PS0002 Introduction to Data Science and Artificial Intelligence Report

Ivan Wong U1840701J
Nigel Yee U1840714B
Tan Jun Yong U1840401G

I. INTRODUCTION

This project aims to give a full analysis of a moderately large dataset, with the data preparation, analysis and machine learning concepts introduced thus far in the course. More specifically, the Glass dataset from the mlbench R package was chosen for this purpose. This report seeks to investigate whether features of glass can be used to predict the type of glass accurately and to what extent. It details the preprocessing techniques used, the machine learning (ML) algorithm that the data was trained on and used for generating predictions, as well as the final analysis of the performance of the entire pipeline. For preprocessing, highly-correlated features were identified for removal, the data underwent normalization, and finally, the training data was upsampled to overcome class-imbalance in the dataset. The ML algorithm chosen was Random Forest, and its performance was then evaluated using overall accuracy and a confusion matrix. To support analyses, the following R packages were used: caret, dplyr and reshape2, in addition to mlbench for the dataset.

II. ANALYSIS

The Glass dataset consists of 214 datapoints. There are 9 features: RI, Na, Mg, Al, Si, K, Ca, Ba and Fe. The class to be predicted for each datapoint is Type, the type of glass given the features of each datapoint. There are 6 Glass Type classes: 1, 2, 3, 5, 6, 7. Since there are 6 distinct categories, this problem can be framed as a classification problem.

Figure 1 plots the distribution of Glass Types in the dataset. Out of 214 datapoints, the proportion in percentage are as follows: Type 1 – 32.7%, Type 2 – 35.5%, Type 3 – 7.9%, Type 4 – 6.1%, Type 5 – 6.1%, Type 6 – 4.2% and Type 7 – 13.6%.

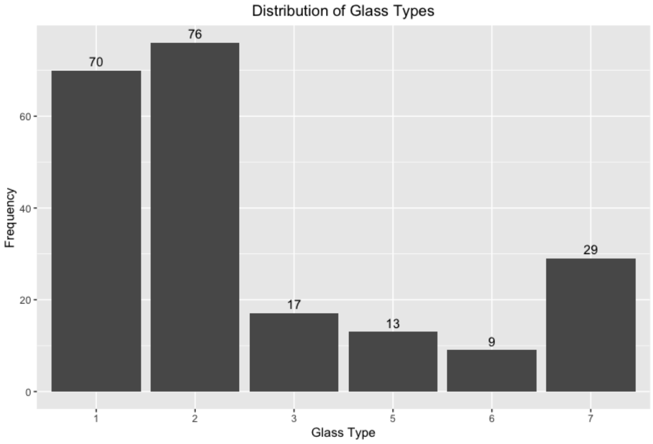

Figure 1: Plot of Glass Type distribution in the dataset

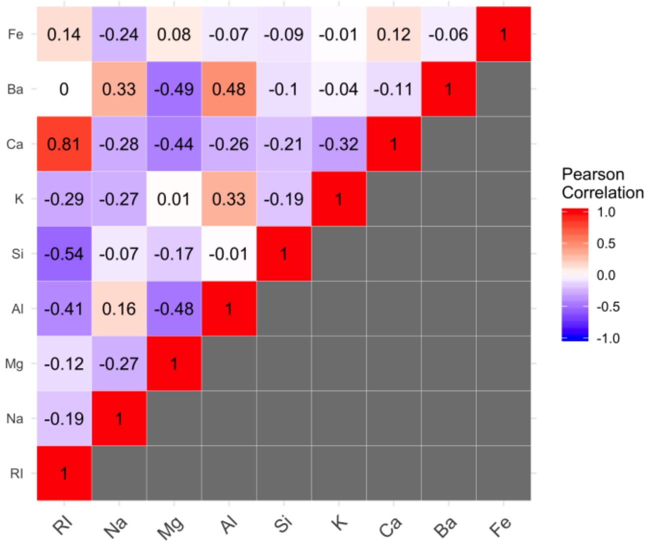

Figure 2: Plot of correlation heatmap between feature variables

Figure 2 plots the correlation heatmap between every feature variable pair using Pearson Correlation. Highly correlated (positively or negatively) features should be excluded from the training set since the presence of highly correlated features can lead to instability in the weights of linear models, creating higher variance and poorer generalization of the trained model. As can be seen in the heatmap, most factors are not strongly correlated to each other. The highest correlation happens between RI and Ca with a correlation coefficient of 0.81. Taking the threshold to be 0.9, we therefore still include both RI and Ca in the training dataset.

