Application of Statistical Learning Methods to Geographically Weighted, Spectral Features for Forest Tree Species Classification

Aristizabal, Fernando[1]
[1]Department of Agricultural and Biological Engineering, Gainesville, FL

(Submitted 09 December 2016)

Remotely-sensed, geographically weighted data for a tract of forested land was created from ASTER imagery by B. Johnson et al to classify the pixels into one of four land covers. The study used a support vector machine with a radial basis kernel to classify the pixels and achieved a test accuracy of 85.9%. The purpose of this document is to examine additional statistical learning methods for classifying this dataset with the intention improving the generalized performance of the originating study.

I. INTRODUCTION

Land cover classification, which is a description of what tangible material is covering the Earth's surface, is of enormous value to society. Land cover is different and should not be confused with land utilization, since a single physical material covering land can be used to describe multiple utilizations and vice versa. High-level land cover classifications could include asphalt, bare ground, forest, and water, but a wide variety of classification combinations to suit diverse applications could be generated. In addition to high-level land cover categories, a plethora of sub-categories organized in hierarchies could be leveraged to further describe land cover (A. Comber et al, 2005).

Two main methods, field survey and remote sensing, exist for gathering land cover information. Over more recent years, major societal investments have taken place to move land cover data gathering to a remote sensing platforms especially satellite-

mounted systems such as the National Aeronautical and Space Administration's (NASA) Moderate-Resolution Imaging Spectroradiometer (MODIS) flown on board satellite's Terra and Aqua. Space-based systems present numerous advantages to ground or aviation based equipment in that they can provide near global spatial coverage at high temporal (3 to 1 days) and spatial resolutions (1 km to 15 meters) (ATBD Modis Land Cover). NASA currently publishes several data products that leverage this imagery to classify land cover in 17 categories on a near global scale (A. Strahler, 1999). These collected advances in remotely sensed, land cover classification have empowered research across a wide spectrum of disciplines by providing an additional dataset to explore and correlate with.

The subject of this document is concerned with exploring a variety of statistical learning methods such as discriminant analysis, support vector machines, and decision trees to classify four tree types in a small forested region in Japan. The Advanced Spaceborne Thermal Emission and Reflection Radiometer (ASTER) collected-data was processed by the authors of the paper *Using Geographically Weighted Variables for Image Classification* by leveraging inverse distance weighting (IDW) interpolation (B. Johnson et al, 2012). The authors then utilized a support vector machine (SVM) with a radial basis kernel (RBF) to classify the spectral and weighted variables to various forest types. The goal of this document is to detail an exploration of eight total statistical/machine learning algorithms including linear discriminant analysis, quadratic discriminant analysis, k-nearest neighbor, SVM with linear kernel, SVM-RBF, decision trees, boosting, and random forests to better the authors best classification accuracy obtained with SVM-RBF of 85.9%.

II. BACKGROUND AND DATA

The data collected for the study by B.Johnson et al. originates from the ASTER instrument on board the Terra satellite launched in 1999. The instrument provides very high resolution imaging for 14 electromagnetic bands ranging from visible to thermal infrared light. The researchers were interested in tree land cover categories specifically in a 13 by 12 km tract of forest in the Ibaraki Prefecture of Japan. The authors were interested in classifying the rectangular batch of forest into four categories including the two species *Cryptomeria japonica* (Sugi, or Japanese Cedar) and *Chamaecyparis obtusa* (Hinoki, or Japanese Cypress). The two additional categories of interest included mixed deciduous broadleaf natural forest and a small portion of land that included other land covers. Table 1 summarizes the four classes of interest as well their factor labels and factor numbers that are used to represent each class in the data and results. The purpose of selecting this region and land cover classes is motivated by determining the influence of planted broad-leaf forest on bird populations during diverse conditions (Y. Yamaura et al.).

Table 1: Summary of the four land cover classifications utilized by B. Johnson et al. And the corresponding class labels.

Class Description	Factor Label	Factor Number
Mixed deciduous broad-leaf forest	d	1
Chamaecyparis obtusa (Hinoki, or Japanese Cypress)	h	2
Other land class/use (agriculture, roads, buildings,etc.)	o	3
Cryptomeria japonica (Sugi, or Japanese Cedar)	s	4

The original ASTER data underwent a series of pre-processing methods in order to prepare for classification which included rectifying and applying expert labels utilizing sparse training polygons. The novelty of the authors' approach involved applying spatial interpolation by IDW in order to yield geographical weighted features and feature labels that place greater weight on label polygons and features that are closer to a given pixel. IDW seeks to create a continuous interpolated hyper-surface from discontinuous data by taking a weighted average of the inverse distance to the p power (O'Sullivan and Unwin 2003). By applying this technique to the first nine spectral bands for both Sugi and Hinoki, the authors generalized the training polygon labels to the entire dataset and generated a set of additional features. Specifically, these 18 additional features were generated by subtracting the original bands value by the interpolated values from both Sugi and Hinoki. The resulting data yields a total of 27 features and one class label with 198 training patterns and 325 testing patterns. That is approximately 37.9% of the data that is used for training and cross-validating while the remaining 62.1% is left for testing.

