close to the global minimum. model-based optimization (MBO) shines when being compared to methods that do not use information from previous runs, such as random search or grid search [40].

For the filter methods, the percentage of features was added to the models as a hyperparameter. For PCA, the number of main components was tuned instead. Random Forest hyperparameter m_{try} was modified into power transformed version $p^{m_{try}}$ (ranging from 1 to p) to work on relative values of the respective datasets feature count. Here p is the number of features of the dataset. This was necessary to ensure that m_{try} was not chosen out of bounds during tuning with respect to the datasets feature count. After the filtering, only a subset of features of the task is used for optimizing the models hyperparameters. The size of this subset is always different since the number of features is also optimized. Hence, the value of m_{try} needs to be created dynamically based on the datasets feature count.

4) Spatial resampling: A spatial nested cross-validation on the plot level was chosen to reduce the influence of spatial autocorrelation as much as possible [41], [42]. Each plot served as one fold within the cross-validation setting, resulting in four iterations total. For the inner level (hyperparameter tuning), $y - 1$ folds were used (with y being the number of plots).

In total the benchmarking matrix consisted of 156 experiments (6 feature sets * 3 ML algorithms * 8 feature-selection methods + 2 * 6 L1 and L2 models).

D. Feature importance estimation

Estimating feature importance for highly-correlated datasets is a complicated task. The correlation between covariates makes it challenging to calculate an unbiased estimate for single features. Methods like partial dependence plots (PDP) do not produce reliable estimates in such scenarios because unrealistic situations between covariates are created [43]. Due to the noisiness of the dataset, estimates from model-agnostic approaches such as permutation-based feature importance should also be taken with care. This method calculates the loss of predictive performance for each feature by permuting it in a random manner. The more important the feature is for the model, the higher the variance in the measure, reflecting the importance of the specific feature for the current model. Permuting every feature p times comes with some computational cost. To balance the effort/gain ratio and cope with limited visual space in plots, we estimated the feature importance only for datasets HR and VI using the best performing learner of the benchmark.

However, as of today there are no unbiased approaches for estimating feature importance for (highly) correlated datasets. Nevertheless, recent methods like accumulated local effects (ALE) plots aim to tackle this problem by utilizing marginal distributions of features over a small window to focus on the effect of a single feature [43], [44]. However, ALE plots need visual inspection and as the number of features increases, making objective comparisons becomes more complicated.

In this work we applied permutation-based feature importance and ALE plots to calculate feature importance for the HR dataset. With the limitations of both methods in mind we aimed to get a general overview of the feature importance of the hyperspectral bands while keeping modest on over-interpreting the results. We used the algorithm which showed the best performance for the HR (SVM) task to calculate the feature importance. ALE plots were not discussed in greater detail and are available as supplementary material.

E. Linking feature importance to wavelength regions

For ecological interpretation purposes we linked the ten most important indices of the winning models for each feature set to the spectral regions of the hyperspectral data. For feature set HR and NRI a direct linking to the respective bands of the hyperspectral sensor was done. For the vegetation indices all bands covered by the spectral range of calculated vegetation indices were counted and summed up.

F. Research compendium

The complete study was done using the open-source statistical programming language R [45]. All R packages used in this study can be found in linked repositories. Due to space limitations we will only mention the packages of the used algorithms and filter methods.

The algorithm implementations of the following packages have been used: *xgboost* [46] (*xgboost*), *kernlab* [47] (Support Vector Machine) and *glmnet* [48] (Ridge Regression). The filter implementations of the following packages have been used: *praznik [49]*, *FSelectorRcpp* [50]. The R package *mlr* [22] was used for all modeling related steps. *drake* [51] was used for structuring the work and reproducibility. This study is available as a research compendium on Zenodo (10.5281/zenodo.2635403). The code base is available on GitHub (https://github.com/pat-s/2019-feature-selection).

IV. Results

A. Predictive performance

Overall, the response variable 'defoliation at trees' could be modeled with an error of 28 %. SVM showed no differences in RMSE across feature sets whereas other learners like RF differed up to seven percentage points (HR-NRI vs. VI) (Figure 3). Ridge and Lasso faced major issues up to the point of not achieving an error below 50 for datasets HR-NRI-VI and NRI-VI. SVM showed the best performance of all learners with an absolute difference up around three percentage points to the next model (RF) (Table IV). A high intra-fold variance was observed: Predicting on Luiando resulted in an RMSE value of 9.0 for learner SVM (without filter) but up to 54.26 RMSE when testing on Laukiz2 (Table V).

The combination of feature sets showed small increases in performance for some learners. RF and XGBoost scored slightly better on the combined datasets HR-NRI and NRI-VI, respectively (Figure 3). Datasets containing derived features (VI, NRI) showed no improvement in performance compared to the raw hyperspectral band information (HR). All learners besides SVM show a substantially worse performance on the VI dataset (around five percentage points).

SVM combined with the 'Carscore' filter achieved the best performance (RMSE of 27.98) (Table III). Regression with Ridge penalty (L2) showed a high variance when comparing results across tasks: In two out of six tasks (all including VI variables) the error was out of bounds (HR-NRI-VI, HR-VI, VI). For HR-NRI and HR the error was around 40 RMSE, and for task NRI the Ridge learner achieved almost the same score as the best performing model (32.9 RMSE) (Table III). In all settings for which Ridge showed such a high error, only one observation in one fold was predicted which such a high value (in the millions). This then caused the error estimates of these folds and the average estimate across all folds to be out of bounds.

