GLASS IDENTIFICATION GIVEN CHEMICAL COMPOSITION: A COMPARATIVE STUDY OF RANDOM FORESTS AND DECISION TREES

Ayliah Fani

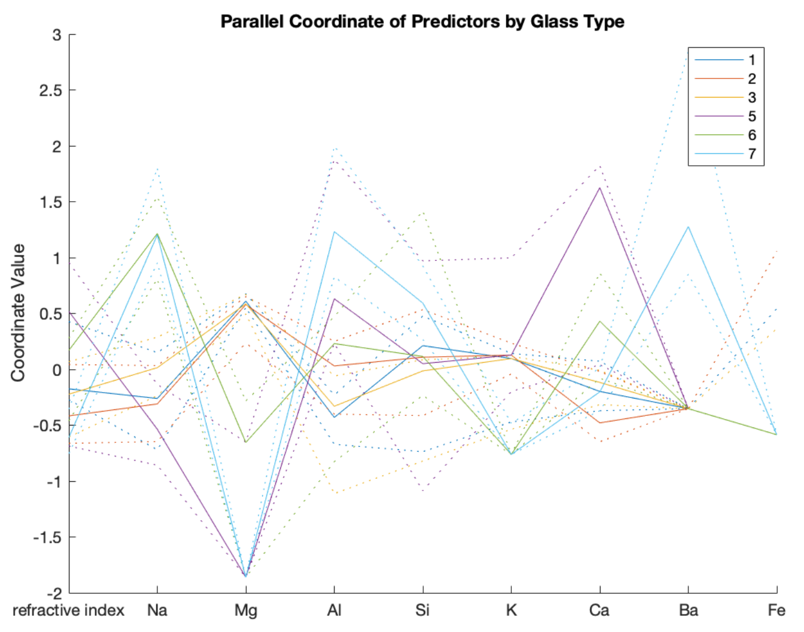
City, University of London Department of Computer Science

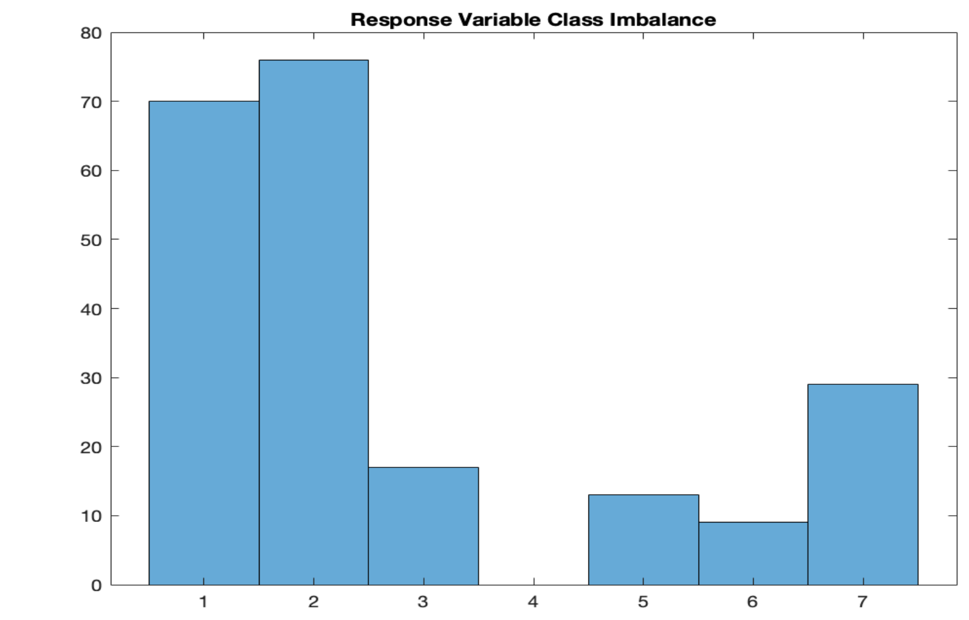
DESCRIPTION AND MOTIVATION OF THE PROBLEM

- The problem definition is a multiclass classification problem in which a glass sample is categorized predictively as one of six glass types provided its respective oxide content (i.e. Na, Fe, K, etcetera), and refractive index.
- Random Forests and Decision Tree machine learning algorithms will be applied to the UCI Glass Identification Dataset to evaluate performance of the methods through their prediction accuracy.
- The two model's predictive performance will be compared with each other, as well as other models in literature.