Now that the data matrices are defined for both training and test sets, the authors leveraged a SVM with a radial-basis kernel (RBF) to train and predict the classification labels on the test set. Only two parameters were selected for additional tuning, including cost and gamma which controls the kernel spread. Wide sweeps were conducted for each variable including 1.25^{-2} to 1.25^{31} for cost and 1.25^{-30} to 1.25^{25} for gamma and the optimal values were concluded to be 1010 and 0.003 for cost and gamma, respectively. In addition to utilizing all 27 features, for the purpose their study the authors also predicted with only the original 9 feature bands to demonstrate the value of their IDW

methodology. Predicting on the test set with all 27 features yielded a classification accuracy of 85.9% while predicting with only 9 features yielded an accuracy of 82.2%

III. METHODOLOGY

Due to the value in monitoring this region for bird habitats, further advances in classification accuracy could be achieved by trying a further array of statistical learning methods as well as a broader range of parameters for each method. This study seeks to make a marginal improvement on the classification accuracy of 85.9% achieved by B. Johnson et al. utilizing all 27 features for training.

A total of eight statistical learning algorithms were trained, tuned, and predicted on including linear discriminant analysis, quadratic discriminant analysis, k-nearest neighbor, SVM with linear kernel, SVM-RBF, decision trees, boosting, and random forests. All experiments were conducted within the R Project for Statistical Computing environment with the assistance of several packages freely available from the Comprehensive R Archive Network (CRAN) including *tree* (decision trees), *randomForest* (random forests), *MASS* (discriminant analysis), *class* (k-nearest neigbor), *e1071* (SVM), *gbm* (generalized boosting models), *and parallel* (parallel computing). A brief conceptual introduction to all the models utilized will be provided, but for more detailed information please review *Introduction to Statistical Learning*.

LDA is a classification technique that can be useful when the response variable involves more than two response classes. LDA makes the assumption that the predictors follow a one-dimensional normal distribution with some correlation between each pair of predictors. LDA decision boundaries are linear by definition and decided upon by maximizing the inter-class variances and minimizing the intra-class variances for each pair of classes. Building on the LDA concept, quadratic discriminant analysis (QDA) also

assumes Gaussian distribution, but does not assume a common covariance matrix for each class contrary to LDA. This does increase computational cost by adding another parameter for every predictor. Comparing LDA and QDA, a bias/variance tradeoff exists since LDA is less flexible than QDA and has lower variance. However, QDA can outperform LDA if the common covariance assumption proves false.

K-nearest neighbors (KNN) is a simple model that seeks to classify testing patterns by finding the k-nearest neighbors by measure of Euclidean distance. The pattern is classified by means of a majority vote, specifically classifying the testing pattern to the class that is most represented in the subset of k-nearest neighbors. K is a tuning parameter that was selected with 10-fold cross validation on the training data.

The support vector machine (SVM) is a linear classifier that seeks to determine a hyperplane decision boundary that maximizes the inter-class soft margin. Several parameters can be tuned for a linear SVM including cost that seeks to add a penalty to the slack variables. Other SVM tuning parameters include tolerance which represents an optimization termination criteria and epsilon which is the insensitive-loss function. Non-linear decision boundaries can be determined by what is known as the kernel trick that projects p-dimensional variables to an infinite dimensional space. A variety of functions can be leveraged as a kernel, but a radial basis kernel was selected for this experiment that is detailed in equation (1). Note the addition of another tuning parameter, gamma.

$$K(x_i, x_{i'}) = \exp\left(-\gamma \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2\right) \quad (1)$$

Decision trees perform classification by binary recursive partition which generates two branches of the tree by selecting a partition of one of the predictors that

minimizes the misclassification error. Trees do need to be pruned to remove terminal nodes that do not differentiate between two separate classes which was conducted with 10-fold cross validation.

Boosting decision trees is a process that seeks to reduce bias and variance by fitting large number of trees by fitting successive trees to the residuals of previous trees. Three main tuning parameters can be leveraged to improve performance including the number of trees, shrinkage parameter, and number of splits. The number of trees refers to the number of decision trees to use in the boosting approach. The shrinkage parameter is a learning or optimization rate, while the number of splits or interaction depth refers to the number of splits each decision tree should have. Training and cross validation boosted trees is very computationally expensive process though it can be parallelized which was successfully executed by the *parallel* package in the CRAN library.

Random forests seeks to build a decision tree structure by randomly sampling m predictors from the full set of p predictors. Tuning parameters for random forests include m or the number of predictors to be randomly sampled, node size which is the minimum number of terminal nodes, and number of trees.

All parameters for all models are selected using a grid search approach over a range that usually spans the default values set by the function. Every combination of parameters is tested using k-fold cross validation with k set to 10. K-fold cross validation partitions the training data patterns into k evenly sized folds. Training is executed on every combination of k-1 folds and validate on the left out fold. The validation accuracy is calculated and averaged for all k fold combinations. The parameter combination with the best cross validation performance is selected for training on the entire training set

and predicted on the test set to determine the generalized prediction accuracy. While this approach is taken to select the best combination of parameters for every model, some of the figures were generated by varying one parameter and keeping the other parameters to their default values. This was done in order to examine the relative importance of each parameter and the relationship with the training set classification accuracy.