FIXME: Mention: All penalized methods tuned using MBO showed exactly the same error across all tasks. FIXME: Mention the bad perf of RIDGE on VI

Effects of filter methods on performance differed greatly between algorithms: SVM showed no variation in performance across filters (Figure 4). Using filters for RF showed a substantial increase in performance for all tasks besides VI for which the overall difference of all filters was also smallest (Figure 4). XGBoost showed a high dependency on filtering the data: In 4 out of 6 tasks using no filter would results in the worst or second worst performance. However, in contrast for dataset NRI using no filter results in the best performance. XGBoost shows the highest overall differences between filters for a single task: For dataset HR, the range is up to 14 percentage points (Carscore vs. no filter)(Figure 4).

When comparing the usage of filters against using no filter at all, there was only one instance (XGBoost on the NRI task) when the model without filtering showed a slighly better performance than the best filtered one (Figure 4). For SVM filters and no filter come in at the same performance for tasks VI and NRI even though Figure 4 lists ”No Filter” as the best option.

The Borda filter did not achieve the best performance for any learner across any task Figure 5. For RF and XGBoost it ranked within the first 50% when looking at all filters used for a task. For XGBoost on the VI task the Borda filter scored the second worst performance.

TABLE III
TOP 10 RESULTS FOR ANY TASK/LEARNER/FILTER COMBINATION, SORTED BY PERFORMANCE.

	Task	Model	Filter	RMSE	SE
1	HR-NRI	SVM	Car	27.98	19.19
2	HR-NRI-VI	SVM	Relief	28.05	19.12
3	HR-NRI	SVM	Relief	28.12	19.12
4	HR	SVM	Car	28.12	19.12
5	HR	SVM	Info Gain	28.12	19.12
6	VI	SVM	Relief	28.12	19.11
7	HR	SVM	CMIM	28.12	19.12
8	NRI-VI	SVM	PCA	28.12	19.12
9	HR-NRI-VI	SVM	PCA	28.12	19.12
10	VI	SVM	No Filter	28.12	19.12

TABLE IV
BEST PERFORMANCE OF EACH LEARNER ACROSS ANY TASK AND FILTER METHOD.

	Task	Model	Filter	RMSE	SE
1	HR-NRI	SVM	Car	27.98	19.19
2	HR-NRI	RF	Car	31.08	17.09
3	NRI-VI	XGBOOST	Relief	32.10	17.73
4	NRI	Ridge-CV	No Filter	32.97	7.24
5	HR	Lasso-CV	No Filter	45.96	18.35

TABLE V
SINGLE FOLD PERFORMANCES FOR LEARNER SVM ON THE HR DATASET WITHOUT USING A FILTER.

	Plot	RMSE
1	Luiando	9.00
2	Laukiz1	21.17
3	Laukiz2	54.26
4	Oiartzun	28.05

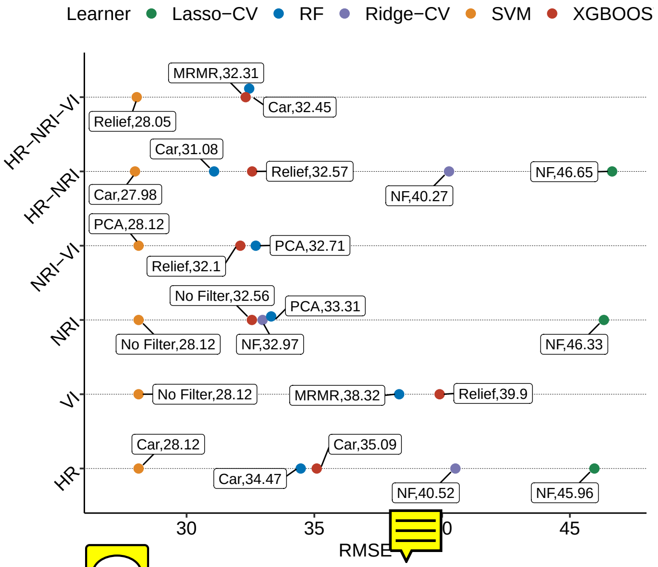

Fig. 3. Scored predictive performance (RMSE) of models across tasks. Prefix 'CV' denotes that the learner was optimized using internal 10-fold CV while prefix 'MBO' means that Bayesian optimization was used for hyperparameter optimization. The abbreviations on the y-axis refer to the combinations of feature sets on which each model was scored on. Labels attached to each point in space show which filter method was used for scoring features during the feature selection process (NF = no filter, Car = 'Carscore', Info = 'Information Gain', Borda = 'Borda'). The second value in the attached label of each point shows the scored RMSE value of the respective setting.

