Manpreet Sangha | Manpreet.sangha@city.ac.uk | 210048348

COMPARISON OF DECISION TREE(DT) AND RANDOM FOREST(RF) APPLIED TO THE RED WINE DATASET

Brief description and motivation of the problem

- Use supervised learning approaches DT and RF to build classification models.
- Compare and analyse performance of the models to predict quality of the red wine ('bad', 'average' and 'good').
- Compare results with similar implementations [10] in the past using same red wine dataset[11][12][13].

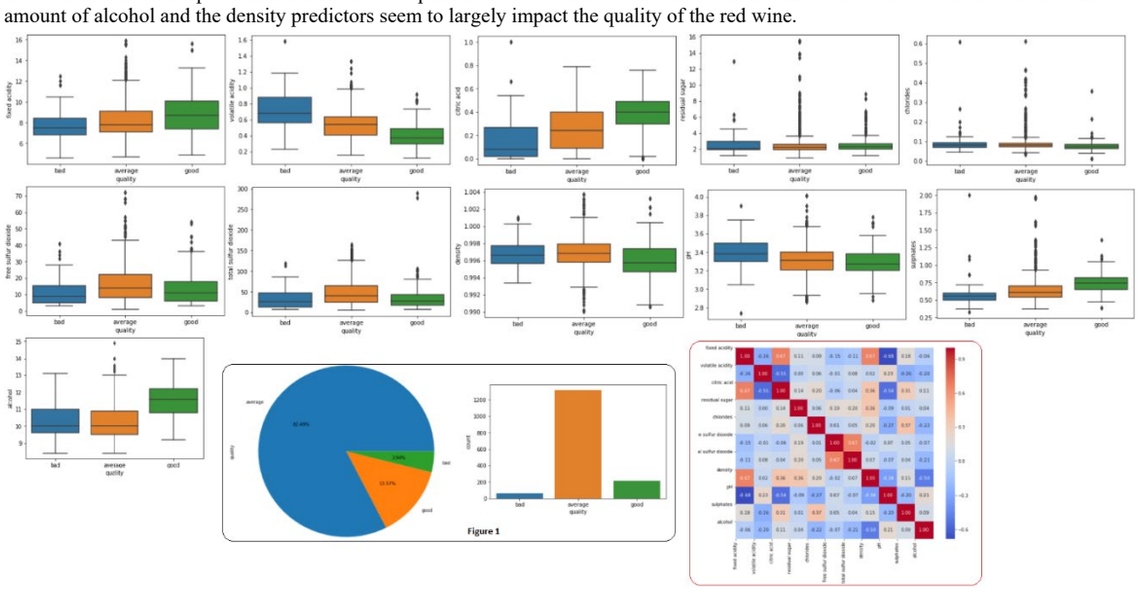
Initial analysis of the data set including basic statistics

- Red wine dataset, has 1599 rows and 12 columns, and is obtained from Kaggle website [11]
- There are 11 predictors (ratio) and 1 target (multiclass) with latter's values ranging from 3 to 8 in original dataset [11]
- For this coursework the red wine will be labelled as bad=0 (3-4), average=0.5 (5-6), good=1 (7-8)

	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
count	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000	1599.000000
mean	8.319637	0.527821	0.270976	2.538806	0.087467	15.874922	46.467792	0.996747	3.311113	0.658149	10.422983	5.636023
std	1.741096	0.179060	0.194801	1.409928	0.047065	10.460157	32.895324	0.001887	0.154386	0.169507	1.065668	0.807569
min	4.600000	0.120000	0.000000	0.900000	0.012000	1.000000	6.000000	0.990070	2.740000	0.330000	8.400000	3.000000
25%	7.100000	0.390000	0.090000	1.900000	0.070000	7.000000	22.000000	0.995600	3.210000	0.550000	9.500000	5.000000
50%	7.900000	0.520000	0.260000	2.200000	0.079000	14.000000	38.000000	0.996750	3.310000	0.620000	10.200000	6.000000
75%	9.200000	0.640000	0.420000	2.600000	0.090000	21.000000	62.000000	0.997835	3.400000	0.730000	11.100000	6.000000
max	15.900000	1.580000	1.000000	15.500000	0.611000	72.000000	289.000000	1.003690	4.010000	2.000000	14.900000	8.000000

Table 1 - all predictors with their minimum, maximum, mean and standard deviation values

- The boxplots help identify the distribution of each predictor with red wine quality and some outliers. The distribution shows that red wine quality increases with increase in fixed acidity, citric acid, sulphates and notably alcohol.
- The count plot and pie chart show that there are 63 bad (3.94%), 1319 average (82.49%), and 217 (13.57%) good instances respectively. The average cases being most instances.
- The correlation heatmap which is a measure of dependence of shows the correlation with the inner variables or with the labels. The amount of alcohol and the density predictors seem to largely impact the quality of the red wine.