IV. RESULTS AND ANALYSIS

Overall, the eight statistical learning methods selected performed well on the test set as all except for QDA performed with greater than or equal to 80% accuracy. QDA performed poorly with 65.8% accuracy, while 1-nearest neighbor performed the best with 86.5% test set accuracy. This model successfully out performed B. Johnson et al best classification accuracy of 85.9% by 0.6%. Below is a summary of all the models selected with their corresponding best parameters and test set accuracy performance.

Table 2: Summary of all models with corresponding test accuracies and parameters selected.

Model	Test Accuracy	Parameters
KNN	84.6%	k = 1
SVM-Radial	85.8%	cost=17.8 | tolerance=1e-5 | epsilon=0.1 | gamma=0.01
SVM-Linear	85.5%	cost=0.269 | tolerance=1e-5 | epsilon=0.1
LDA	84.6%	N/A
Boosting	80.6%	shrinkage=0.398 | number of trees=500 | depth=1
Random Forests	80.3%	mtry=13 | node size=5 | number of trees=50
Trees	80.0%	size=4
QDA	65.8%	N/A

As previously mentioned, the best parameters were selected by an extensive parameter grid search, however parameters were individually tuned while keeping other parameters set at default values in order to generate figures and to examine how

performance varies with different parameter values. A full array of figures is presented in the appendix section.

V. DISCUSSION AND CONCLUSION

The underlying goal of this exercise was to seek a better alternative to the radial-SVM presented by B. Johnson et al for classifying geometrically weighted, spectral features of a forested area in Japan. This exercise was successfully able to out perform the 85.9% accuracy mark by 0.6% through the utilization of 1-nearest neighbors. Several other classifiers were leveraged with good results but were not able to beat their implementation of the radial-SVM.

VI. REFERENCES

Comber, Alexis, Peter Fisher, and Richard Wadsworth. "What Is Land Cover?" *Environment and Planning B* (2005): 1-15. Web. 6 Dec. 2016.

Strahler, Alan, Doug Muchoney, Jordan Borak, Mark Friedl, Sucharita Gopal, Eric Lambin, and Aaron Moody. "MODIS Land Cover Product." *Algorithm Theoretical Basis Document (ATBD)* 5 (1999): 1-72. Web. 6 Dec. 2016.

Johnson, Brian Alan, and Zhixiao Xie. "Using Geographically Weighted Variables for Image Classifcation." *Remote Sensing Letters* (2012): 1-9. Web. 6 Dec. 2016.

Yamaura, Yuichi, Susumu Ikeno, Makoto Sano, and Kenichi Ozaki. "Bird Response to Broad-Leaved Forest Patch Area in Plantation Landscape Across Seasons." *Biological Conservation* 2155-2165 142.10 (2009): Web. 6 Dec. 2016.

O'sullivan, D. and Unwin, D. 2003. *Geographic information analysis*. Hoboken, N.J.: Wiley.

R Core Team (2016). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna,

Austria. URL https://www.R-project.org/.

Brian Ripley (2016). tree: Classification and Regression Trees. R package version 1.0-37. https://CRAN.R-project.org/package=tree

A. Liaw and M. Wiener (2002). Classification and Regression by randomForest. R News 2(3), 18--22.

Venables, W. N. & Ripley, B. D. (2002) Modern Applied Statistics with S. Fourth Edition. Springer, New York. ISBN 0-387-95457-0

Venables, W. N. & Ripley, B. D. (2002) Modern Applied Statistics with S. Fourth Edition. Springer, New York. ISBN 0-387-95457-0

David Meyer, Evgenia Dimitriadou, Kurt Hornik, Andreas Weingessel and Friedrich Leisch (2015). e1071: Misc Functions of the Department of Statistics, Probability Theory Group (Formerly: E1071), TU Wien. R package version 1.6-7. https://CRAN.R-project.org/package=e1071

Greg Ridgeway with contributions from others (2015). gbm: Generalized Boosted Regression Models. R package version 2.1.1. https://CRAN.R-project.org/package=gbm

R Core Team (2016). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

James, Gareth, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An Introduction to Statistical Learning*. New York: Springer Science, 2015. Print.

VII. APPENDIX

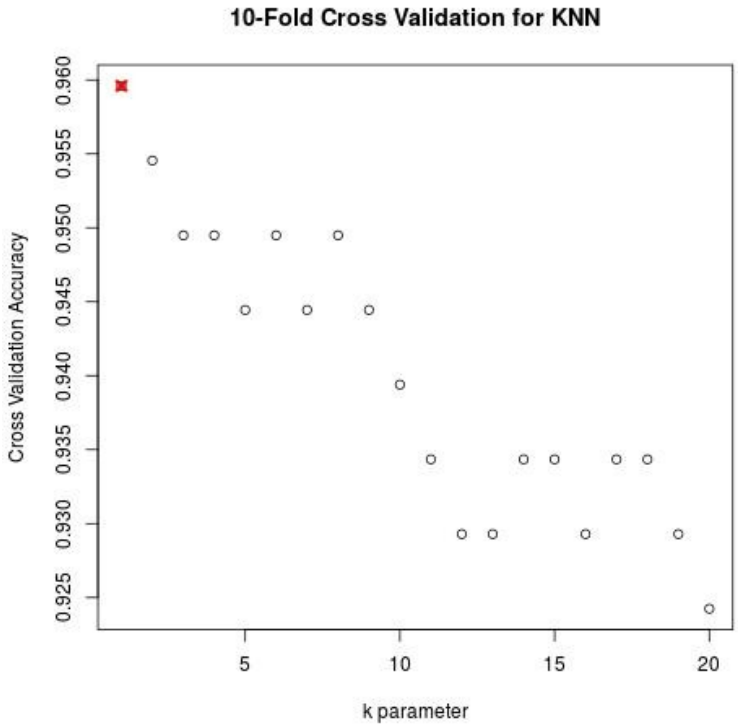

Figure 1: Tuning of k parameter in k-nearest neighbor.