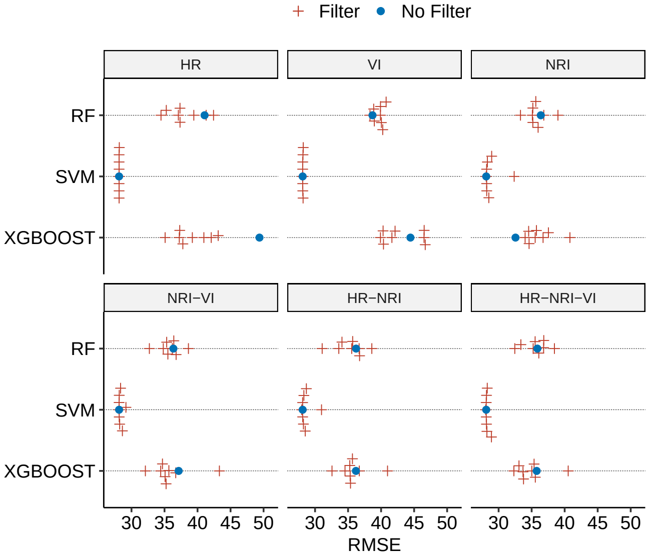

Fig. 4. Model performances in RMSE when using no filter method compared to all other filters across all tasks.

B. Variable importance

1) Permutation-based Variable Importance: The most important features of each dataset showed an average decrease in RMSE of 1.57 (HR, B69) and 1.79 (VI, Vogelmann2) (Figure 6). For both datasets most features among the ten most important ones cluster around a wavelength range of 700 nm - 750 nm (the so called ”red-edge”). For feature set HR four features in the infrared region (920 nm - 1000 nm) were identified by the model to be most important (causing a mean decrease in RMSE of around one percentage point). Overall, most features show only a small importance with average decreases in RMSE below 0.5.

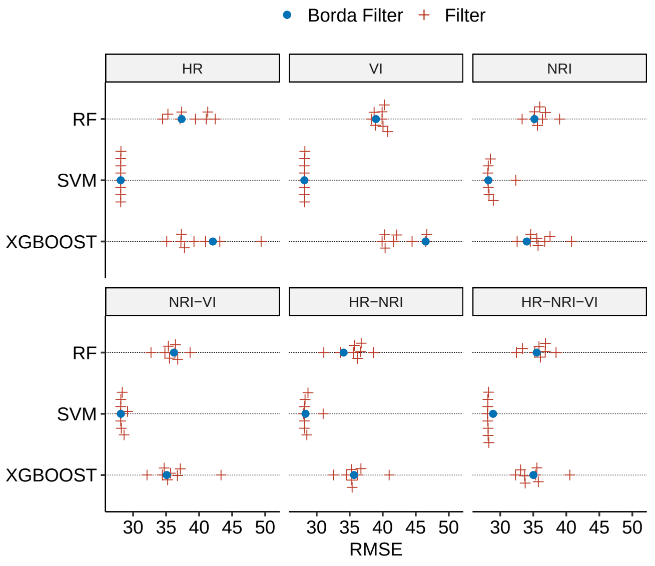

Fig. 5. Model performances in RMSE when using the Borda filter method compared to all other filters for each learner across all tasks.

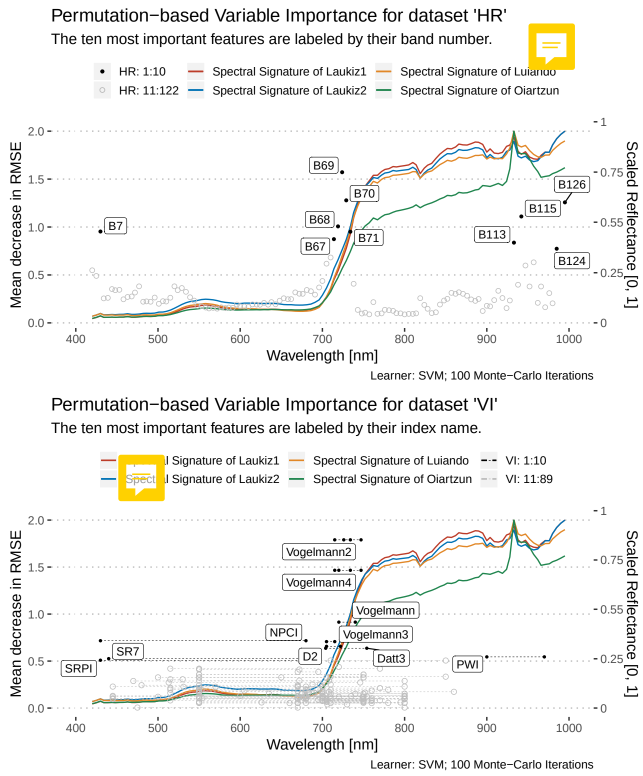

Fig. 6. Variable importance for feature sets HR and VI: Mean decrease in RMSE for one-hundred feature permutations using the SVM learner. The wavelength range on the x-axis matches the range of the hyperspectral sensor (400 nm - 1000 nm). For each dataset, the ten most important features were highlighted as black dots and labeled by name. Grey 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.

V. Discussion

A. Derivation of Indices

The decision to use a buffer of 2 m to generate the index value for each observation has to be seen critical. Due to the reported geometric offset of up to 1 m for the hyperspectral data, the possibility of assigning a wrong pixel value to a tree observation would be high. Using a buffer of more than 2 meters increases the probability of including information from other trees into the pixel value, blurring the actual value of the tree observation. In our view using a buffer of 2 m is a good compromise between both worlds. In addition, a s[illegible]nalysis showed that some trees would be assigned missing values when relying on a single pixel. This then would result in the removal of that particular tree which we wanted to avoid. Even though we provide no results showing the influence of multiple buffer values on the extraction, we hypothesize that the relationships between features would not change substantially, leading to the almost identical model results.

