Random Forest: A Conceptual Primer

Evan Muise
email: evanmuise@gmail.com

2021-June-17

Abstract

Random Forest is a powerful ensemble machine learning algorithm suitable for both classification and regression problems. By combining bootstrap aggregation with decision trees and the random subspace method, Random Forest becomes a more accurate predictor than individual, or even bagged regression trees. In this overview, the major statistical concepts behind the algorithm are discussed, alongside assumptions and disgnostic procedures associated with the technique. A literature review is conducted concerning the use of the algorithm in various fields, including ecology, geography, and remote sensing. R packages and other software available to utilize the algorithm are presented, and a case study is conducted concerning the classification of broadleaf or coniferous trees using LiDAR data collected in 2015 at the UBC Vancouver Campus. The results found that vertical entropy and maximum height metrics derived from the point clouds are the most important predictors, however; low Cohen's kappa scores and classification accuracies lead to the authors recommending additional data to improve classification accuracy in future projects.

Contents

1 Introduction

Random Forest is a powerful algorithm developed by Breiman (2001) for both classification and regression. It is an incredibly robust machine learning algorithm which is commonly used in many fields of study, including remote sensing (Belgiu and Dragut, 2016; Chan and Paelinckx, 2008; Pal, 2005; Rodriguez-Galiano et al., 2012), ecological modelling (Cutler et al., 2007; De'ath, 2007), and economic geology (Rodriguez-Galiano et al., 2015), amongst many other fields. Random Forest is a combination of classification and regression trees (CART), and bagging (bootstrap aggregation), and is considered an ensemble method (Breiman, 2001). This paper will delve into the statistical background behind the components of the Random Forest algorithm, identify assumptions and diagnostic procedures, discuss examples of the algorithm's usage in peer-reviewed literature, identify available packages for implementing the algorithm, and finally, demonstrate an example usage of the algorithm.

2 Statistical Concepts

2.1 Classification and Regression Trees

Classification and regression trees (hereafter CART; also referred to as decision trees) are predictive models that predict output features by creating splits in various attributes in the dataset (Rokach and Maimon, 2007). These splits create nodes, labeled with input features (example shown in Figure 1). In the case of a continuous input variable, the node will be split based on being higher or lower than a value in the input variable (e.g. $zmax < 21.02$ at the first node in Figure 1) In the opposite case, where the input variable is a class (such as land cover), the node will be based on one value versus all others. CART models are scale invariant, can ignore irrelevant features, and are easily interpretable by the end user (Breiman, 2017). In addition, CART models can handle highly nonlinear and conditional relationships. However, not all is perfect with these classification trees. They often overfit the model leading to low bias and high variance. Due to this tendency to overfit, methods such as bagging and boosting are frequently employed to aid with these problems (Sutton, 2005).