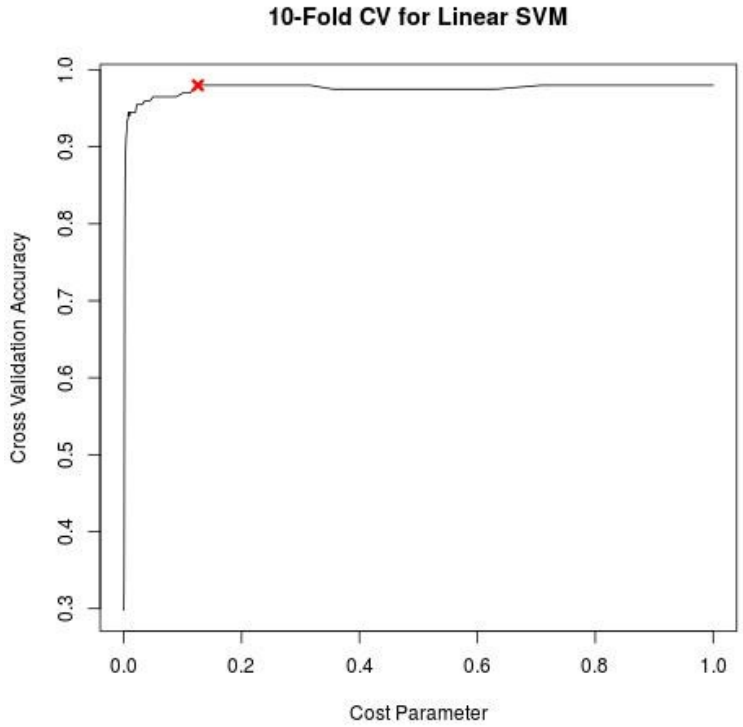

Figure 2: Tuning cost parameter for linear SVM

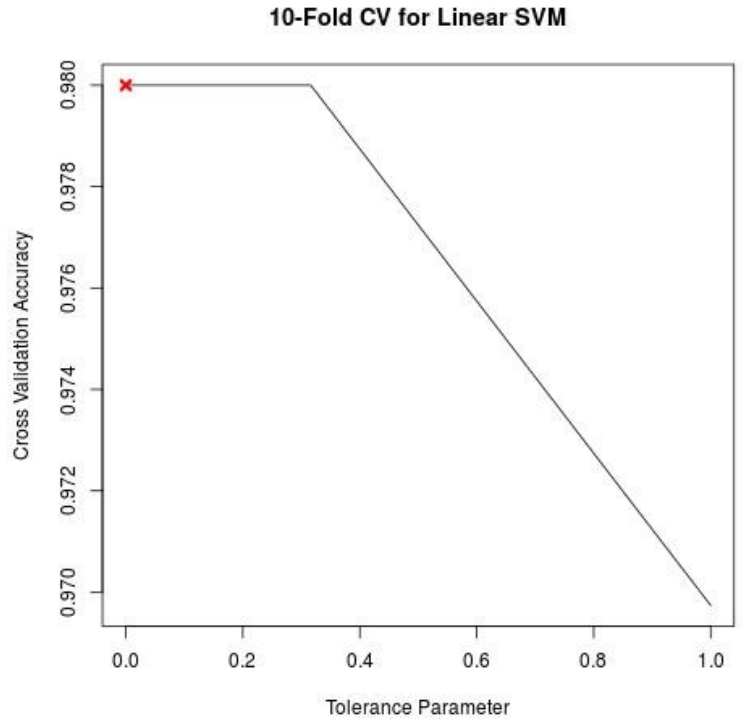

Figure 3: Tuning of tolerance parameter for linear SVM

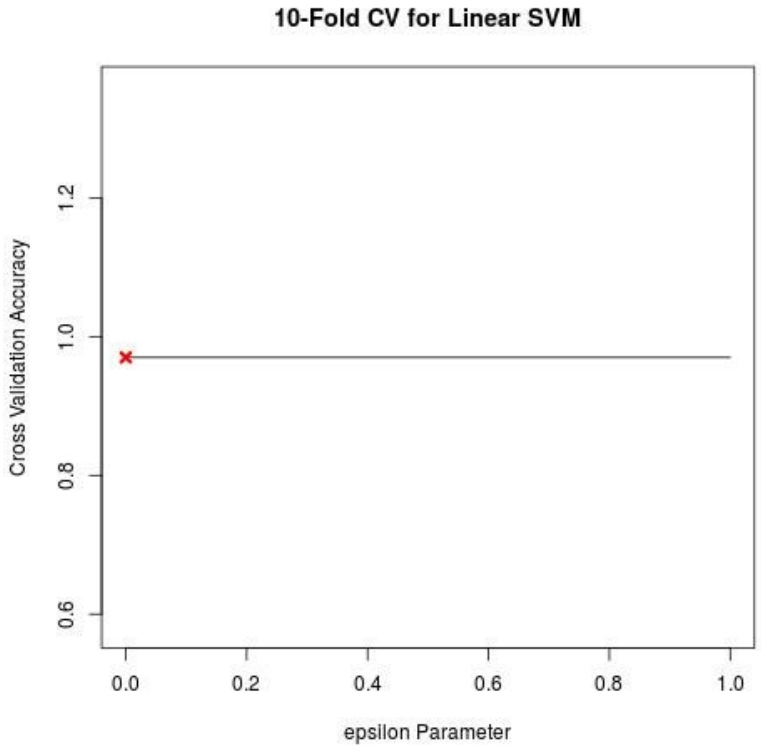