INITIAL ANALYSIS OF THE DATA SET AND BASIC STATISTICS

- UCI Glass Identification Dataset
- The dataset gives 6 types of glass defined in terms of their oxide content (sodium, potassium, magnesium, aluminium, calcium, silicon, barium and iron) and refractive index. The study of classification of types of glass was motivated by criminological investigation. Glass fragments are frequently found at crime scenes and when properly identified, fragment origin can be determined and as evidence [6].
- The dataset contains 214 instances, and 10 attributes: 9 continuous predictors variables, and one discrete response variable
- The response variable contains 7 classes which pertain to the following glass types; 1: building windows float processed, 2: building windows not float processed, 3: vehicle windows float processed, 4: vehicle windows not float processed (none in this database), 5: containers, 6: tableware, 7: headlamps
- There is a class imbalance present seen from the histogram on the right.. Glass type 4 is not present in the dataset, therefore future predictions cannot be made accurately from the trained models. There is also far more data of classes 1 and 2, which make up 68% of the data, than the other class types, making the remaining 32% of the dataset.
- Parallel coordinate plots aid in the visualization of high-dimensional datasets. The parallel coordinates plot shows the relationship between the standardized predictors and the associated class from the median and outer quantiles of each predictor. Higher or lower oxide contents, particularly Mg, Ca, and Fe, have a strong correlation with specific glass types.





Feature	Refractive Index	Na	Mg	Al	Si	K	Ca	Ba	Fe
Mean	1.5284	13.4079	2.6845	1.4449	72.6509	0.4971	8.9570	0.1750	0.0570
Std	0.0030	0.8166	1.4424	0.4993	0.7745	0.6522	1.4232	0.4972	0.0974

COMPARISON OF MACHINE LEARNING MODELS

RANDOM FORESTS

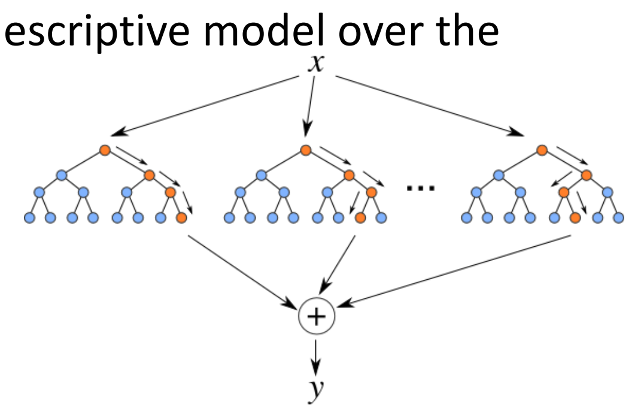
- Random forests (RF) are a supervised ensemble learning method where multiple un-pruned decision trees are combined for classification and regression problems.
- The random forest is built by randomly sampling a feature subset for each decision tree, and/or by the random sampling of training data set for each individual decision tree [2][9].
- Multiple versions of a predictor are generated and then used to get an aggregated predictor. The aggregation makes a prediction for each ensemble via a majority vote when predicting a class [4][2].
- The model is more robust, accurate, and handles overfitting better than its component classifiers [9].

Pros

- Overcomes the problem of overfitting[2].
- In training data, RF is less sensitive to outlier data [2].
- Parameters can be set easily and therefore eliminates the need for pruning the trees[2].
- implicit feature selection, feature importance, and accuracy is generated automatically [2][3].
- Bagging decreases correlation between each DT, and thus increases RF's predictive accuracy [5].

Cons

- Computationally complex and slower when forest becomes large
- Not a well descriptive model over the prediction.

