Dataset:
Wine Quality Ratings and Chemicals (Red Wine)

1) Data & Pre-Processing:

The dataset provided is in the csv format and there are no missing values in the dataset. The dataset provided constitutes of 12 features; each defining characteristics of Red Wine. Our target variable is 'Quality' and the input variables are:

1 - fixed acidity
2 - volatile acidity
3 - citric acid
4 - residual sugar
5 - chlorides
6 - free sulphur dioxide
7 - total sulphur dioxide
8 - density
9 - pH
10 - sulphates
11 - alcohol

The dataset does not require feature scaling because the standard deviation is low. The classes are ordered but not balanced, that is, there are more number of incidences for classes that are of 'average' quality compared to 'low' or 'medium' quality wines (See figure 1). Because our target classes are unbalanced, we've done upscaling which we have discussed later in the report.

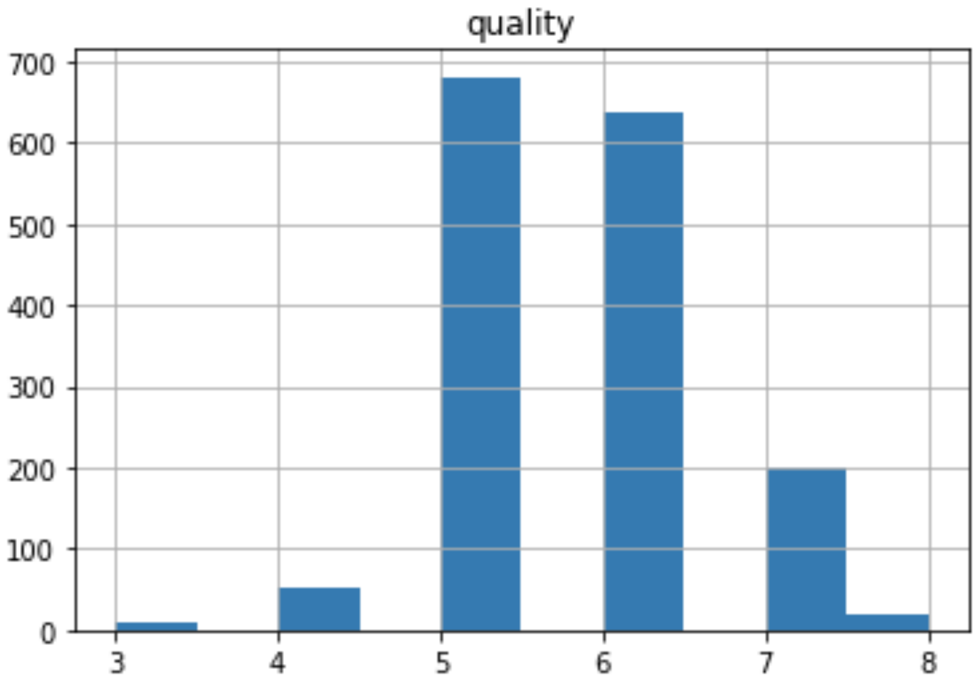
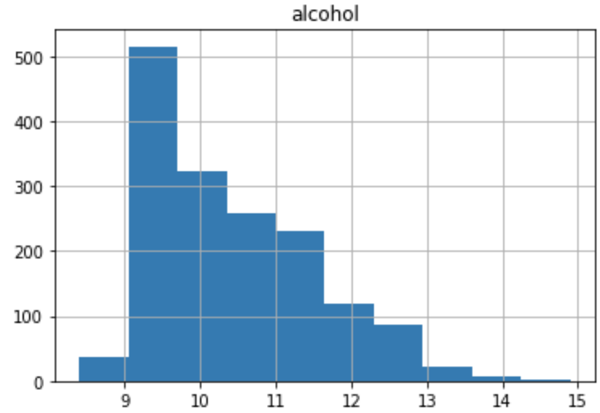
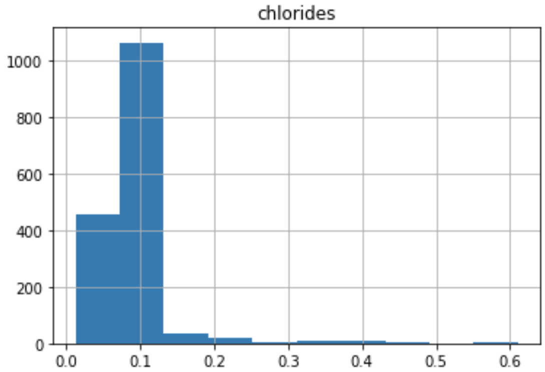