Figure 4: Tuning of epsilon parameter in linear SVM

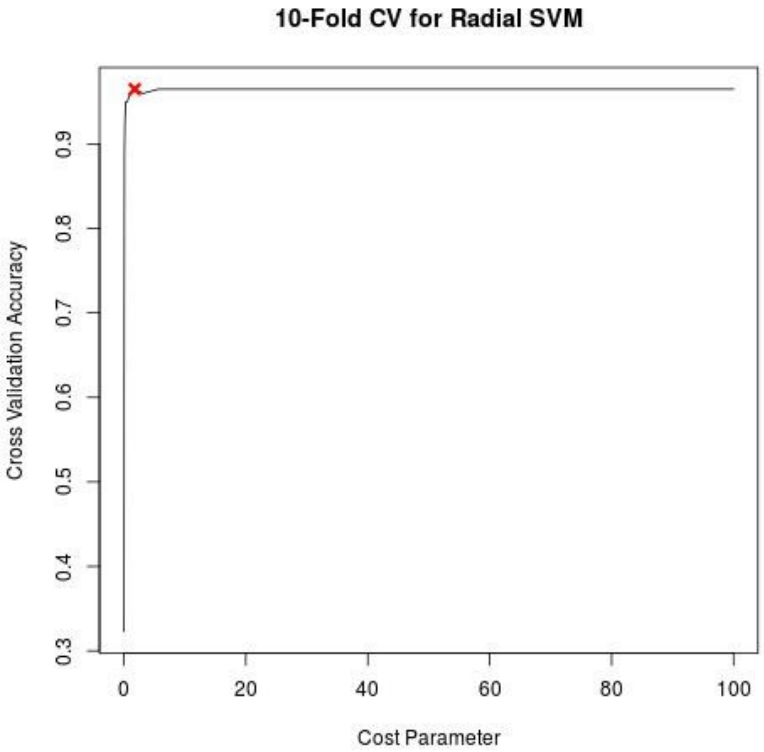

Figure 5: Tuning cost parameter in radial SVM

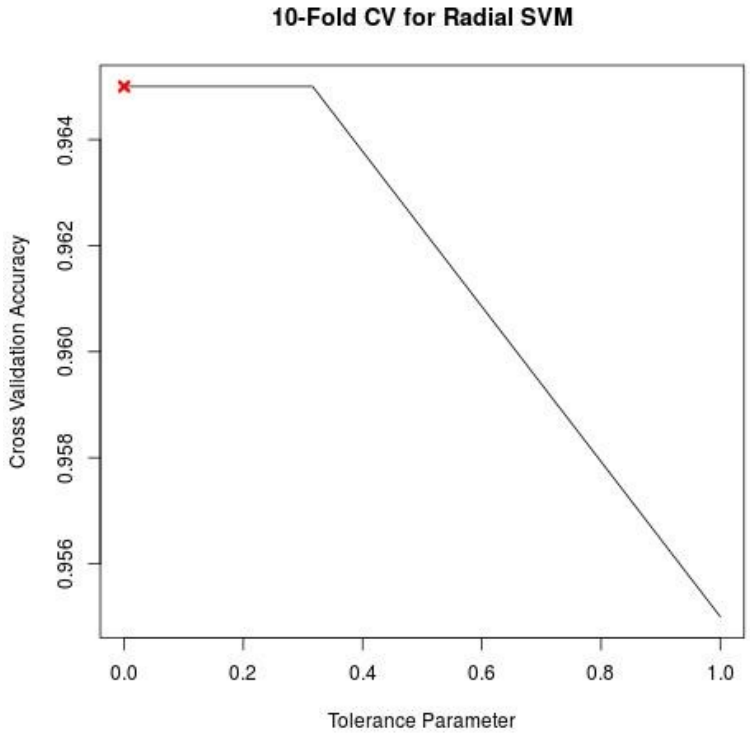

Figure 6: Tuning of tolerance in radial SVM

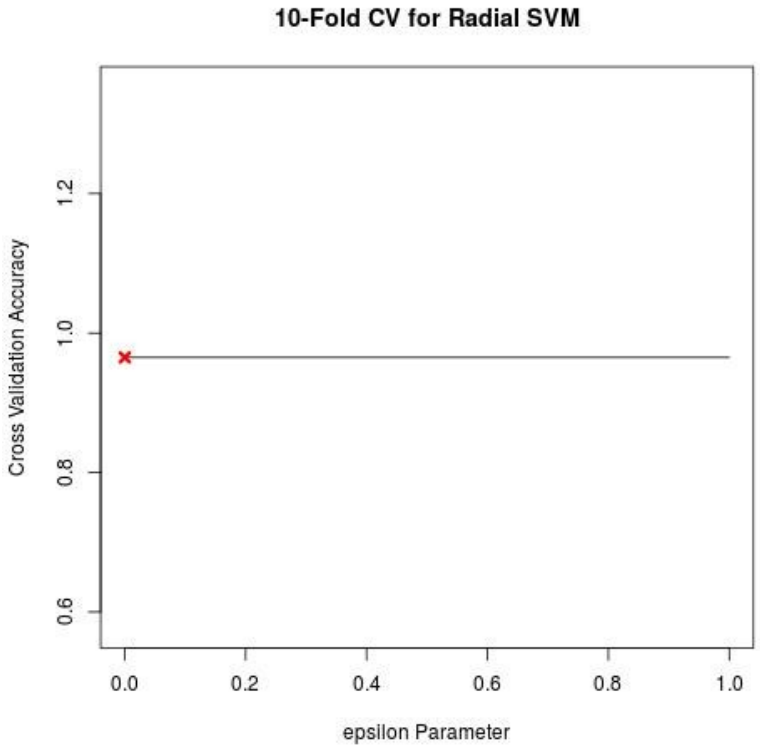

