Wine Rating Prediction

This report is with respect to the wine rating prediction system that our firm wants to develop. We choose the Red Wine dataset in order to conduct this prediction. The dataset can be described as follows:-

It has 1599 instances. This dataset consists of 12 attributes which are described below:

1. Fixed Acidity	The predominant **fixed acids** found in **wines** are tartaric, malic, citric, and succinic. Their respective levels found in **wine** can vary greatly.
2. Volatile Acidity	It is mostly caused by bacteria in the **wine** creating acetic **acid** — the **acid** that gives vinegar its characteristic flavor and aroma.
3. Citric Acid	This acid gives citric flavor to the wine.
4.Residual Sugar	It refers to any natural grape **sugars** that are left over after fermentation ceases.
5.Chlorides	Salt concentration in wine,
6.Free Sulfur dioxide	Used as a preservative to control bacterial growth in the wine.
7.Total Sulfur dioxide	Mixture of free sulfur dioxide and sulfur bound with acetaldehyde, some polyphenols, ketones, sugars or acids.
8.Density	It is determined by the concentration of alcohol, sugar, glycerol, and other dissolved solids.
9. pH	It is the measure of acidity in the wine.
10.Sulphates	They are used as preservatives.
11.Alcohol	It refers to the alcohol content in the wine and measured in %.
12.Quality	Quality score between 0-10. (That's what we have predicted.)

Table 1: Red Wine dataset features

Data processing is defined as the operations performed on a given set of data to extract the required information in an appropriate form. We processed the data using two algorithms and evaluated it using suitable metrics so as to predict the rating of the red wine which ranged from 0-10.

We pre-processed the data too which included feature scaling using standard scaling method of the data set so that all the data is between -1 and 1, such that we don't find it difficult to process it as the values can be too large.

We have chosen to work with **Scikit-Learn** in python, using two machine learning algorithms which are "Random Forest" and "Decision Tree". This choice is explained by the fact that we are interested in classification in our experiments on the red wine dataset. These algorithms are classification dedicated, and here are our motivation to choose them:

Decision tree: also known as classification tree, it can be used in decision analysis to visually and explicitly represent decisions and decision making. As the name goes, it uses a tree-like model of decisions. It's used for deriving a strategy to reach a particular goal. This algorithms have some good advantages: Simple to understand, interpret and visualize, also it can handle both numerical and categorical data as well as multi output problems handling. It requires relatively little effort from users for data preparation and nonlinear relationships between parameters do not affect the tree performance. In a decision tree an input is entered at the top (of the tree) and as it traverses down, the data gets bucketed into smaller sets.

Random forest: the random forest algorithm takes the decision tree algorithm process to the next level by combining trees with the notion of an ensemble (ensemble of trees). It creates a set of decision trees from randomly selected subset of training set. It then aggregates the votes from different decision trees, by reducing the amount of correlation between trees, and thus helping reduce the variance of the final tree to decide the final class of the test object. It predicts based on the majority of votes from each of the decision trees made. A single decision tree may be prone to a noise, but aggregate of many decision trees reduce the effect of noise and give more accurate results, in other ways it's a way to improve the performance of decision tree algorithm as well as avoiding the training set over fitting.

We have done feature scaling using standardization because variables don't have the same scale which can cause issues while testing our model.
After a considerable amount of iterations we have set the number of decision trees to be 400 (n_estimators = 400) as it give us the optimum results for our model, below and above 400 the value of accuracy decreases.
To build our model we have used 70/30 split method to divide our dataset into test and train data, also we have used cross validation to estimate the confidence interval. Accuracy and Root Mean Square Error (RMSE) being the two metrics we've used.