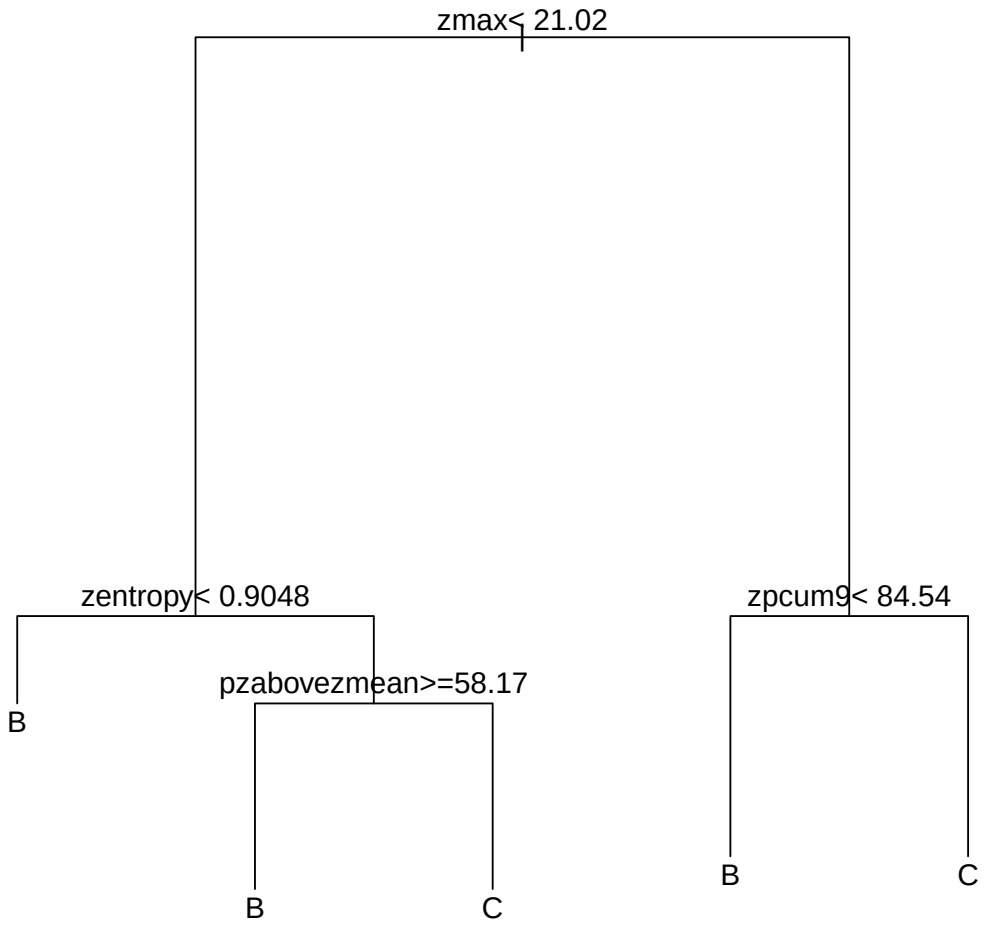

Figure 1: An example pruned classification tree with 7 nodes predicting broadleaf (B) or coniferous (C) class in trees at the UBC Vancouver Campus

2.2 Bagging (Bootstrap Aggregation)

Bootstrapping is a resampling method for calculating statistics on a dataset. The specific methodology is to resample with replacement in order to mimic the sampling process. This involves taking a dataset, and selecting the same number of observations, but allowing for the same observation to be reobserved. This allows users to derive estimates of variance and confidence intervals for a single dataset (Breiman, 1996).

Bagging, or bootstrap aggregation, occurs when a bootstrap resampled dataset is used to create a model

multiple times. The results from this **ensemble** of models is aggregated to generate a predictor from the many models (Breiman, 1996). This has previosuly been applied to CART models using all input features in the form of bagged regression trees (Sutton, 2005). Bagging does not reweight the input models to improve accuracy; a simple vote or average is used from all of the created models (Sutton, 2005). This ensemble method can have improved accuracy over a single CART model. However, while bagged models may have higher accuracy, the same predictors can dominate the models, reducing the potential maximum accuracy (Ho, 2002).

2.3 Random Forest

While bagged CART models can be a powerful method for classification and regression, with marked improvements over non-ensemble CART models, there is still room for improvement. When strong predictor features are present in the data, it is entirely possible for these few strong predictors to dominate the ensemble (Ho, 2002). Where Random Forest differs from bagged regression trees is in the use of the Random Subspace method, which uses a subset of all potential input features to train each tree. The Random Subspace method prevents these strong predicting features from dominating the resultant model (Tin Kam Ho, 1998).

In summary, the Random Forest algorithm creates many classification and regression tree models based off a bootstrap of the dataset **and** features (Breiman, 2001). These models are then aggregated into an ensemble model by averaging the model outputs (regression), or via votes (classification). This is a powerful advancement from simple CART models, which frequently overfit, and on bagged CART models, which may become dominated by strong predicting features (Breiman, 2001; Tin Kam Ho, 1998).

3 Assumptions and Diagnostic Procedures

3.1 Assumptions

One of the strengths of Random Forest is the algorithm's robustness. No formal distributions need to be followed, and the algorithm can handle both categorical and numerical data, which can be skewed or multi-modal. The algorithm's robustness leads to the Random Forest being incredibly powerful, and useful in many circumstances for both classification and regression.

3.2 Diagnostics

3.2.1 Out of Bag Error

The out of bag error is a validation method specifically applied to algorithms making using of the bagging approach outlined in Section 2.2. For each tree, the model is trained on those samples that are within the bootstrap, and then tested on those that were not included on the bootstrap for each tree. Those that are not included in the bootstrap are termed the "Out of Bag sample," and the error is calculated for each tree as the number of correctly predicted rows from the out of bag sample. It is recorded as each new tree

is generated and the ensemble predictions change. An example of how out of bag error changes as more trees are produced and added to the ensemble model can be found in Section 6 in Figure 3.

3.2.2 Variable Importance

Due to the high volume of variables potentially included in a Random Forest model, it can be relevant to the researcher to examine which variables contribute most to the model. This can be accomplished due to the recording of the Out of Bag error after each tree is generated. Each instantaneous (after each tree) error, can be compared to those found after the final trees are produced. This generates a score which can be used to rank the variables. Those with larger scores are considered more important than those with smaller scores (Zhu et al., 2015). It should be noted that there is a bias in Random Forest to favour categorical variables with high numbers of levels, however this can be overcome using other variants on the Random Forest algorithm (Altmann et al., 2010; Toloşi and Lengauer, 2011).

3.2.3 Cohen's Kappa

Cohen's kappa (k) is a measurement of categorical accuracy. It can be used with the Random Forest algorithm when used a classifier, but not as a regressor. k includes the possibility of chance agreement, rendering it a more robust measurement than percent agreement. It is calculated using the confusion matrix of a classification algorithm.

$$k = \frac{p_o - p_e}{1 - p_e} = 1 - \frac{1 - p_o}{1 - p_e} \tag{1}$$

The calculation for Cohen's Kappa is shown in Equation (1), where p_o is the observed agreement, and p_e is the hypothetical probability of chance agreement (Cohen, 1960). More detailed equations and calculations used in Cohen's kappa for nominal data of many classes can be found in Cohen (1960).