Figure 7: Tuning of epsilon parameter in radial SVM

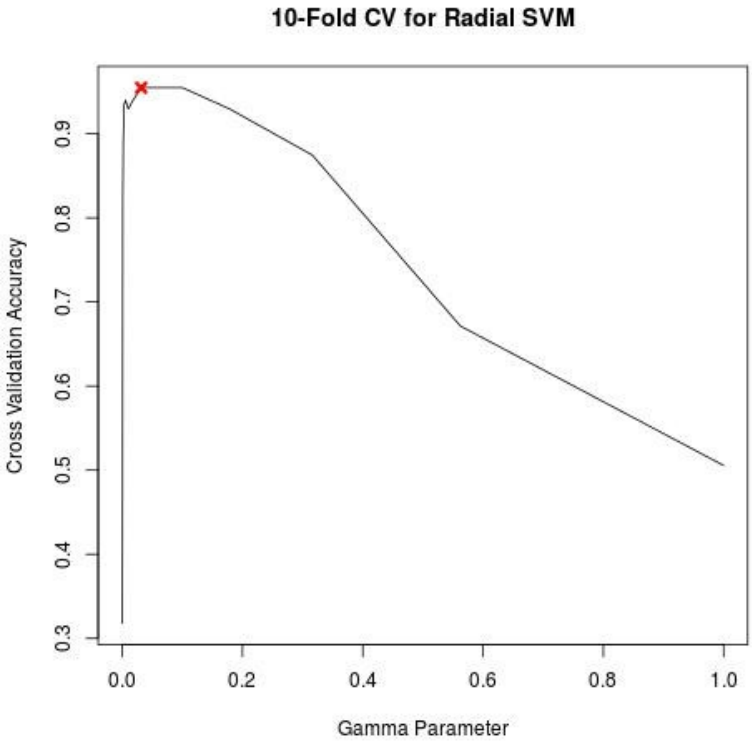

Figure 8: Tuning gamma parameter for radial SVM

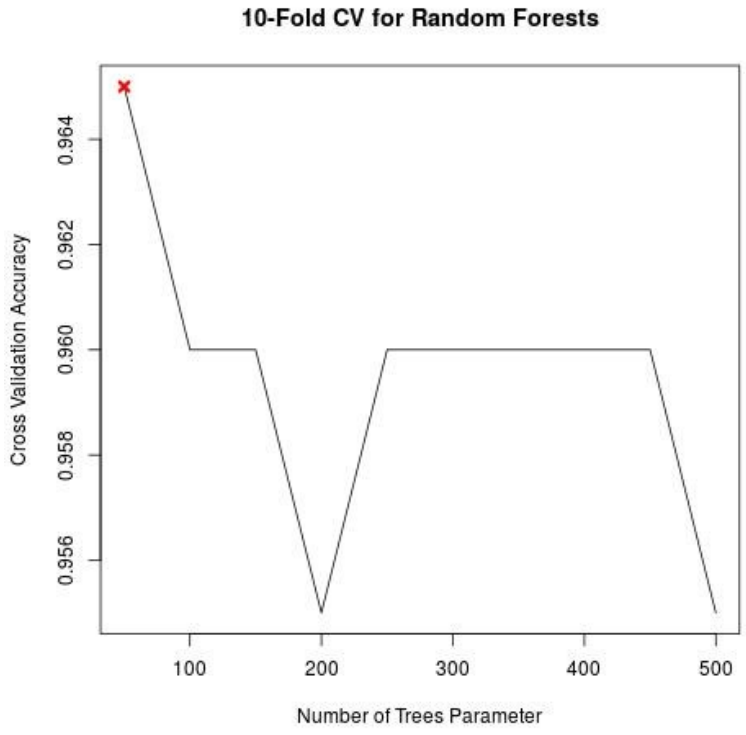

Figure 9: Tuning of number of trees for random forests