Summary of the two ML models with their pros and cons

Decision Tree

- partitions an input dataset into a treelike structure following a supervised, top-down, and heuristic approach to maximize information gain about the attributes [5][6][7] for classification or regression purposes [2].
- We start with root node which may branch out to a decision or a leaf node, according to decision criteria at the nodes, with each node specifying a "test" on an attribute.

Pros

- Easily scrutinized and understood by humans [5].
- Flexibility to handle discrete and continuous inputs [4]p603,[5].
- Fast to fit, relatively robust to outliers, scale well to large datasets [4]603.
- Requires little or no need for data preparation as split points are based on ranking of the data points [4]603.

Cons

- Small changes to input can cause large impact on the tree structure (i.e., high variance) [4]p604.
- Noise in the data results in overfitting [6].
- The level of difficulty to work with a DT increases with increasing depth.
- Difficult to implement in regression problems as DT's results in discontinuities at the split boundaries [2]p666.

Random Forests

- is an ensemble learning method used for classification [13] or simply a collection of DT's that are generated using random subsets of the main dataset using random predictors.
- Random predictors for each split in the DT are drawn from a finite set of predictors with replacement each time [3]p242-43.
- The majority vote from the set of predictions determines the outcome [3]p241.

Pros

- The outcome is less variable and more reliable [3]p244.
- Useful for getting feature importance and uncertainty estimates
- Removes the need to create validation set as the model can be trained over the entire training set.
- The effect of noise is less as the RF procedure doesn't give weight to any subset of predictors [8]22.

Cons

- RF models are not easily interpretable to humans [9].
- Computationally intensive as large datasets consume a lot of memory [9].
- Hyperparameter tuning leads to overfitting [9].

Hypothesis statement

- Based on previous papers RF model has the lowest average error rate when run against large number of trees [13] and is more accurate to predict the wine quality dataset [10][12][13].
- The RF testing time is considerably longer than a DT. The generalisation error reduces when the number of DT's are large.
- Over/under sampling introduces cost into the RF model. The minority class (i.e., 'bad' wine quality) is likely to have the highest cost of misclassification.

Description of the choice of training and evaluation methodology

- Splitting the red wine dataset randomly into training (70% = 1119 instances) and testing (30% = 480 instances) datasets
- Cross validating the training model using 10 fold cross validation.
- Using classification error to compare the best performing model for Decision Trees

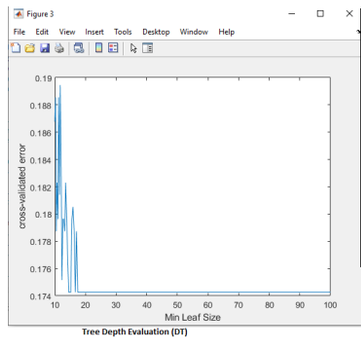
Choice of parameters and experimental results

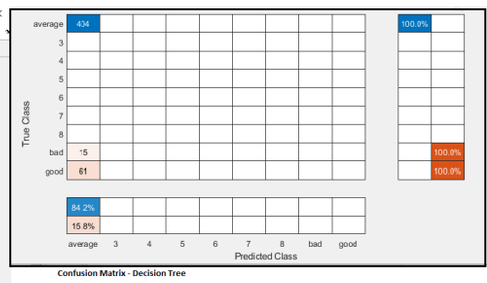
Decision Tree

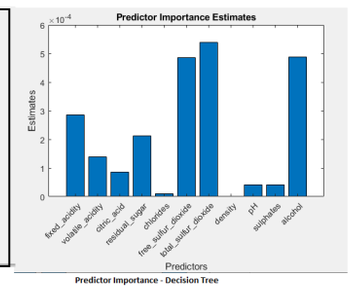
- Choice of parameters
  - Best leaf size is between 20-100 as per cross validated error graph
  - Auto optimising the 10 fold cross validated Decision Tree shows best performance with 526 as minimum leaf size
- Experimental results
  - Resubstitution classification loss for model generated using 'deviance or cross entropy' (0.1242) is more than 'Gini index'(0.1108). Hence, model generated by 'Gini Index' performs slightly better.
  - Controlling leafiness using exponentially spaced set of values from 10 through 100 that represent the minimum number of observations per leaf node [16]
  - Best performing model with lowest classification error is the one generated with min 10 to 100 leaf per node.