Another critical point is that the exact number of contributing pixels to the final index value of an observation cannot be determined as it depends on the location of the tree within the pixel grid. According to the extract function of the raster package, a pixel is included if the buffer of the point observation hits the centroid of the respective pixel, not just any part of it. As the buffer is circular, determining the total number of contributing pixels for each tree depends on the exact location of a tree within the pixel grid. Next, if a tree observation is located at the border of the plot, some directions of the buffer will contain no values and the subsequent index value will be calculated using [illegible] pixels than if the tree observation is located in the middle of the plot. The magnitude of uncertainty introduced by these facts cannot be quantified. However, we assume these to be of minor importance for the subsequent model fitting step.

B. Performance vs. plot characteristics

The high differences in RMSE observed for the fold performances (Table V) are a result of model overfitting. An RMSE of 54.[illegible] indicates a complete failure of the model in the prediction stage for this plot (Laukiz2). Laukiz2 differs highly in the distribution of the response variable defoliation compared to all other plots (Figure 1). The model overfitted on learning relationships related to medium-high defoliation values but missed out on low defoliation values. In cases when Laukiz2 is in the training set, the model is able to half the error and even reduce it down to a value of [illegible] RMSE in the case of Luiando.

C. Predictive Performance

The best aggregated performance of this study (SVM + Carscore filter, 27.98 RMSE) has to be seen in the light of model overfitting (see subsection V-B). Leaving out the performance on Laukiz2 when aggregating results, the mean RMSE would be 19. 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. With the clear presence of model overfitting in this study it can be postulated that more training data representing all health stages of a plot is 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 making statements about the overall performance of the fitted models of this study due to their high variance when predicting. However, when inspecting the fold level performances, one can say that the model does a decent job predicting defoliation ranging from 50% to 100% but fails for 0% - 50%. This applies to all learners of this study.

1) Algorithm differences: An interesting find is the strength of the SVM algorithm when comparing its predictive performance to its competitors (Table IV).

FIXME: Add refs There were few environmental modeling studies only in recent years in which SVM outperformed other models that clearly. The absolute difference of around three percentage points is also worth noting: We could not find a study in which SVM came out as the best model in such clear way. All of these statements are based on the assumption that a proper hyperparameter tuning was conducted for all models (which we also claim for ourselves in this study).

We were a bit surprised about the performances of the penalized methods Lasso and Ridge. The former was not able to achieve an error below 45.96 RMSE across any dataset while Ridge came at least somewhat close to RF and XGBoost for some feature sets. However, there were also three instances (VI, NRI-VI, HR-NRI-VI) for which the error of the fitted Ridge model was out of bounds. We cannot preclude possible mistakes on our side when optimizing the penalization terms of the penalized learners.

FIXME: Talk about L1 and L2 model performance

2) Feature set differences: One objective which this study aimed to answer was whether expert-based or data-driven feature engineering has a positive influence on model performance. With respect to Figure 3 one can say that overall the different feature sets did not result in substantial changes in performance. RF and XGBoost showed a somewhat surprising decrease in performance of about six percentage points on task VI compared to all others. We have no explanation for this specific result, especially because the SVM does not show this difference.

FIXME: Talk about L1 and L2 models -¿ there we have quite some differences!

D. Feature selection methods

The usefulness of filters in this study was apparent across all feature sets, even when p was rather small (HR). Even though the effect of feature selection methods varied across algorithms, small to substantial improvement in predictive performance were achieved in almost all cases. No optimal algorithm/filter/dataset setting was observed in this study, leading to the conclusion that one needs to try out various combinations of such to find the best performing one.

We have no explanation why the Borda ensemble method used in this study did not score better than the average performance of other filter methods. Ensemble methods shine in complicated scenarios when simple filter methods face problems due to their focus on certain areas of features. We assume that the datasets used in this study were not complex enough to bring up these problems for simple features. Therefore it makes sense that, due to its averaging nature, the Borda filter scored average results across all instances. However, this cannot be known upfront and we believe that an ensemble filter should be part of every filter portfolio when doing filter-based feature selection.

The PCA approach did neither show the best or worst result in the lineup. It shines in terms of runtime since it reduces the amount of features highly. However, it also removes the option for a post-analysis of feature importance by squashing predictors into principal components and users need to decided upfront what they value more. Since filters are only calculated once due to caching, the runtime advantage might in fact be negligible in practice.

1) Linking feature importance to spectral characteristics: Not surprisingly the most important features for both HR and VI datasets were identified around the red edge of the spectra, specifically in the range of 700 nm to 750 nm.

FIXME: Add ref This area has the highest ability to distinguish between reflectances related to a high density / high healthiness of vegetation and its opposite. It is also worth mentioning that four out of ten of the most important features of dataset HR lie between 920 nm and 1000 nm. Looking at the spectral curves of the plots, we can observe quite some variance in this area, especially for Oiartzun, which might explain why these features were considered as important by the model.

E. Comparison to other studies

Most other studies analyzing defoliation operated on the plot rather than the tree level. This is due to the low spatial resolution of used satellite products which served as the input data, making a tree-level study infeasible [7], [52], [53].

Studies focusing on tree-level defoliation used ground-level methods such as airborn laser scanning (ALS) or light detection and ranging (LiDAR) [54], [55]. [54] used ordinary least squares (OLS) regression methods while [55] retrieved information on ground-level RGB photos using convolutional neural networks (CNN). Both study designs are substantially different compared to the setup of this work. In addition, no spatial CV or feature-selection (FS) was used. [8] 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 metrics were found to be most helpful for the model. Incorporating such metrics (both spectral and structural) could be a possible enhancement for future works.