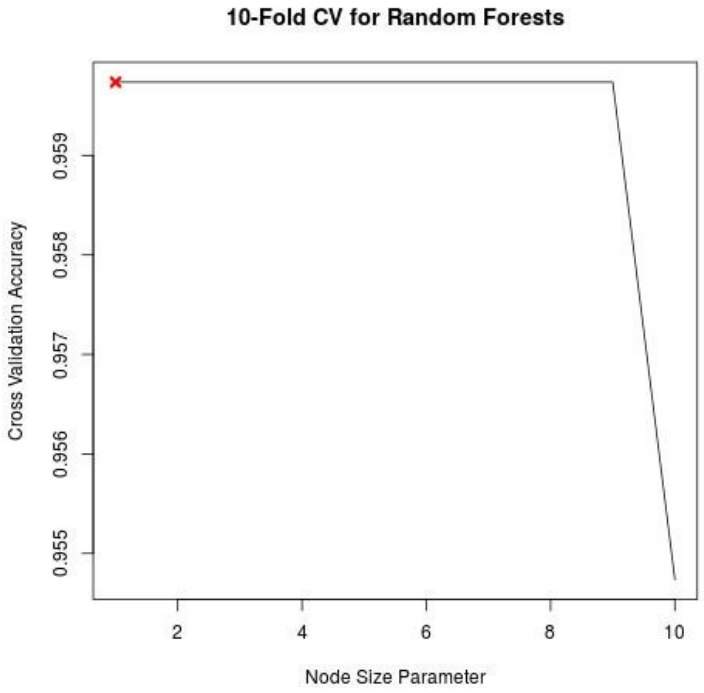

Figure 10: Tuning of node size parameter for random forests

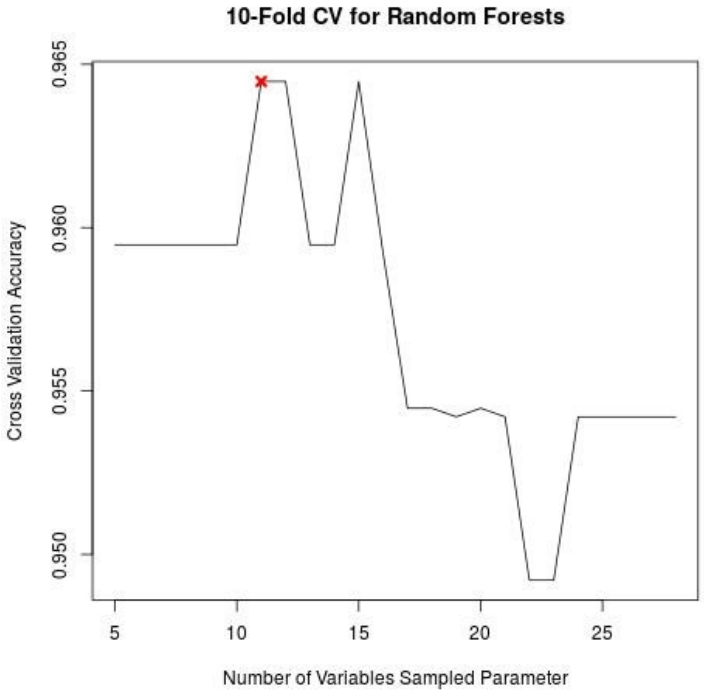

Figure 11: Tuning of mtry or number of variables sampled parameter for random forests