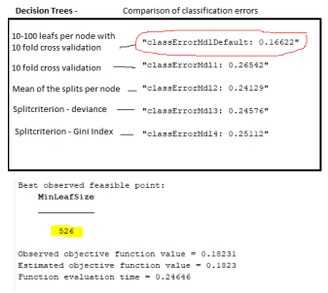
Random Forests

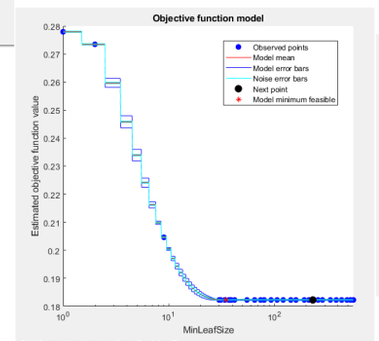
- Choice of parameters
  - Using bootstrapped-aggregated (bagged) decision trees i.e., TreeBagger[15] with random predictor selection at each split
- Experimental results
  - The number of predictors to select at random is 4 per split.
  - The number of trees in the forest is 140 trees.

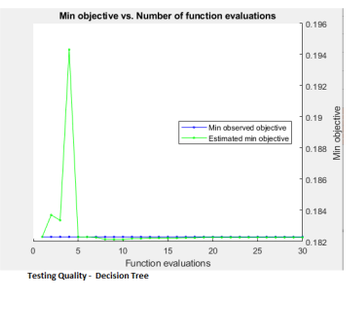


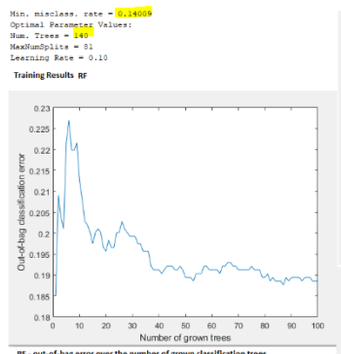


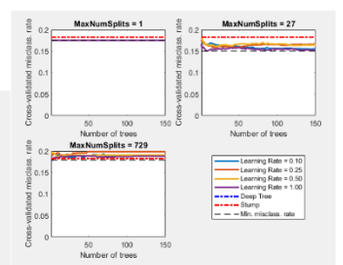


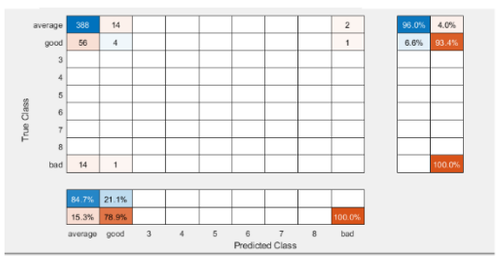


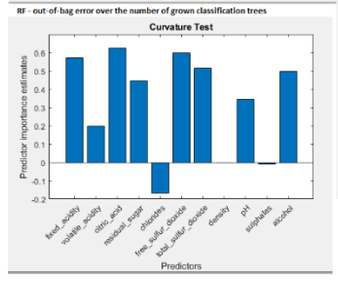


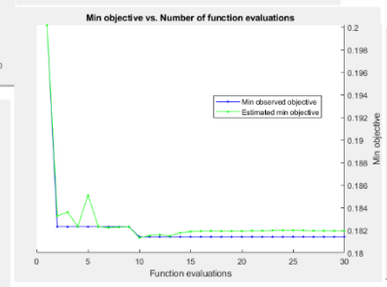


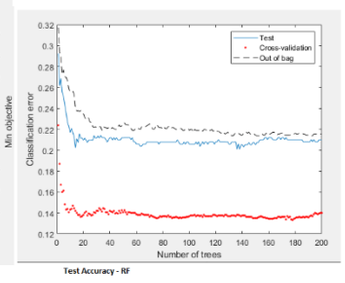












RF Experimentation results

Analysis and critical evaluation of results

- Only 15 out of 480 testing samples, as per confusion matrix for decision trees, were classified as 'bad' meaning the sample instances of quality was not extensive or not correctly labelled in the original data. Further testing with more extensive data may yield better predictive estimates of the wine quality labels.
- The training data is not enough to evaluate the predictive quality of the RF. The ensemble will overfit and produce optimistic predictive results. To obtain better idea of the ensemble, evaluate the ensemble on complete dataset, then by cross validation and then on the out of bag data in case classification ensembles which is the case for this coursework. Applying the methodology and plotting the loss or classification error for out of bag, testing data and the cross validation remain the same as the number of trees grow in the ensemble as shown in the 'Test Accuracy' curve of RF experimentation. [14]
- The RF ensemble was optimised using cross validation by exponentially increasing the tree-complexity level for subsequent ensembles from decision stump (one split) to at most n-1 splits (n= number of instances in the data). The learning rate of 0.1 was applied and that can be experimented further to see whether further optimization can be achieved [16].
- Each curve (between cross validated misclassification rate and number of trees for RF) contains a minimum cross-validated misclassification rate occurring at the optimal number of trees in the ensemble. The learning rates used were 0.10, 0.25, 0.50, 1.00. The maximum number of splits achieved is 81, number of trees is 140 and the learning rate is 0.10 that yields the lowest misclassification rate 0.14009. [18]