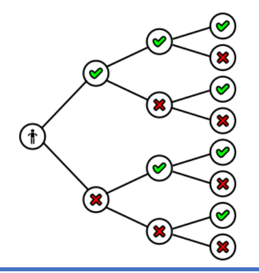


DECISION TREES

- A Decision tree (DT) is a supervised tree-based method where a tree structure having different nodes (ie. root, intermediate, and leaf nodes) is used to solve regression and classification problems [7].
- The decision tree is derived from independent variables, with each node containing a decision or condition over a feature, in turn leading examples to the final prediction [7].
- Trees constructed with fixed training data are prone to overfitting. This is rectified with pruning the fully grown tree. Pruning increases generalization accuracy on unseen data at the expense of the accuracy on the training data [1].

Pros

- There are no prior assumptions about the nature of the data [10].
- Provide human-readable rules of classification and are easy to understand [2][10].
- Can classify both categorial and numerical data [10]



Cons

- High likelihood of overfitting and prone to outliers.
- Trees can become complex if dealing with large or complicated numeric datasets [10].
- Deficient when used with continuous variables.
- Not possible to predict beyond the minimum and maximum limits of the response variable in the training data [4]

HYPOTHESIS STATEMENT

- Both algorithms are expected to have similar predictive accuracies, where the difference in observed misclassification rates of the two models will be statistically insignificant given the small size of the dataset. In the case of larger datasets difference in misclassification rates would be significant.
- Random forests will have a smaller classification loss, increased accuracy, and precision than that of the decision tree model [2] as reported by Gupta [7] and Maclin and Opitz [9].
- Although both models are sensitive to imbalanced data, the random forest model will generally outperform the decision tree
- Both models will be sensitive to hyperparameter tuning

METHODOLOGY

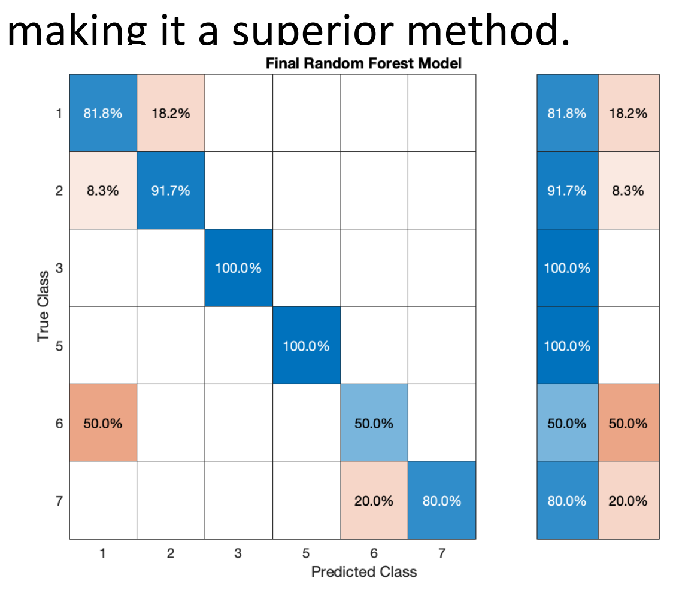
Compare performance measures: classification accuracy, loss, k-fold loss, confusion matrices, time complexity for both algorithms and with values cited in [7].

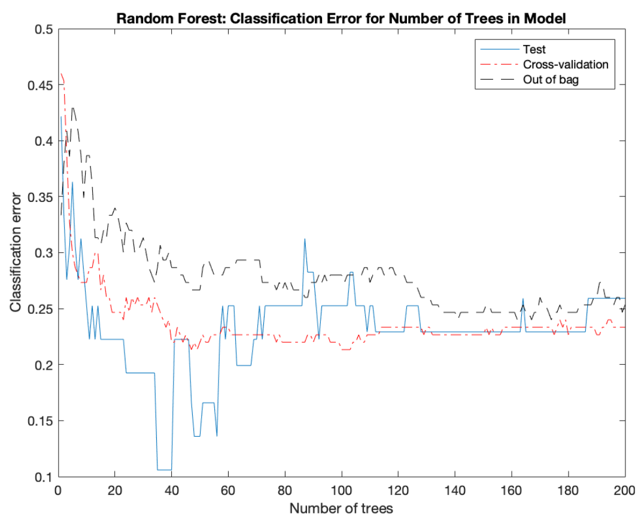
1. Data was pre-processed and statistically evaluated. Feature selection and cleaning was already present in the dataset, therefore nothing was removed or changed. Dataset was partitioned into train, test, and validation sets comprising 15%, 70%, and 15% of the dataset respectively.
2. RF and DT models were trained and performance was individually evaluated via 10-fold cross validation, classification loss, and confusion matrices.
3. Hyperparameters were optimised for both models and impact of hyperparameter optimisation and lack thereof on the models were determined.
4. Final models were created from derived insights and overall performance was compared between the two models.