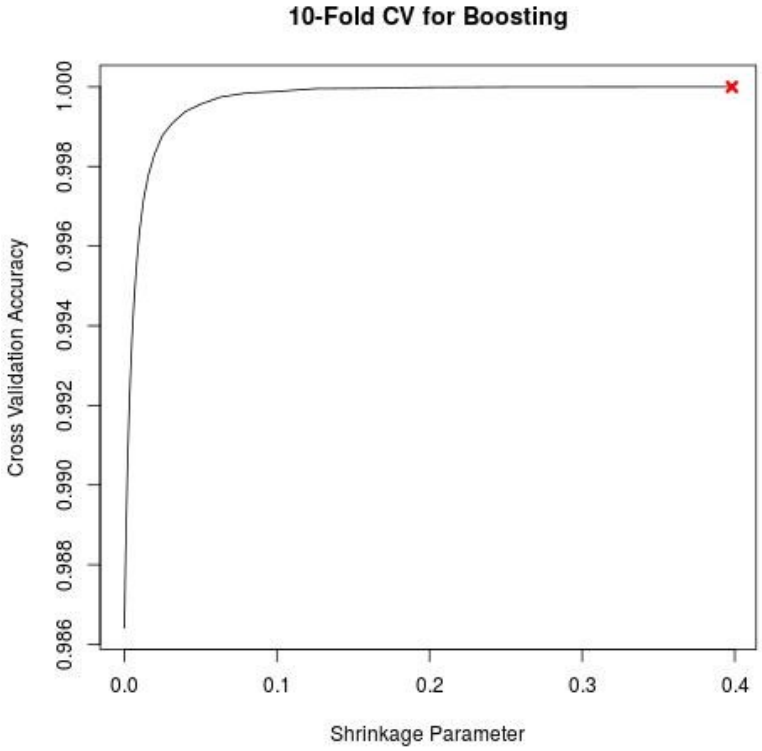

Figure 12: Tuning of shrinkage parameter for boosting

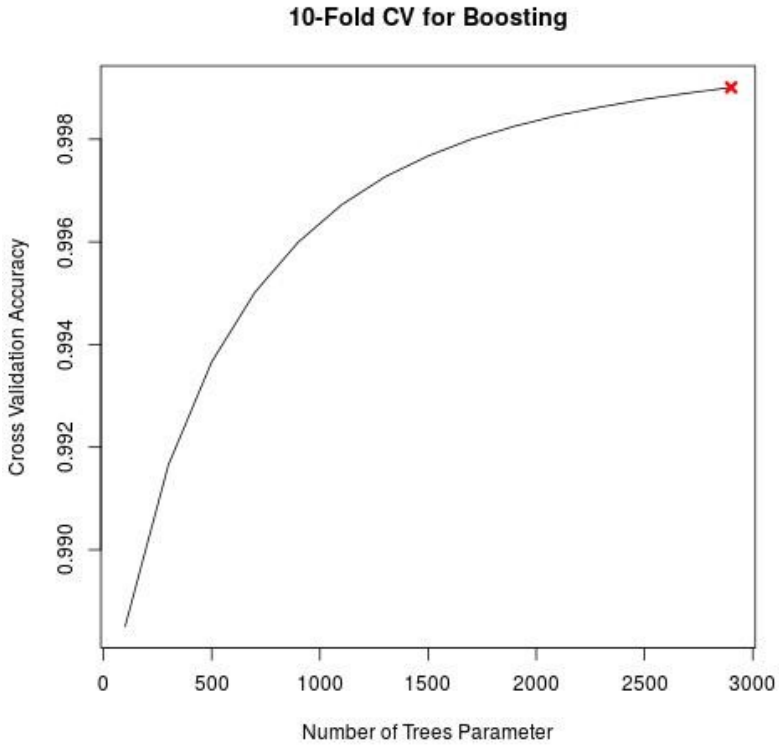

Figure 13: Tuning of number of trees parameter for boosting

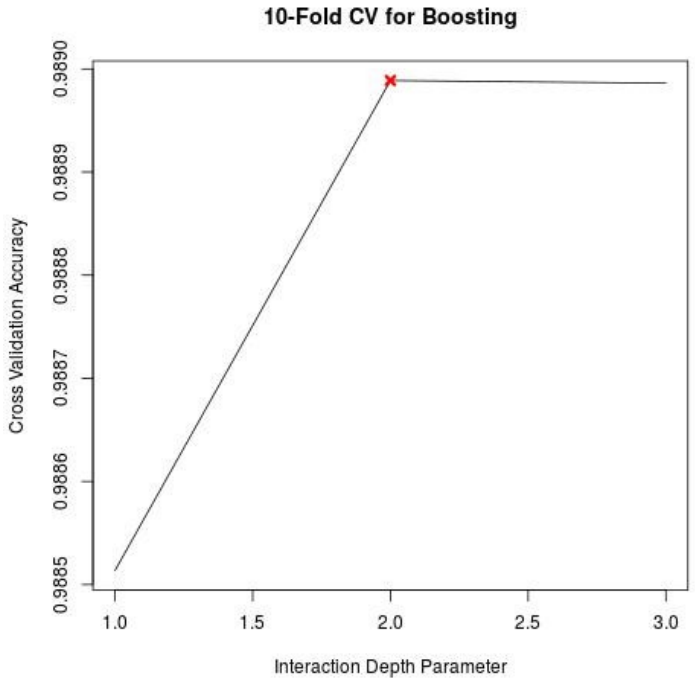

Figure 14: Tuning interaction depth for boosting

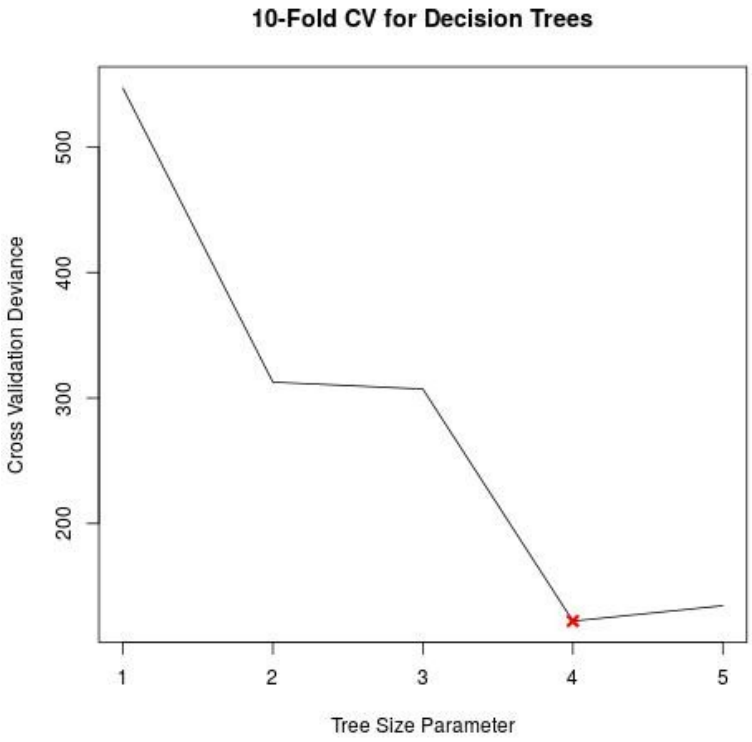

Figure 15: Tuning of tree size parameter for decision trees