[56], [57] are studies which are more similar in their methodology but focus on a different response variable (woody cover). [56] used machine learning with ALS data to study dieback of trees for eucalyptus forests. A grid-search was used for hyperparameter tuning and forward feature-selection (FFS) for variable selection. [57] analyzed woody cover in South Africa using spatial CV and FS approach [58] with a Random Forest classifier.

In summary, we could not find studies using filter methods for FS or NRI indices in their work with a relation to forest health. Most studies used only one algorithm (usually Random Forest) without (strong) arguments why this particular one has been selected or why only one model was used. This is not surprising: Most environmental/ecological datasets are not high-dimensional. In contrast, the number of predictors is often mo[illegible]c and issues related to correlations are solved manually instead of relying on an automated approach. The bioinformatics field faces high-dimensional feature sets more often. Hence more studies using (filter-based) feature-selection approaches can be found for this field [59], [60]. If a field only rarely faces high-dimensional dataset issues the motivation and expertise of using advanced methods to solve such are rather low. We hope that this work can give some guidance and serve as a starting point for tackling high-dimensional problems in environmental modeling.

VI. Outlook and conclusion

This study analyzed defoliation at trees in north Spain by using hyperspectral data as input for machine learning models making heavy use of filter-based feature selection methods. It was shown that substantial differences in performance can occur depending on which feature selection methods and machine learning algorithms are 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 showed their ability to improve predictive performance for datasets with many features. Ensemble filter methods did not show a substantial improvement over le[illegible]y filter methods.

FIXME: ”derivated” is not a real word? Creating feature sets composed out of derivated features did not help. In contrast, dataset VI showed a substantially worse performance than the dataset with the original features (HR). Combining feature sets did not show a substantial improvement on predictive performance.

Features along the red-edge wavelength region were most important for models to achieve good predictions. With respect to dedicated vegetation indices, the ”Vogelmann” index with all of its versions was seen as the most important index for the tested SVM algorithm.

The potential of predicting defoliation for single trees with the given toolset of this study was rather limited, seen on the average error of 27%. However, with better training data covering more variety of the response variable, performance could be highly improved in future studies.

Appendix A
SVM ALE plots for task HR

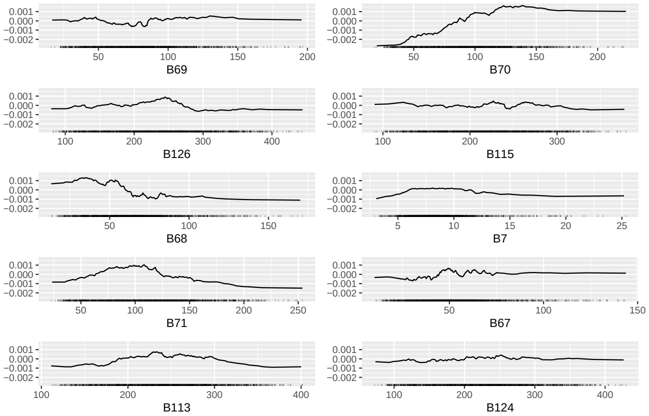

Fig. 7. ALE plots on dataset HR of SVM learner. Subset showing the ten most important features according to the permutation-based variable importance. The y-axis shows the deviation to the mean prediction for each feature, with the mean prediction being centered at zero.

Appendix B
SVM ALE plots for task VI

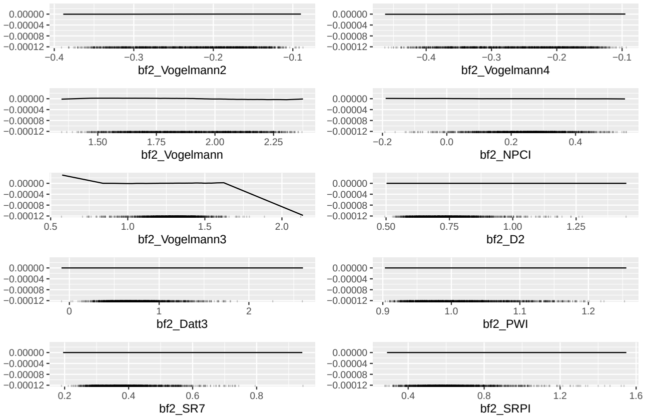

Fig. 8. ALE plots on dataset VI of SVM learner. Subset showing the ten most important features according to the permutation-based variable importance. The y-axis shows the deviation to the mean prediction for each feature, with the mean prediction being centered at zero.