3.2.4 Receiver Operating Characteristic Area Under the Curve

The Receiver Operating Characteristic (ROC) is a diagnostic plot that is used to examine binary classifiers, such as the one used in Section 6 in Figure 4. The ROC plots the false positive rate against the true positive rate for a binary classifier. This shows the performance of the model at all classification thresholds.

The Area Under the Curve (AUC) for the ROC is another diagnostic included in this plot. Higher AUC values are desirable, with a perfect model having an AUC of 1.0. The AUC is scale-invariant, and measures prediction quality regardless of the classification threshold (Fawcett, 2006).

3.2.5 Regression Diagnostics

Diagnostic procedures for operating Random Forest as a regression algorithm are similar to other regression diagnostics. The R^2 is commonly used alongside various error statistics such as Root Mean Square Error, Mean Absolute Error, among others. These can be compared between model parameters or other types of regression.

4 Literature Review

Random Forest is frequently used as a classification algorithm in the geographical sciences (Belgiu and Dragut, 2016; Rodriguez-Galiano et al., 2012). It is often used in land-cover classifications (Belgiu and Dragut, 2016; Rodriguez-Galiano et al., 2012), due to the algorithm's robustness when supplied with highly dimensional data. In addition, the high processing speeds and ability to manage multi-modal data afforded by the algorithm led to its widespread adoption (Rodriguez-Galiano et al., 2012), alongside other algorithms such as Support Vector Machines or Artificial Neural Networks (Belgiu and Dragut, 2016). While boosting based ensemble methods can produce better accuracy results, they can be sensitive to outliers and overfit, in addition to requiring more computational resources than bagging based methods, such as Random Forest (Xu, 2014). These factors have led to Random Forest becoming a leading algorithm for remote sensing classification problems.

In ecology, Random Forest is frequently used for ecological modelling and prediction (De'ath, 2007), as well as classification (Cutler et al., 2007). Hollister et al. (2016) used Random Forest to compare input models (GIS and non-GIS based) for modelling lake trophic state across the continental United States. Their analysis is fully reproducible and is available on github. Cutler (2007) conducted a review on the usage of Random Forest for classification in ecology, and also gave examples on the method, using it to examine invasive plant species, rare lichen presence, and identify bird nesting sites. Prasad (2006) used Random Forest to predict vegetation maps under various climate scenarios, and found that Random Forest performed better when examining the Kappa statistics, correlation estimates, and spatial distribution of importance values. Authors are finding that Random Forest is a suitable method when examining classification and regression modelling problems in ecology (Cutler et al., 2007; De'ath, 2007).

In the geographic sciences, Random Forest is frequently used to map nutrients and water (Grimm et al., 2008; Naghibi et al., 2016; Rahmati et al., 2016). In Iran, groundwater potential has been examined by both Naghibi et al. (2016), and Rahmati et al. (2016) using Random Forest in recent years. Soil information, including soil organic carbon (2008) and soil class predictions Brungard et al. (2015) have also been studied using the alogrithm. These studies are not only using Random Forest for prediction and modelling, but are also using the algorithm to assess variable importance. Random Forest is commonly being used as a predictor for subsurface mapping in the soil sciences, and has been found to be a useful tool in this field as well (Brungard et al., 2015; Grimm et al., 2008; Naghibi et al., 2016; Rahmati et al., 2016).

5 Available Software Implementations

Due to Random Forest' prevalence as a classification and regression algorithm, it is frequently used and developed for new languages and software packages. While not an exhaustive list, included here is a starting point for running the algorithm in various programming languages and tools.

In R, there is the package **randomForest** (Breiman et al., 2018), as well as the implementation in **tidymodels** (Kuhn and Wickham, 2021). Other software and programming languages also have implementations of the Random Forest algorithm. Notably, **Scikit-learn** (Pedregosa et al., 2011) is a package devoted to machine learning in Python which includes the Random Forest ensemble algorithm. Other programming langauges such as MATLAB also include a Random Forest tool (MATLAB, n.d.). Random For-

est is implemented in the ArcGIS software as the Forest-based Classification and Regression tool (ESRI, n.d.).

6 Technique Application

6.1 Methods

6.1.1 Data

LiDAR data was collected over the University of British Columbia's Point Grey Campus in 2015 (University of British Columbia, 2015). The primary goals of this collection was to obtain an accurate digital terrain model of the Point Grey cliff face for geomorphologic modelling. A secondary goal was to generate elevation layers to examine buildings and trees on the campus for landscape and urban planning.

Between 2005 and 2010, 2937 trees were measured on the Point Grey campus. The information included alongside the majority of these measurements were genus, date updated, and location. Each genus was identified as broadleaf or coniferous for the Random Forest model to predict. This dataset is available from the UBC Faculty of Forestry teamshare drive, and is not publicly available. Building footprints were collected from the Vancouver Open Data portal (City of Vancouver, 2009).

6.1.2 Study Area

A study area of a single LiDAR tile on the UBC Point Grey Campus was used (Figure 2). This was done to reduce processing time. A total of 2408 trees were identified in the study area.