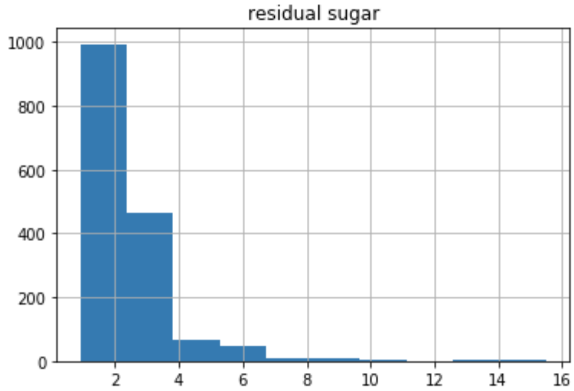
Figure 1: Cases in various classes is imbalanced. Example – It's less in 3,4 and 8

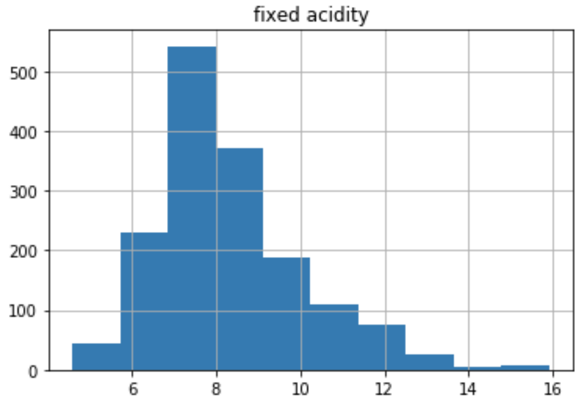
2) Algorithm and Feature Selection

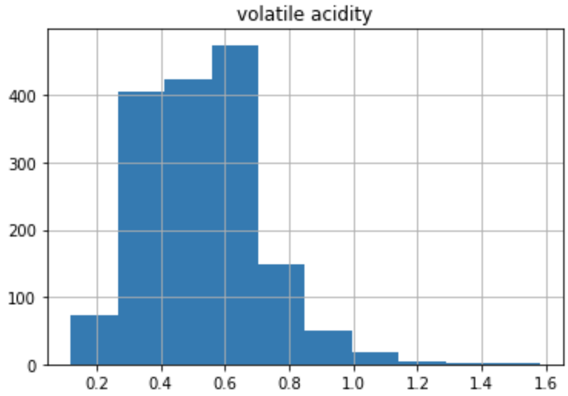
We started with studying histograms that show us the spread of each variable in the dataset.