Appendix C
Correlation among filter methods

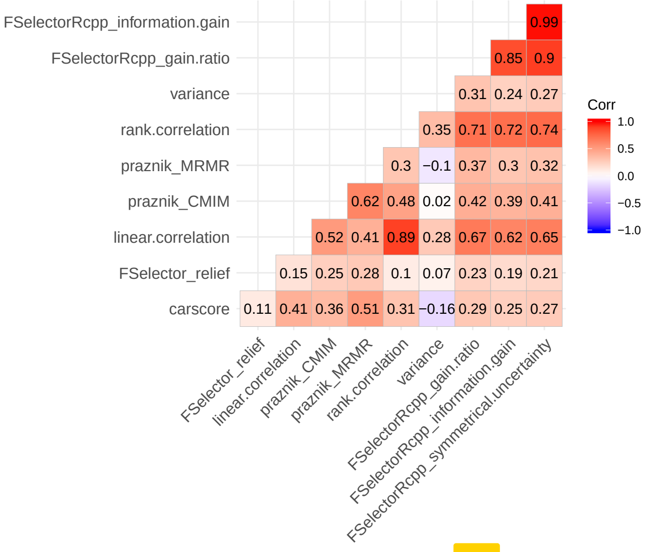

Fig. 9. Spearman correlation of filter rankings between various filter methods. Results of the NRI feature set are shown.

Appendix D
Effect of different n_{bins} values on filter 'information gain'

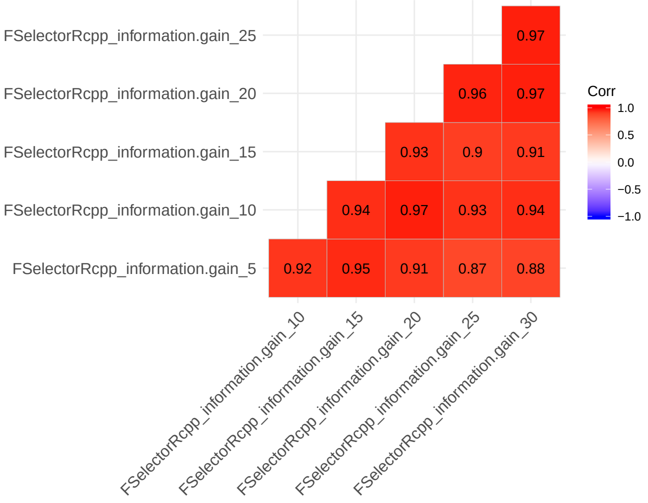

Fig. 10. Spearman correlation of filter information gain using different n_{bins} values for discretization of the numeric response.

References

[1] D. J. Lary, A. H. Alavi, A. H. Gandomi, and A. L. Walker, "Machine learning in geosciences and remote sensing," *Geoscience Frontiers*, vol. 7, no. 1, pp. 3–10, Jan. 2016, 00000.

[2] Y. Ma, H. Wu, L. Wang, B. Huang, R. Ranjan, A. Zomaya, and W. Jie, "Remote sensing big data computing: Challenges and opportunities," *Future Generation Computer Systems*, vol. 51, pp. 47–60, Oct. 2015, 00000.

[3] J. Mascaro, G. P. Asner, D. E. Knapp, T. Kennedy-Bowdoin, R. E. Martin, C. Anderson, M. Higgins, and K. D. Chadwick, "A Tale of Two "Forests": Random Forest Machine Learning Aids Tropical Forest Carbon Mapping," *PLOS ONE*, vol. 9, no. 1, p. e85993, Jan. 2014, 00074.

[4] M. Urban, C. Berger, T. E. Mudau, K. Heckel, J. Truckenbrodt, V. Onyango Odipo, I. P. J. Smit, and C. Schmullius, "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*, vol. 10, no. 9, p. 1482, Sep. 2018, 00000.

[5] P. Hawryło, B. Bednarz, P. Wezyk, and M. Szostak, "Estimating defoliation of Scots pine stands using machine learning methods and vegetation indices of Sentinel-2," *European Journal of Remote Sensing*, vol. 51, no. 1, pp. 194–204, Jan. 2018, 00000.

[6] K. Zhang, B. Thapa, M. Ross, and D. Gann, "Remote sensing of seasonal changes and disturbances in mangrove forest: A case study from South Florida," *Ecosphere*, p. e01366, 2016, 00000.

[7] P. A. Townsend, A. Singh, J. R. Foster, N. J. Rehberg, C. C. Kingdon, K. N. Eshleman, and S. W. Seagle, "A general Landsat model to predict canopy defoliation in broadleaf deciduous forests," *Remote Sensing of Environment*, vol. 119, pp. 255–265, Apr. 2012, 00064.

[8] T. R. H. Goodbody, N. C. Coops, T. Hermosilla, P. Tompalski, G. McCartney, and D. A. MacLean, "Digital aerial photogrammetry for assessing cumulative spruce budworm defoliation and enhancing forest inventories at a landscape-level," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 142, pp. 1–11, Aug. 2018, 00000.

[9] Y. Jiang, T. Wang, C. A. J. M. de Bie, A. K. Skidmore, X. Liu, S. Song, L. Zhang, J. Wang, and X. Shao, "Satellite-derived vegetation indices contribute significantly to the prediction of epiphyllous liverworts," *Ecological Indicators*, vol. 38, pp. 72–80, Mar. 2014, 00000.

[10] J. Adamczyk and A. Osberger, "Red-edge vegetation indices for detecting and assessing disturbances in Norway spruce dominated mountain forests," *International Journal of Applied Earth Observation and Geoinformation*, vol. 37, pp. 90–99, May 2015, 00000.

[11] G. V. Trunk, "A Problem of Dimensionality: A Simple Example," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-1, no. 3, pp. 306–307, Jul. 1979, 00279.

[12] H. Xu, C. Caramanis, and S. Mannor, "Statistical Optimization in High Dimensions," *Operations Research*, vol. 64, no. 4, pp. 958–979, Jul. 2016, 00000.