Figure 2: Location of the LiDAR tile on UBC Vancouver Campus.

6.1.3 Pre-processing

The single tile of the LiDAR data acquired over UBC was filtered for duplicate points, and normalized using functions within the lidR package (Roussel and Auty, 2021). These normalized points were used to create a canopy height model (CHM) using the pitfree algorithm (Khosravipour et al., 2014). This CHM was then masked for buildings to reduce the number of erroneous tree crowns created. Treetops were then delineated using the lmf algorithm inspired by Popescu et al. (2004). Trees were then segmented in the point cloud using the Dalponte (2016) algorithm for tree segmentation, with a minimum pixel height for trees of four. The segmented trees then had their crowns delineated, and standard lidR metrics were produced for each crown (see the lidR documentation for details). The tree dataset was then spatially joined to the nearest centroid of each crown, in order to create predictor variables for each tree measured on the UBC campus.

6.1.4 Analysis

The trees dataset was split into training and testing portions at 50% of the dataset each. Any tree without genus identification was removed from the dataset, as well as any trees with missing values in any variable. Intensity statistics created by the standard lidR metrics were removed, but all others were used as predictors. This include height statistics, quantiles, percentiles, and number of pulses returned. A total of 44 predictor variables were created.

A Random Forest classifier was created using 500 trees with two resample variables per tree. Importance values were retained. Confusion matrices were created for both the testing and training set, and used to calculate Cohen's kappa k. ROC curves were created for each of the potential classes, and the area under the curve was calculated for each. After each tree, out of bag and class error were calculated.

6.2 Results and Discussion

Figure 3 shows the error rates as trees are generated for each class, as well as the out of bag error. As additional trees are generated, error in broadleaf and out of bag categories is reduced. After approximately 100 trees, the error in broadleaf and out of bag categories stabilizes. Conversely, as the number of trees increases, coniferous tree error increases, until a much larger number of trees have been created. This could potentially be caused by the large amount of variation in tree canopy between species within each division (broadleaf and coniferous), and would need to be investigated further with additionally ancillary data to improve accuracy further.