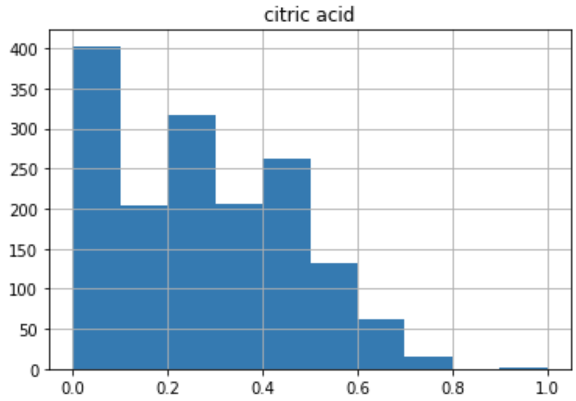


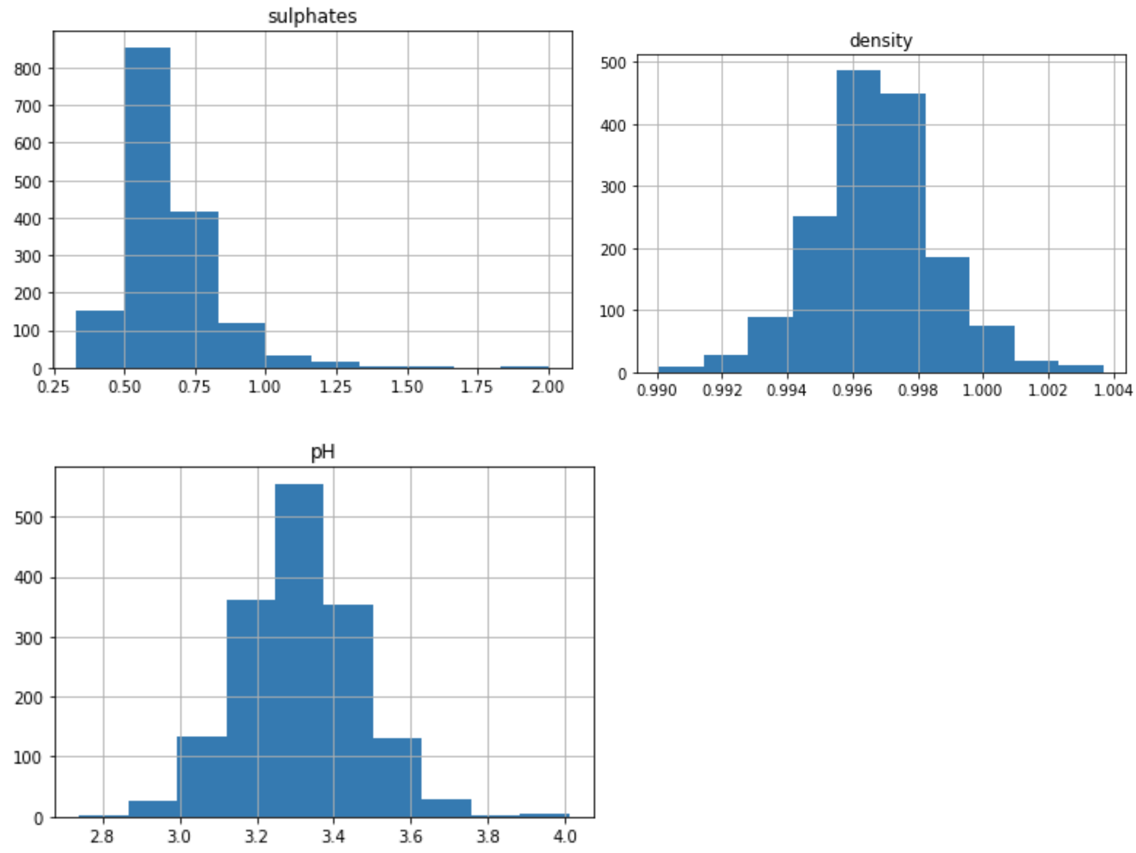












We then split our dataset into training and test data post which we applied feature selection method to our test features. There were some variables which we thought would be less relevant to our model. We tried to do feature elimination using RFE and the results are shown in the table below:

Feature	**Importance**
fixed acidity	0.07463742
volatile acidity	0.11723644
citric acid	0.07186342
residual sugar	0.07368053
Chlorides	0.08891028
Free sulphur dioxide	0.06679164
Total sulphur dioxide	0.10765634
density	0.07984014
pH	0.06908044

sulphates	0.10187993
alcohol	0.14842343

As per the table, we dropped a couple of features with least importance, but it didn't prove helpful as the accuracy of our model dropped significantly.

To counter check our approach, we also drew the co-relation matrix (See Figure 2).