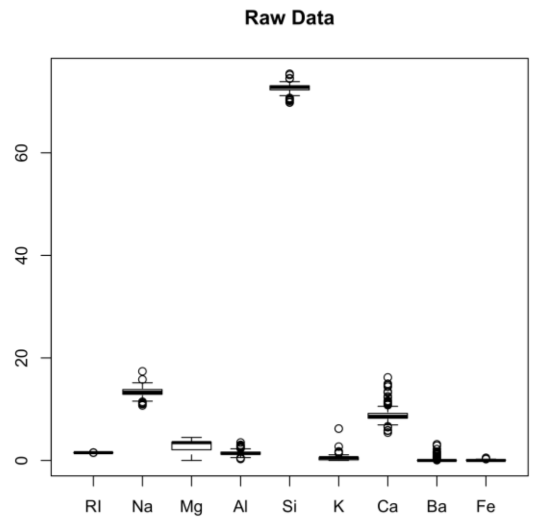

Figure 3: Boxplot of feature value distribution before normalization

Figure 3 shows the boxplots of the distribution of each feature's values, with the bottom whisker being the minimum value, lower box line being the 25th percentile, middle box line being the 50th percentile, upper box line being the 75th percentile and the highest whisker line being the maximum value. Dots plotted are deemed as outliers as they occur too infrequently. It can be seen that the features have varying distributions (different means and standard deviations), and hence differing scales. Varying scales can lead to poor fitting of linear models, as features with larger magnitudes erroneously receive higher importance. Hence the features were standardized to have mean of 0 as can be seen in Figure 4, which shows the distribution of feature values after normalization.

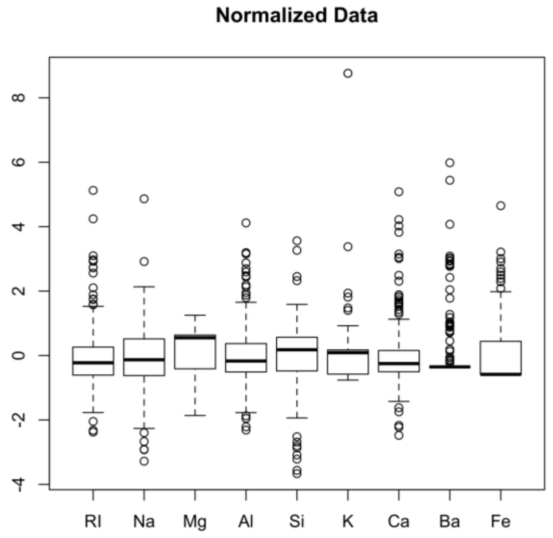

Figure 4: Plot of feature value distribution after normalization

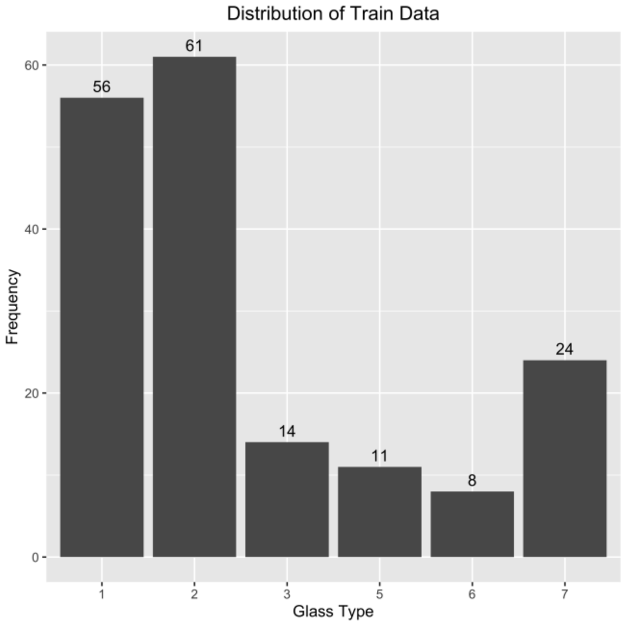

Figure 5: Plot of train data class distribution before upsampling

The dataset was split into a training set and test set, by randomly sampling the dataset with a 80-20 split. Due to class imbalance in the training set, as can be seen in Figure 5, every Glass Type was upsampled to have an equal number of 61 samples each, to prevent bias in the model. The test data Glass Type distribution is as follows in Figure 6.

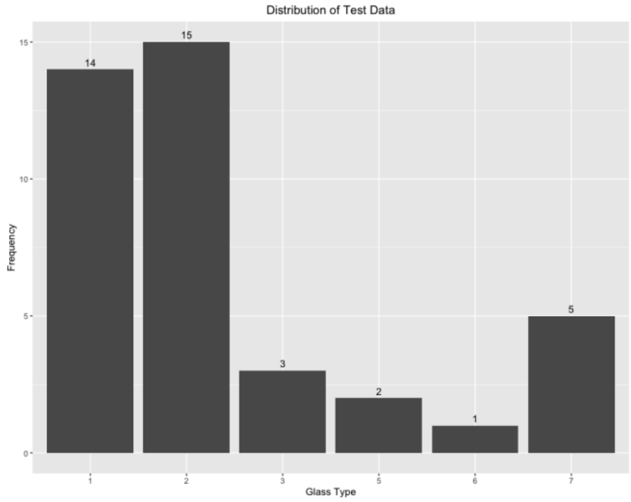

Figure 6: Plot of test data class distribution

Class balancing was not done on the test dataset because the distribution of the data should be preserved "as-is", to reflect the natural class imbalance of the raw dataset and allow test results to be valid.