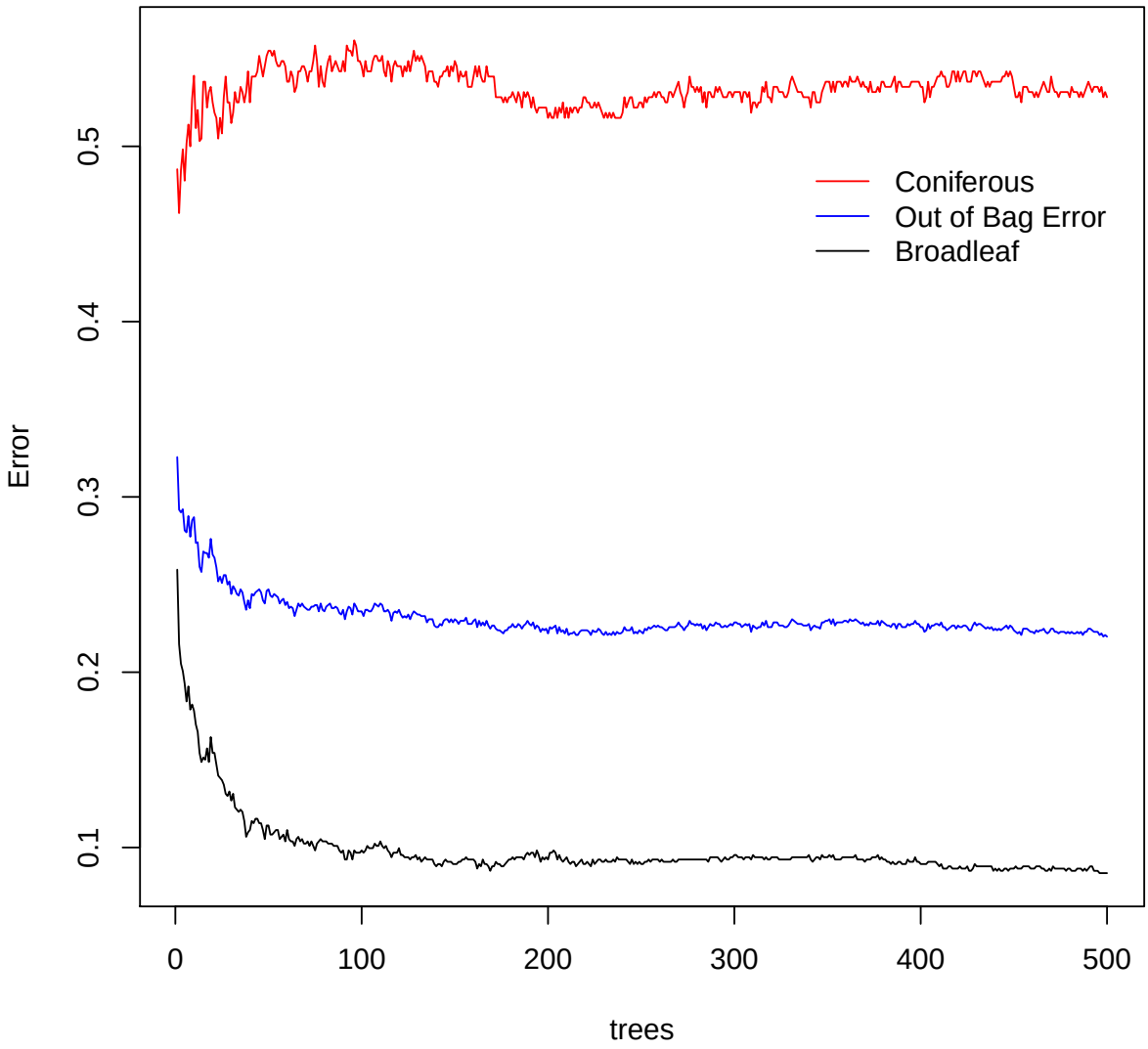

Figure 3: Error rates for coniferous, broadleaf, and out of bag samples as trees are generated in the Random Forest algorithm.

Confusion matrices were generated for both training and testing datasets after all trees were produced (Table 1). Cohen's kappa was calculated for both training and testing datasets, and was found to be 0.4818027 and 0.4265049, respectively. With a relatively low k in both training and testing, it is likely that additional data would need to be included to improve classification accuracy.

Table 1: Confusion matrices for testing (a) and training (b) datasets using the Random Forest classifier on LiDAR data on the UBC Vancouver Campus.

a	Broadleaf	Coniferous
Broadleaf	716	161
Coniferous	61	174

b	Broadleaf	Coniferous
Broadleaf	707	66
Coniferous	179	160

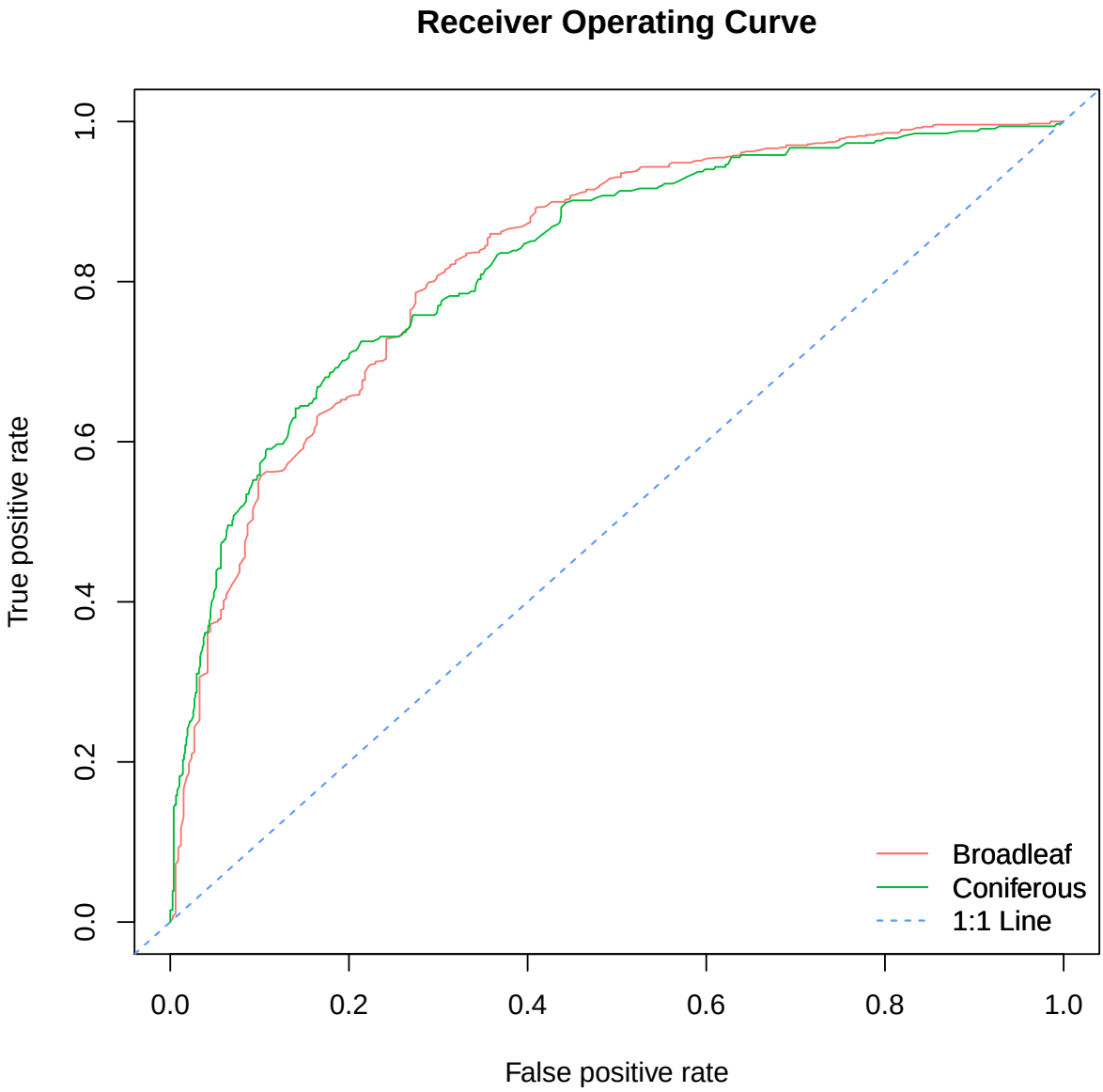

Figure 4: Receiver operating characteristic curve for each classification performed by the Random Forest algorithm on trees in the UBC Vancouver Campus.