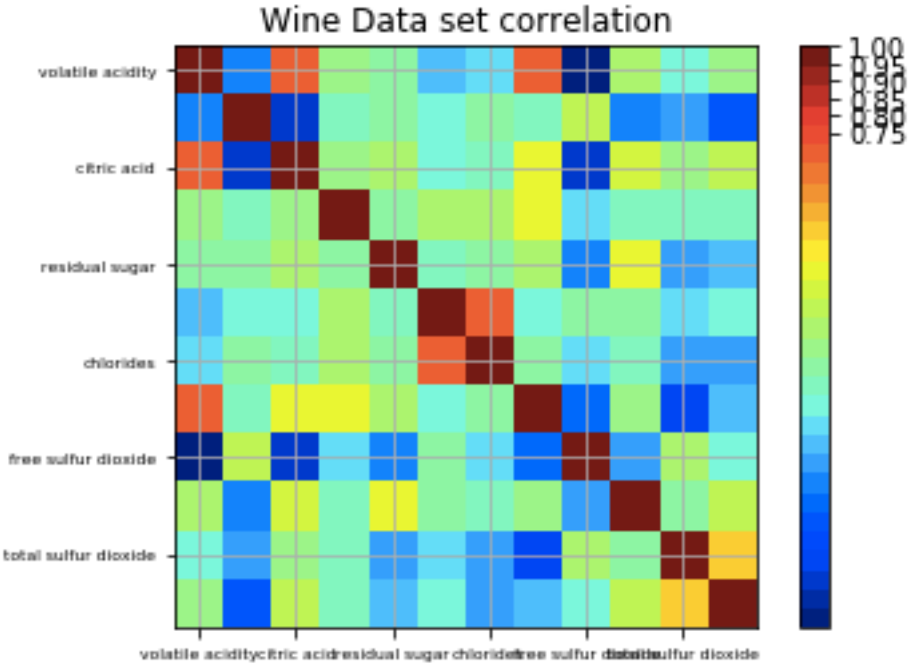


Figure 2

We selected the correlation threshold value of 0.8 for feature elimination. The correlation between variables came out to be very low(high correlation between citric acid and volatile acid which is evident from their chemical resemblance) so we decided to not drop the variables as dropping any variable was highly influencing our results.

2.1 Algorithms:

We preferred to use the Random Forest Algorithm over Support Vector Machine as Random Forest helped us classify into different classes whereas the working of SVM is that it gives us distance to the boundary of classes which we didn't desire in this. Random Forest also gave us better results in terms of accuracy.
We decided to compare our results of Random Forest model with the most common classification method i.e. Logistic Regression for benchmarking our current model.

Description of evaluation metrics used:

- **Accuracy_score**: It will give us the result of how precise our model was to predict a set of labels in our test sample after comparing it to actual value (test data).
- **Cohen_kappa_score**: This will provide us with a score that will define the degree of correlation between our set of annotators that are passed to this function. In this case it will produce a result in between target variable(test sample) and prediction of independent variables(test sample) using Random Forest and Logistic Regression. Kappa values can range from 0 to 1, where 1 means perfect agreement and 0 means No agreement. The values in between have derivative meanings such as 0.20 will mean slight agreement and 0.60 will be substantial agreement.
- **Cross_val_score:** This will divide our training set into 'n' parts of which 'n-1' would be used to train the data. It then runs a loop and returns an average value which describes the model.
- **Recall_score:** Returns ratio tp / (tp + fn) where tp is the number of true positives and fn the number of false negatives.

3) Evaluation:

We evaluate the efficiency of our model by using the above evaluation metrics and analysed the results. Due to large imbalance in the data, the results were not satisfactory. After up-sampling the data the efficiency of the model increased significantly. As the incidences in the data set was low, down-sampling would mean significant loss of data making the model in-efficient and unpredictive.

Below are results from before and after Upscaling of dataset.

	Metrics	Random Forest	Logistic Regression
Before Upscaling	accuracy_score	60.625	55.56
	cohen_kappa_score	0.360	0.25
	cross_val_score	61.087	-
	recall_score	32.60	32.03

	Metrics	Random Forest	Logistic Regression
After Upscaling	accuracy_score	83.74	52.49
	cohen_kappa_score	0.7921	0.36
	cross_val_score	84.65	-
	recall_score	84.37	39.7

It's clear that Upscaling our dataset, significantly increased accuracy score of our Random Forest model from 60.625% to 83.74%. The kappa score also increased from 0.36 to 0.79 which means there was agreement between our annotators more after Upscaling our dataset. In terms of recall score as well there is a commendable increase from 32.60 to 84.37 which again proves that better classification was done by Random Forest after upscaling the data.

However, when the results of our Logistic Regression model; both before and after upscaling didn't give us significant improvement as compared to Random Forest. The accuracy score before upscaling was 55.56 which decreased more after upscaling to 52.49. The Kappa score did increase by 0.11 after upscaling but it was not good enough. Similarly, the recall score increased slightly but still made us doubt our model.

4) Conclusion:

The analysis preformed on the wine data set can be concluded as follows:

1-The alcohol content of the wine is the most prevalent feature in determining the Quality of the wine. The analysis however clearly shows that there is no specific alcohol value for which provides the bias for the quality.

2- The distribution spread of chlorides and residual sugar is highly imbalanced.

Based on the results from both our models, Random Forest model was clearly better at predicting the Quality of wine. Since we did upscale the data to get improved results, it would've been better if there were more explanatory features. Given this dataset, it doesn't look like the chemical properties (features of this dataset) are enough to justify the quality of wine since at each quality level, variability of the predictors is high, and the groups are not well separated.