[13] N. Mesanza, E. Iturritxa, and C. L. Patten, "Native rhizobacteria as biocontrol agents of Heterobasidion annosum s.s. and Armillaria mellea infection of Pinus radiata," *Biological Control*, vol. 101, pp. 8–16, Oct. 2016, 00000.

[14] E. Iturritxa, T. Trask, N. Mesanza, R. Raposo, M. Elvira-Recuenco, and C. L. Patten, "Biocontrol of Fusarium circinatum infection of young Pinus radiata trees," *Forests*, vol. 8, no. 2, p. 32, Jan. 2017, 00000.

[15] E. Iturritxa, N. Mesanza, and A. Brenning, "Spatial analysis of the risk of major forest diseases in Monterey pine plantations," *Plant Pathology*, vol. 64, no. 4, pp. 880–889, 2014, 00000.

[16] R. J. Ganley, M. S. Watt, L. Manning, and E. Iturritxa, "A global climatic risk assessment of pitch canker disease," *Canadian Journal of Forest Research*, vol. 39, no. 11, pp. 2246–2256, Nov. 2009, 00000.

[17] L. van der Maaten, E. Postma, and H. Herik, "Dimensionality Reduction: A Comparative Review," *Journal of Machine Learning Research - JMLR*, vol. 10, Jan. 2007, 00000.

[18] T. Hastie, J. Friedman, and R. Tibshirani, *The Elements of Statistical Learning*. Springer New York, 2001, 00000.

[19] Johnstone Iain M. and Titterington D. Michael, "Statistical challenges of high-dimensional data," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engi-*

neering Sciences, vol. 367, no. 1906, pp. 4237–4253, Nov. 2009, 00000.

[20] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," *Computers & Electrical Engineering*, vol. 40, no. 1, pp. 16–28, Jan. 2014, 01259.

[21] P. Drotár, J. Gazda, and Z. Smékal, "An experimental comparison of feature selection methods on two-class biomedical datasets," *Computers in Biology and Medicine*, vol. 66, pp. 1–10, Nov. 2015, 00029.

[22] B. Bischl, M. Lang, L. Kotthoff, J. Schiffner, J. Richter, E. Studerus, G. Casalicchio, and Z. M. Jones, "mlr: Machine learning in R," *Journal of Machine Learning Research*, vol. 17, no. 170, pp. 1–5, 2016, 00000.

[23] T. Abeel, T. Helleputte, Y. Van de Peer, P. Dupont, and Y. Saeys, "Robust biomarker identification for cancer diagnosis with ensemble feature selection methods," *Bioinformatics*, vol. 26, no. 3, pp. 392–398, Feb. 2010, 00000.

[24] P. Drotár, M. Gazda, and J. Gazda, "Heterogeneous ensemble feature selection based on weighted Borda count," in *2017 9th International Conference on Information Technology and Electrical Engineering (ICITEE)*, Oct. 2017, pp. 1–4, 00000.

[25] T. G. Dietterich, "Ensemble Methods in Machine Learning," in *Proceedings of the First International Workshop on Multiple Classifier Systems*. Springer-Verlag, Jun. 2000, pp. 1–15, 05299.

[26] R. Polikar, "Ensemble Learning," in *Ensemble Machine Learning: Methods and Applications*, C. Zhang and Y. Ma, Eds. Boston, MA: Springer US, 2012, pp. 1–34, 00224.

[27] M. Feurer, A. Klein, K. Eggensperger, J. Springenberg, M. Blum, and F. Hutter, "Efficient and Robust Automated Machine Learning," in *Advances in Neural Information Processing Systems 28*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds. Curran Associates, Inc., 2015, pp. 2962–2970, 00000.

[28] K. Pearson, "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*, vol. 2, no. 11, pp. 559–572, Nov. 1901, 00000.

[29] J. R. Quinlan, "Induction of decision trees," *Machine Learning*, vol. 1, no. 1, pp. 81–106, Mar. 1986, 20401.

[30] X.-M. Zhao, "Maximum Relevance/Minimum Redundancy (MRMR)," in *Encyclopedia of Systems Biology*, W. Dubitzky, O. Wolkenhauer, K.-H. Cho, and H. Yokota, Eds. New York, NY: Springer New York, 2013, pp. 1191–1192, 00000.

[31] V. Zuber and K. Strimmer, "High-Dimensional Regression and Variable Selection Using CAR Scores," *Statistical Applications in Genetics and Molecular Biology*, vol. 10, no. 1, 2011, 00000.

[32] K. Kira and L. A. Rendell, "The feature selection problem: Traditional methods and a new algorithm," in *Proceedings of the Tenth National Conference on Artificial Intelligence*. AAAI Press, Dec. 1992, pp. 129–134, 01970.

[33] F. Fleuret, "Fast Binary Feature Selection with Conditional Mutual Information," *The Journal of Machine Learning Research*, vol. 5, pp. 1531–1555, Jan. 2004, 00000.

[34] S. Das, "Filters, Wrappers and a Boosting-Based Hybrid for Feature Selection," in *ICML*, 2001, 00000.

[35] I. Jolliffe and J. Cadima, "Principal component analysis: A review and recent developments," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 374, no. 2065, p. 20150202, Apr. 2016, 00000.

[36] R. Kohavi and G. H. John, "Wrappers for feature subset selection," *Artificial Intelligence*, vol. 97, no. 1, pp. 273–324, Dec. 1997, 00000.