The ROC curve generated for each class shows that the classifier is better than randomness for both broadleaf and coniferous classifications (Figure 4). Identical area under the curve values were found (0.8283832), as it is a binary classification. While the classifier is better than the random 1:1 line shown in Figure 4, the confusion matrices (Table 1 and kappa scores were not suitable to be highly accurate for this classification.

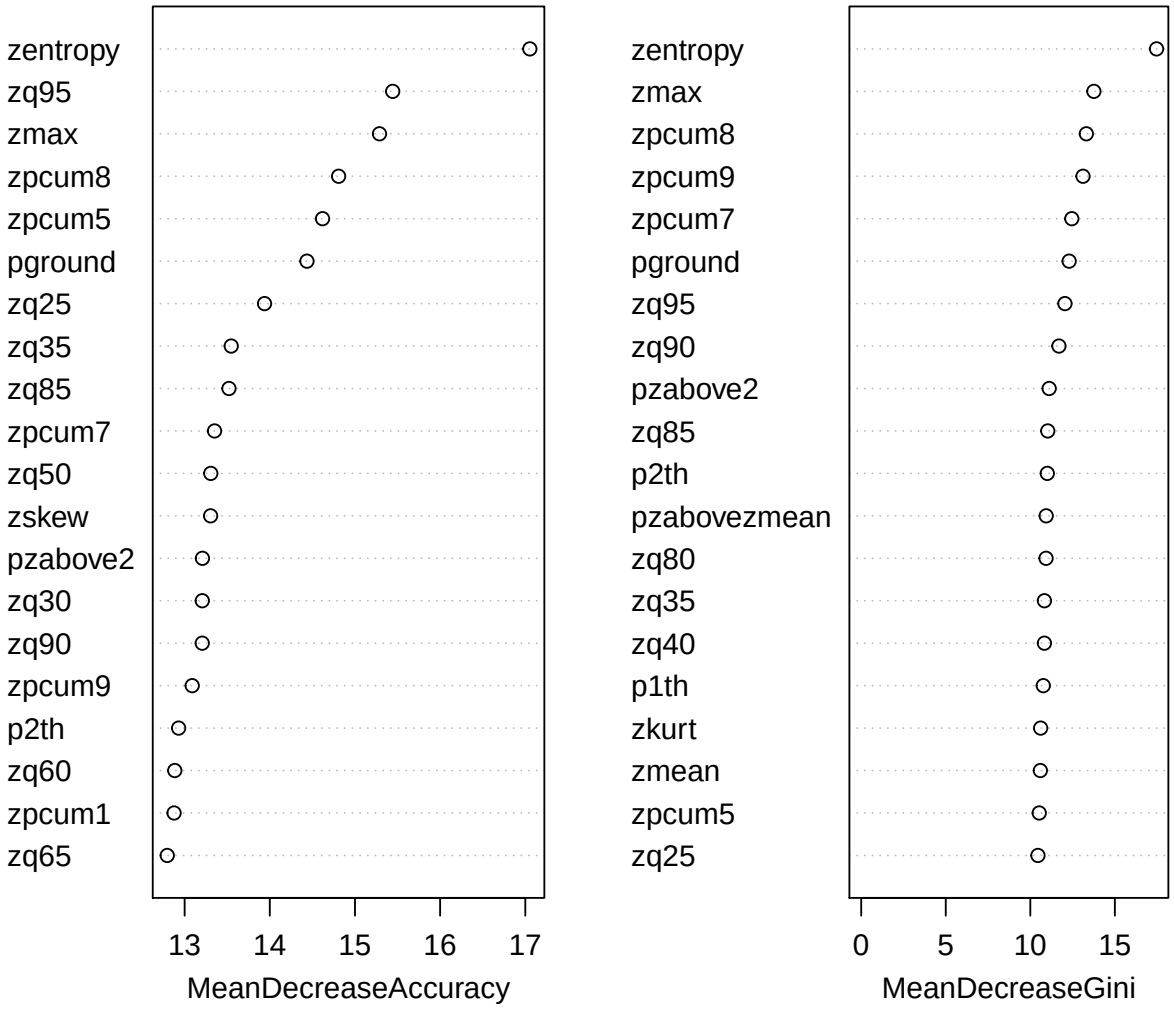

Figure 5: Variable important plot for classification of broadleaf or coniferous trees on the UBC Vancouver Campus.