The random forest (RF) algorithm was selected as the algorithm for classification on this dataset. The main reason is due to random forest being an ensemble algorithm of weak random decision tree learners. Each decision tree selects features randomly to split on, and predictions are generated by averaging the output predictions of all the random tree learners. This allows the model to have higher complexity together with lower variance. Furthermore, as a tree-based algorithm, RF has high explainability on each feature's predictive ability. The RF model was trained using 10-fold cross validation, which splits the training dataset into 10 partitions. Over 10 iterations, each iteration would train a RF on 9 randomly selected partitions, and 1 validation partition for evaluating accuracy. The reported training accuracy of the RF is the result of the average of all 10 iterations of cross-validation.

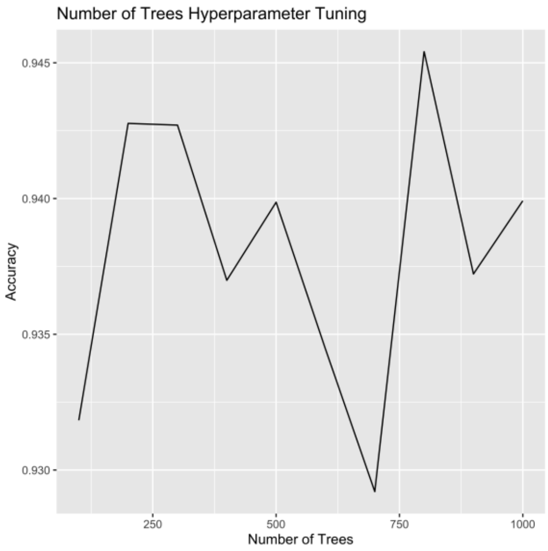

Figure 7: Plot of random forest accuracy as number of trees varies

In addition to cross-validation, the number of trees hyperparameter was optimised. Figure 7 shows how the RF varied in accuracy as the number of trees was varied in increments of 100, from 100 trees to 1000 trees. The best model with the highest training accuracy of 94.5% (10-fold cross validation) was achieved using 800 trees.

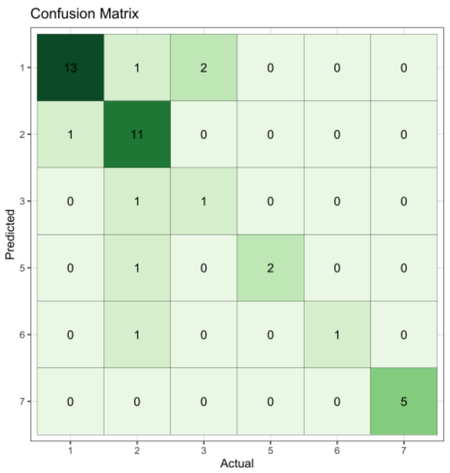

Figure 8: Plot of confusion matrix on test dataset

The overall accuracy of the RF model was 82.5%, predicting 33 test samples correctly out of 40, with the given confusion matrix in Figure 8. Type 1, Type 2 and Type 7 had high accuracies, at 92.8%, 73.3% and 100% respectively. However, Type 3, 5 and 6 have very sparse datapoints, leading to inaccurate estimation of their class accuracies.

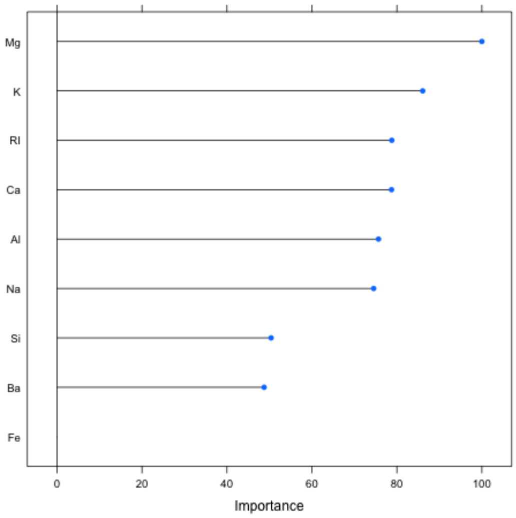

Figure 9: Plot of feature importance

Figure 9 indicates that the Mg content of a sample has the highest predictive power of the Glass Type, while its Fe content is the least predictive.

III. CONCLUSION

It can be concluded that the given features in the dataset has high predictive ability to predict the Glass Type, since the accuracy of the final model was a considerably high 82.5%.

However, the size of the dataset that was used to train the model was considerably small and imbalanced, which required upsampling for the training set and also resulted in only 40 samples for the test dataset. This caused some Glass Type classes such as 3, 5 and 6 had barely more than a couple of datapoints, which were definitely insufficient to calculate class-level accuracy, precision and recall. A larger dataset would enable the splitting into train, validation and test sets, which would allow hyperparameter tuning to be based more accurately on the validation set instead of the train set, in so preventing overfitting from occurring.

Finally, RF performance can be compared against other simpler and more complex classifier algorithms, such as multi-class logistic regression, support vector machines and XGBoost, to ascertain the best model for this dataset.