[37] B. Bischl, J. Richter, J. Bossek, D. Horn, J. Thomas, and M. Lang, "mlrMBO: A Modular Framework for Model-Based Optimization of Expensive Black-Box Functions," *ArXiv e-prints*, Mar. 2017, 00000.

[38] F. Hutter, H. H. Hoos, and K. Leyton-Brown, "Sequential model-based optimization for general algorithm configuration," in *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2011, pp. 507–523, 00000.

[39] D. R. Jones, M. Schonlau, and W. J. Welch, "Efficient global optimization of expensive black-box functions," *Journal of Global Optimization*, vol. 13, no. 4, pp. 455–492, Dec. 1998, 00000.

[40] J. Bergstra and Y. Bengio, "Random Search for Hyper-parameter Optimization," *J. Mach. Learn. Res.*, vol. 13, pp. 281–305, Feb. 2012, 00000.

[41] P. Schratz, J. Muenchow, E. Iturritxa, J. Richter, and A. Brenning, "Hyperparameter tuning and performance assessment of statistical and machine-learning algorithms using spatial data," *Ecological Modelling*, vol. 406, pp. 109–120, Aug. 2019, 00000.

[42] A. Brenning, "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, Jul. 2012, R package version 2.1.0.

[43] C. Molnar, *Interpretable Machine Learning - A Guide for Making Black Box Models Explainable*, 2019.

[44] D. W. Apley and J. Zhu, "Visualizing the Effects of Predictor Variables in Black Box Supervised Learning Models," *arXiv:1612.08468 [stat]*, Aug. 2019.

[45] R Core Team, *R: A Language and Environment for Statistical Computing*, Vienna, Austria, 2019, 00000 R version 3.6.1.

[46] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16. New York, NY, USA: ACM, 2016, pp. 785–794, 00000.

[47] A. Karatzoglou, A. Smola, K. Hornik, and A. Zeileis, "Kernlab – An S4 Package for Kernel Methods in R," *Journal of Statistical Software*, vol. 11, no. 9, pp. 1–20, 2004, 00000 R package version 0.9-25.

[48] J. Friedman, T. Hastie, and R. Tibshirani, "Regularization paths for generalized linear models via coordinate descent," *Journal of Statistical Software*, vol. 33, no. 1, pp. 1–22, 2010, 00000.

[49] M. B. Kursa, *Praznik: Collection of Information-Based Feature Selection Filters*, 2018, 00000.

[50] Z. Zawadzki and M. Kosinski, *FSelectorRcpp: 'Rcpp' Implementation of 'FSelector' Entropy-Based Feature Selection Algorithms with a Sparse Matrix Support*, 2019, 00000.

[51] W. M. Landau, "The drake R package: A pipeline toolkit for reproducibility and high-performance computing," *Journal of Open Source Software*, vol. 3, no. 21, 2018.

[52] K. M. de Beurs and P. A. Townsend, "Estimating the effect of gypsy moth defoliation using MODIS," *Remote Sensing of Environment*, vol. 112, no. 10, pp. 3983–3990, Oct. 2008, 00000.

[53] R. Rengarajan and J. R. Schott, "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. International Society for Optics and Photonics, Sep. 2016, p. 997503, 00000.

[54] R. Meng, P. E. Dennison, F. Zhao, I. Shendryk, A. Rickert, R. P. Hanavan, B. D. Cook, and S. P. Serbin, "Mapping canopy defoliation by herbivorous insects at the individual tree level using bi-temporal airborne imaging spectroscopy and LiDAR measurements," *Remote Sensing of Environment*, vol. 215, pp. 170–183, Sep. 2018, 00000.

[55] U. Kälin, N. Lang, C. Hug, A. Gessler, and J. D. Wegner, "Defoliation estimation of forest trees from ground-level images," *Remote Sensing of Environment*, vol. 223, pp. 143–153, Mar. 2019, 00000.

[56] I. Shendryk, M. Broich, M. G. Tulbure, A. McGrath, D. Keith, and S. V. Alexandrov, "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*, vol. 187, pp. 202–217, Dec. 2016, 00000.

[57] M. Ludwig, T. Morgenthal, F. Detsch, T. P. Higginbottom, M. Lezama Valdes, T. Nauß, and H. Meyer, "Machine learning and multi-sensor based modelling of woody vegetation in the Molopo Area, South Africa," *Remote Sensing of Environment*, vol. 222, pp. 195–203, Mar. 2019, 00000.

[58] H. Meyer, C. Reudenbach, T. Hengl, M. Katurji, and T. Nauss, "Improving performance of spatio-temporal machine learning models using forward feature selection and target-oriented validation," *Environmental Modelling & Software*, vol. 101, pp. 1–9, Mar. 2018, 00000.

[59] Y. Guo, F.-L. Chung, G. Li, and L. Zhang, "Multi-Label Bioinformatics Data Classification With Ensemble Embedded Feature Selection," *IEEE Access*, vol. 7, pp. 103 863–103 875, 2019, conference Name: IEEE Access.

[60] M. Radovic, M. Ghalwash, N. Filipovic, and Z. Obradovic, "Minimum redundancy maximum relevance feature selection approach for temporal gene expression data," *BMC Bioinformatics*, vol. 18, no. 1, p. 9, Jan. 2017.

References