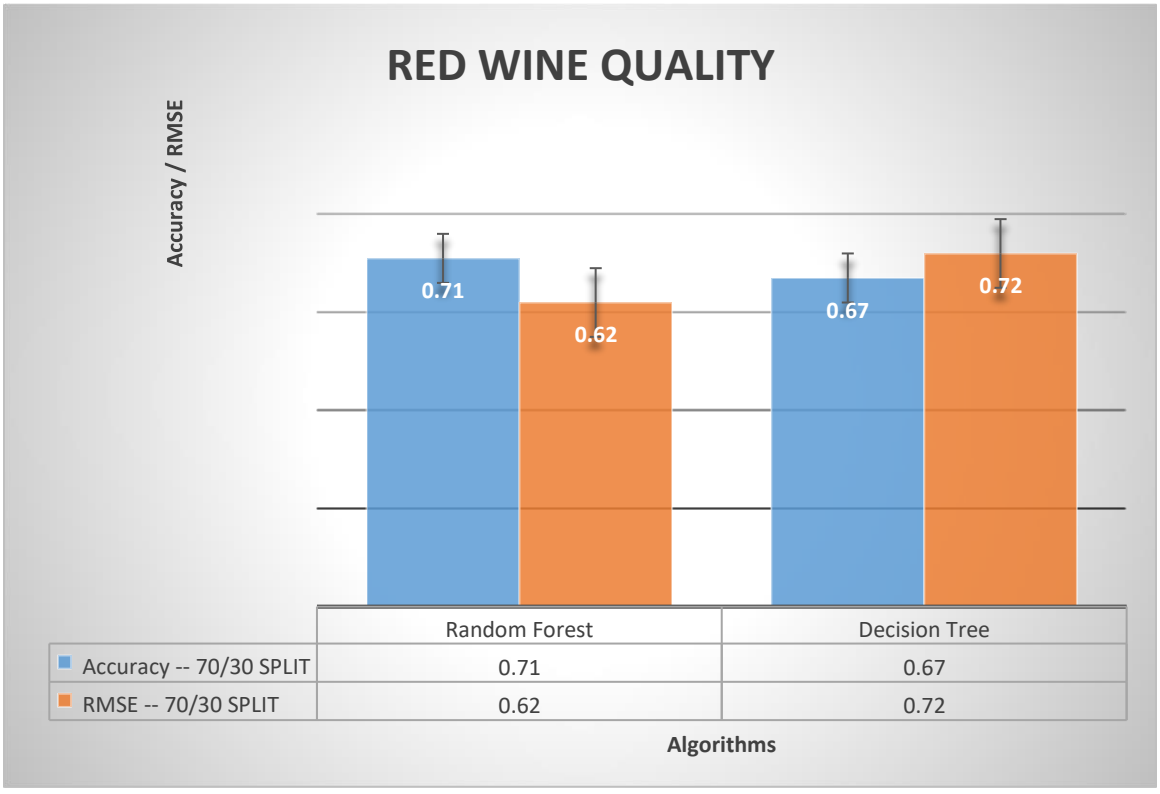

Figure 1: Red wine quality classification: Accuracy and RMSE as a function of algorithms.

As can be seen from the figure above, Random forest accuracy score is 0.71 and the Decision Tree accuracy is 0.67(about 0.04 difference) Also, the confidence interval for Random forest accuracy ranges between 0.68-0.73 while confidence interval for the Decision Tree accuracy ranges between

0.55-0.71.On the other hand Random forest RMSE is 0.62 and the Decision Tree RMSE is 0.72(about 0.10 difference) Moreover, the confidence interval for Random forest RMSE ranges between 0.600.67 while confidence interval for the Decision Tree RMSE ranges between 0.66-0.92 again Random forest with less error rate performs better . It should be kept in mind that we used a dataset of only 1599 instances, it will be interesting to see how the algorithms perform using a much bigger dataset, i.e. with millions of instances and more than 12 features, then compare the evaluation metrics scores to see which algorithm outweighs the other.

We can conclude by our evaluation that Random Forest algorithm gives better results than the Decision Tree algorithm. The figure described above clearly shows the same. We got to know this on the basis of the metrics used and the confidence intervals. Clearly, Random forest had a better accuracy and lesser RMSE than decision tree algorithm. As we know that random forest is the advanced version of the decision tree algorithm, so it was expected that the results would be better. According to us, both the algorithms that we used fitted well and gave a considerable good result, but if we have to choose any one algorithm out of the two that was worth pursuing, then it has to be the random forest algorithm ,the advantages have been mentioned more specifically in the evaluation part above.