Variable importance was calculated for each input variable, and the top 20 are shown in Figure 5. The highest importance variables are those associated with entropy and maximum height

($zentropy, zq95, zmax$, etc). As a metric of vertical diversity and evenness, entropy is a complex metric which is not easily interpretable. Other metrics related to height had high variable importance, and the Gini was decreased similarly with these metrics (Figure 5).

6.3 Conclusion

An introduction to the statistical basis of Random Forest was discussed. Important assumptions and diagnostic procedures were identified. A literature review was conducted concerning the usage of Random Forest in geographic and environmental sciences, with a focus on remote sensing, ecology, and geography. R packages and other software packages were identified, and a small case study was conducted.

The case study has shown a workflow for utilizing the lidR (Roussel and Auty, 2021) and randomForest (Breiman et al., 2018) R packages to conduct a species identification workflow. Classification diagnostics were assessed, including variable importance, out of bag error, Cohen's Kappa (k), ROC and AUC. Accuracy could be improved by including additional predictors, such as multispectal LiDAR (Budei et al., 2018), geographic ancilliary variables such as slope, elevation, aspect, or soil data (Hollister et al., 2016). Ways to improve the accuracy and validation of the method in this context could involve looking for spatial autocorrelation in the accuracy results, or applying k-fold cross validation to the model.

Acknowledgements

The following packages were used in the production of this document: **base** (R Core Team, 2021), **lidR** (Roussel and Auty, 2021), **raster** (Hijmans, 2020), **sp** (Pebesma and Bivand, 2021), **ggmap** (Kahle et al., 2019), **ggspatial** (Dunnington, 2021), **randomForest** (Breiman et al., 2018), **rpart** (Therneau and Atkinson, 2019), **sf** (Pebesma, 2021), **tidyverse** (Wickham, 2021), **knitr** (Xie, 2021a), **bookdown** (Xie, 2021b), and **here** (Müller, 2020).

In addition, I would like to thank Dr. Moore for his help throughout GEOB503, and for the excellent primer on the bookdown package.

References

Altmann, A., Toloşi, L., Sander, O., Lengauer, T., 2010. Permutation importance: A corrected feature importance measure. Bioinformatics 26, 1340–1347. https://doi.org/10.1093/bioinformatics/btq134

Belgiu, M., Dragut, L., 2016. Random forest in remote sensing: A review of applications and future directions. Isprs Journal of Photogrammetry and Remote Sensing 114, 24–31. https://doi.org/10.1016/j.isprsjprs.2016.01.011

Breiman, L., 2017. Classification and Regression Trees. Routledge.

Breiman, L., 2001. Random Forests. Machine Learning 45, 5–32. https://doi.org/10.1023/A:1010933404324

Breiman, L., 1996. Bagging predictors. Machine Learning 24, 123–140. https://doi.org/10.1007/BF00058655

Breiman, L., Cutler, A., Liaw, A., Wiener, M., 2018. randomForest: Breiman and cutler's random forests for classification and regression.

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

Budei, B.C., St-Onge, B., Hopkinson, C., Audet, F.-A., 2018. Identifying the genus or species of individual trees using a three-wavelength airborne lidar system. Remote Sensing of Environment 204, 632–647. https://doi.org/10.1016/j.rse.2017.09.037

Chan, J.C.-W., Paelinckx, D., 2008. Evaluation of Random Forest and Adaboost tree-based ensemble classification and spectral band selection for ecotope mapping using airborne hyperspectral imagery. Remote Sensing of Environment 112, 2999–3011. https://doi.org/10.1016/j.rse.2008.02.011

City of Vancouver, 2009. Building footprints 2009.

Cohen, J., 1960. A Coefficient of Agreement for Nominal Scales. Educational and Psychological Measurement 20, 37–46. https://doi.org/10.1177/001316446002000104

Cutler, D.R., Edwards, T.C., Beard, K.H., Cutler, A., Hess, K.T., Gibson, J., Lawler, J.J., 2007. Random Forests for Classification in Ecology. Ecology 88, 2783–2792. https://doi.org/https://doi.org/10.1890/07-0539.1

Dalponte, M., Coomes, D.A., 2016. Tree-centric mapping of forest carbon density from airborne laser scanning and hyperspectral data. Methods in Ecology and Evolution 7, 1236–1245. https://doi.org/10.1111/2041-210X.12575

De'ath, G., 2007. Boosted Trees for Ecological Modeling and Prediction. Ecology 88, 243–251. https://doi.org/https://doi.org/10.1890/0012-9658(2007)88%5B243:BTFEMA%5D2.0.CO;2

Dunnington, D., 2021. Ggspatial: Spatial data framework for ggplot2.

ESRI, n.d. Forest-based classification and regression (spatial statistics).

Fawcett, T., 2006. An introduction to ROC analysis. Pattern Recognition Letters 27, 861–874. https://doi.org/10.1016/j.patrec.2005.10.010

Grimm, R., Behrens, T., Märker, M., Elsenbeer, H., 2008. Soil organic carbon concentrations and stocks on Barro Colorado Island Digital soil mapping using Random Forests analysis. Geoderma 146, 102–113. https://doi.org/10.1016/j.geoderma.2008.05.008

Hijmans, R.J., 2020. Raster: Geographic data analysis and modeling.