CHOICE OF PARAMETERS AND EXPERIMENTAL RESULTS

RANDOM FORESTS

- Random forest hyperparameters: number of trees, maximum features, minimum sample leaf
- **Parameters:** the number of trees in the ensemble were optimised by observing the associated classification loss and out of bag error with a forest with n trees.
- **Results:** it was determined that the best results were obtained with a forest of 49 trees as seen from the cross validated model.
- Classification accuracy of the model was 84% and time complexity was 543ms.
- From the error versus number of trees plot below, one can see that the error starts to converge as the number of trees increases; at approximately 70 trees.
- Overall, the RF model has higher classification accuracy and lower loss than that of its DT counterpart, making it a superior method.



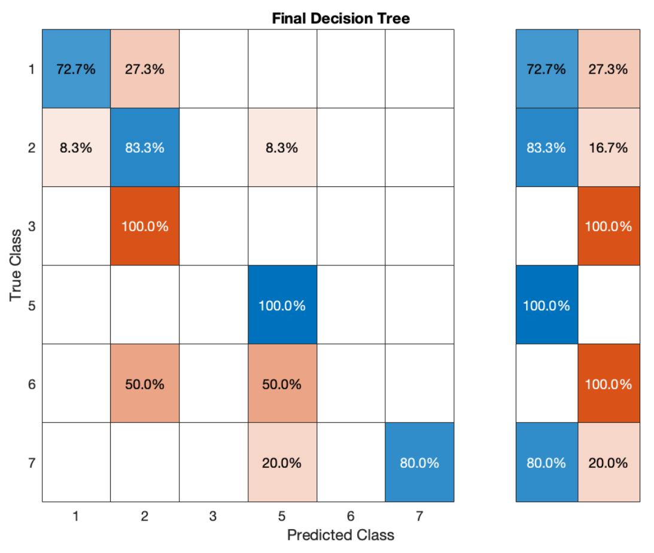


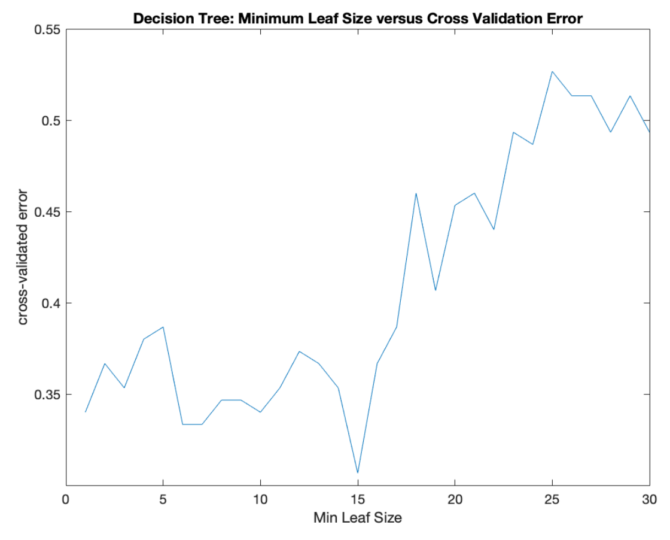
Classification Accuracy	84.38%
Loss	0.1359
Time Complexity (ms)	543.621

DECISION TREES

Decision tree hyperparameters: criterion, max depth, minimum samples split, minimum samples leaf.

- **Parameters:** The minimum leaf size and pruning level were optimised to find the tree depth with lowest classification loss.
- Pruning optimizes tree depth (leafiness) by merging leaves on the same tree branch. Each leaf has at least 'MinLeafSize' observations. Smaller values of `MinLeafSize` yield deeper trees [8].
- **Results:** It was determined that a minimum leaf size was 15 observations, and a pruning level of 1 delivered the lowest classification loss and highest accuracy while maintaining complexity and avoiding overfitting.
- Accuracy of the model was 75% and time complexity was 45.5ms.