Lessons learned

- Be careful with data preparation as improper usage may increase the cross-validation errors and misclassification costs
- Lots of experimentation needed to tune hyperparameters for RF

Future work

- Applying prediction selection techniques such as 'curvature' and 'interaction-curvature' to enhance tree interpretation and predictor importance [16]
- Experimenting with classification loss functions such as 'logit','quadratic','exponential' [17] to lower predictive inaccuracy of classification models
- Create models with other algorithms to deal with outliers as evident in the boxplots in Initial Analysis section
- Experimentation with pruning methodologies to create balanced trees

References

[1] Garbade, M.J. (2018). Regression Versus Classification Machine Learning: What's the Difference? [online] Medium. Available at: https://medium.com/quick-code/regression-versus-classification-machine-learning-whats-the-difference-345c56dd15f7

[2] Bishop, C.M. 2006, Pattern recognition and machine learning, Springer, New York.

[3] James, G.(.M. 2021, An introduction to statistical learning: with applications in R, Second edn, Springer, New York.

[4] Murphy, K.P., 1970 2012, *Machine learning: a probabilistic perspective,* MIT Press, London;Cambridge, Mass;.

[5] Quinlan, J.R. 1990, "Decision trees and decision-making", IEEE transactions on systems, man, and cybernetics, vol. 20, no. 2, pp. 339-346.

[6] Quinlan, J. 1996, "Learning decision tree classifiers", ACM computing surveys, vol. 28, no. 1, pp. 71-72.

[7] QUINLAN, J.R. 1999, "Simplifying decision trees", International journal of human-computer studies, vol. 51, no. 2, pp. 497-510.

[8] Breiman, L. 2001, "Random Forests", Machine learning, vol. 45, no. 1, pp. 5-32

[9] Kho, J. (2018). Why Random Forest is My Favorite Machine Learning Model. [online] Medium. Available at: https://towardsdatascience.com/why-random-forest-is-my-favorite-machine-learning-model-b97651fa3706.

[10] Haoyu Zhang, Zhile Wang, Jiawei He, and Jijiao Tong. 2021. Construction of Wine Quality Prediction Model based on Machine Learning Algorithm. In 2021 5th International Conference on Artificial Intelligence and Virtual Reality (AIVR) (AIVR 2021). Association for Computing Machinery, New York, NY, USA, 53–58. DOI: https://doi.org/10.1145/3480433.3480443

[11] UCI Machine Learning (2009). *Red Wine Quality*. [online] Kaggle.com. Available at: https://www.kaggle.com/uciml/red-wine-quality-cortez-et-al-2009.

[12] Radosavljević, D., Ilić, S. and Pitulić, S., A DATA MINING APPROACH TO WINE QUALITY PREDICTION.

[13] G. Hu, T. Xi, F. Mohammed and H. Miao, "Classification of wine quality with imbalanced data," 2016 IEEE International Conference on Industrial Technology (ICIT), 2016, pp. 1712-1217, doi: 10.1109/ICIT.2016.7475021.

[14] uk.mathworks.com. (n.d.). Test Ensemble Quality - MATLAB & Simulink - MathWorks United Kingdom. [online] Available at: https://uk.mathworks.com/help/stats/methods-to-evaluate-ensemble-quality.html.

[15] uk.mathworks.com. (n.d.). Create bag of decision trees - MATLAB - MathWorks United Kingdom. [online] Available at: https://uk.mathworks.com/help/stats/treebagger.html?searchHighlight=treebagger&s_tid=srchtitle_treebagger_1

[16] uk.mathworks.com. (n.d.). Create bag of decision trees - MATLAB - MathWorks United Kingdom. [online] Available at: https://uk.mathworks.com/help/stats/improving-classification-trees-and-regression-trees.html

[17] uk.mathworks.com. (n.d.). Create bag of decision trees - MATLAB - MathWorks United Kingdom. [online] Available at: https://uk.mathworks.com/help/stats/compactclassificationensemble.loss.html

[18] lost-contact.mit.edu. (n.d.). fitcensemble. [online] Available at: https://lost-contact.mit.edu/afs/inf.ed.ac.uk/group/teaching/matlab-help/R2016b/stats/fitcensemble.html