Ho, T.K., 2002. A Data Complexity Analysis of Comparative Advantages of Decision Forest Constructors. Pattern Analysis & Applications 5, 102–112. https://doi.org/10.1007/s100440200009

Hollister, J.W., Milstead, W.B., Kreakie, B.J., 2016. Modeling lake trophic state: a random forest approach. Ecosphere 7, e01321. https://doi.org/https://doi.org/10.1002/ecs2.1321

Kahle, D., Wickham, H., Jackson, S., 2019. Ggmap: Spatial visualization with ggplot2.

Khosravipour, A., Skidmore, A.K., Isenburg, M., Wang, T., Hussin, Y.A., 2014. Generating pit-free canopy height models from airborne lidar. Photogrammetric Engineering & Remote Sensing 80, 863–872. https://doi.org/10.14358/PERS.80.9.863

Kuhn, M., Wickham, H., 2021. Tidymodels: Easily install and load the tidymodels packages.

MATLAB, n.d. Create bag of decision trees.

Müller, K., 2020. Here: A simpler way to find your files.

Naghibi, S.A., Pourghasemi, H.R., Dixon, B., 2016. GIS-based groundwater potential mapping using boosted regression tree, classification and regression tree, and random forest machine learning models in iran. Environmental Monitoring and Assessment 188, 44. https://doi.org/10.1007/s10661-015-5049-6

Pal, M., 2005. Random forest classifier for remote sensing classification. International Journal of Remote Sensing 26, 217–222. https://doi.org/10.1080/01431160412331269698

Pebesma, E., 2021. Sf: Simple features for r.

Pebesma, E., Bivand, R., 2021. Sp: Classes and methods for spatial data.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E., 2011. Scikit-learn: Machine learning in python. Journal of Machine Learning Research 12, 28252830.

Popescu, S.C., Wynne, R.H., 2004. Seeing the trees in the forest. Photogrammetric Engineering & Remote Sensing 70, 589–604. https://doi.org/10.14358/PERS.70.5.589

Prasad, A.M., Iverson, L.R., Liaw, A., 2006. Newer Classification and Regression Tree Techniques: Bagging and Random Forests for Ecological Prediction. Ecosystems 9, 181–199. https://doi.org/10.1007/s10021-005-0054-1

R Core Team, 2021. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria.

Rahmati, O., Pourghasemi, H.R., Melesse, A.M., 2016. Application of GIS-based data driven random forest and maximum entropy models for groundwater potential mapping: A case study at mehran region, iran. Catena 137, 360–372. https://doi.org/10.1016/j.catena.2015.10.010

Rodriguez-Galiano, V.F., Ghimire, B., Rogan, J., Chica-Olmo, M., Rigol-Sanchez, J.P., 2012. An assessment of the effectiveness of a random forest classifier for land-cover classification. ISPRS Journal of Photogrammetry and Remote Sensing 67, 93–104. https://doi.org/10.1016/j.isprsjprs.2011.11.002

Rodriguez-Galiano, V., Sanchez-Castillo, M., Chica-Olmo, M., Chica-Rivas, M., 2015. Machine learning predictive models for mineral prospectivity: An evaluation of neural networks, random forest, regression trees and support vector machines. Ore Geology Reviews 71, 804–818. https://doi.org/10.1016/j.oregeorev.2015.01.001

Rokach, L., Maimon, O.Z., 2007. Data Mining With Decision Trees: Theory And Applications. World Scientific.

Roussel, J.-R., Auty, D., 2021. lidR: Airborne LiDAR data manipulation and visualization for forestry applications.

Sutton, C.D., 2005. Classification and Regression Trees, Bagging, and Boosting. Elsevier, pp. 303–329. https://doi.org/10.1016/S0169-7161(04)24011-1

Therneau, T., Atkinson, B., 2019. Rpart: Recursive partitioning and regression trees.

Tin Kam Ho, 1998. The random subspace method for constructing decision forests. IEEE Transactions on Pattern Analysis and Machine Intelligence 20, 832–844. https://doi.org/10.1109/34.709601

Toloşi, L., Lengauer, T., 2011. Classification with correlated features: Unreliability of feature ranking and solutions. Bioinformatics 27, 1986–1994. https://doi.org/10.1093/bioinformatics/btr300

University of British Columbia, 2015. University of british columbia point grey campus LiDAR, 2015. https://doi.org/11272.1/AB2/KET75X

Wickham, H., 2021. Tidyverse: Easily install and load the tidyverse.

Xie, Y., 2021b. Bookdown: Authoring books and technical documents with r markdown.

Xie, Y., 2021a. Knitr: A general-purpose package for dynamic report generation in r.

Xu, L., 2014. A comparative study of different classification techniques for marine oil spill identification using RADARSAT-1 imagery. Remote Sensing of Environment 10.

Zhu, R., Zeng, D., Kosorok, M.R., 2015. Reinforcement Learning Trees. Journal of the American Statistical Association 110, 1770–1784. https://doi.org/10.1080/01621459.2015.1036994