Classification Accuracy	75%
Loss	0.2786
Time Complexity (ms)	45.5215

ANALYSIS AND CRITICAL EVALUATION OF RESULTS

- **Decision Tree:** Classification accuracy of 75% was significantly higher than that achieved by Gupta [7] which was 68% and a Maclin and Opitz [9] at 69%. Time complexity was also significantly lower than that reported by [7]; 45.5ms versus 442ms.
- The initial DT model had a classification loss of 0.36. Changing the optimal number of leaves to 15 reduced the loss to 0.30, and pruning reduced loss further to 0.27.
- **Random forest:** Classification accuracy of 84% for the RF model was similar to that achieved by Gupta [7] and Maclin and Opitz [9], 80 percent and 71 percent respectively. Gupta's time complexity- 4484ms was ~8 times larger than that obtained in this study, 543ms.
- Changing the number of trees in the forest to 49 which corresponded to the lowest cross validation error made a significant improvement on classification loss which was reduced to 0.1359 from 0.2592 when using the default 100 provided by Matlab.
- **Overall:** Hypothesis test result 'h' was found to be 0 indicating to not reject the null hypothesis that the two models have equal predictive accuracies for this dataset. Acuraccy of the RF model was 10% higher than DT.
- RF had a significantly higher time complexity than DT. This indicates that RF is more computationally demanding than that seen by a DT. With a larger dataset this time complexity would become much more apparent. Given the smaller size of the dataset- having only 214 instances, the time complexity is not of major concern when comparing the two methods in this instance.

LESSONS LEARNED AND FUTURE WORK

- **Lessons learned:** Random forests outperform decision trees, are more robust, and accurate. Although this comes at a cost due to increasing time complexity and computational requirements as a dataset become larger and more complex.
- **Future work:** Further optimise the models through feature engineering, determining feature importance, and further modifying hyperparameters; maximum feature and minimum sample leaf for RF, and criterion, minimum sample split, minimum samples leaf.
- Look at boosted tree models such as RUSboost to account for the class imbalance present in the dataset.

References

[1] Tin Kam Ho, 'Random decision forests', in *Proceedings of 3rd International Conference on Document Analysis and Recognition*, Montreal, Que., Canada, 1995, vol. 1, pp. 278–282, doi: 10.1109/ICDAR.1995.598994.

[2] Ali, R. Khan, N. Ahmad, and I. Maqsood, 'Random Forests and Decision Trees', *International Journal of Computer Science Issues*, vol. 9, no. 5, pp. 272-278, Sep. 2012.

[3] N. Horning, 'Introduction to decision trees and random forests', American Museum of Natural History's Centre for Biodiversity and Conservation.

[4] L. Breiman, 'Bagging predictors', *Mach Learn*, vol. 24, no. 2, pp. 123–140, Aug. 1996, doi: 10.1007/BF00058655.

[5] 'Bootstrap Aggregation, Random Forests and Boosted Trees | QuantStart'. https://www.quantstart.com/articles/bootstrap-aggregation-random-forests-and-boosted-trees/ (accessed Dec. 07, 2020).

[6] I. W. Evett and E. J. Spiehler, 'Rule Induction in Forensic Science', in *Rule Induction in Forensic Science*.

[7] A. Gupta, 'Classification Of Complex UCI Datasets Using Machine Learning And Evolutionary Algorithms', vol. 4, no. 05, p. 10, 2015.

[8] 'Improving Classification Trees and Regression Trees - MATLAB & Simulink'. https://www.mathworks.com/help/stats/improving-classification-trees-and-regression-trees.html.

[9] R. Maclin and D. Opitz, 'An Empirical Evaluation of Bagging and Boosting', *Proceedings of the National Conference on Artificial Intelligence*, Feb. 1998.

[10] Y. Zhao and Y. Zhang, 'Comparison of decision tree methods for finding active objects', *Advances in Space Research*, vol. 41, no. 12, pp. 1955–1959, Jan. 2008, doi: 10.1016/j.asr.2007.07.020.