Using Random Forest to predict the red wine quality

Final Project Report for DATA 1030, Fall 2021 at Brown University

Supervised by Prof. Andras Zsom

https://github.com/kelsier-wang/data-1030-project.git

Xufan Wang

1. Introduction

Red wine is very normal in human being's daily life. Nowadays, red wine not only represents elegance but also can improve human being's heart. As a result, more and more wine products are produced and sold with the help of high-speed developed information technology. However, it is hard to determine wine quality manually. There is no standard rule, and so the quality is heavily based on experts' own flavor and experience. Such manually way to determine wine quality is both inaccurate and time-cost. So, using a model to predict the wine quality is useful and meaningful for most of the consumers.

This project attempts to create a diagnostic tool that will leverage machine learning to classify a wine's quality. The dataset used for this project came from the UCI Machine Learning repository. This dataset is related to red variants of the Portuguese "Vinho Verde" wine (P. Cortez, A. Cerdeira, F. Almeida, T. Matos and J. Reis.). It contains 11 characteristics of red variants, altogether 1599 datapoints which represent the Portuguese "Vinho Verde" wine. The target variable is the quality of the wine. Data is well-documented. The features of red wine are based on physicochemical tests and the target variable is based on sensory data.

The features are all numerical and continuous, include:
(From the data website, there is no units for each feature)

1. Alcohol: the amount of alcohol in wine
2. Volatile acidity: are high acetic acid in wine which leads to an unpleasant vinegar taste
3. Sulphates: a wine additive that contributes to SO2 levels and acts as an antimicrobial and antioxidant
4. Citric Acid: acts as a preservative to increase acidity (small quantities add freshness and flavor to wines)
5. Total Sulfur Dioxide: is the amount of free + bound forms of SO2
6. Density: sweeter wines have a higher density
7. Chlorides: the amount of salt in the wine
8. Fixed acidity: are non-volatile acids that do not evaporate readily
9. pH: the level of acidity
10. Free Sulfur Dioxide: it prevents microbial growth and the oxidation of wine
11. Residual sugar: is the amount of sugar remaining after fermentation stops.

Some public examples include a project of Dexter Nguyen from Duke University. He solves the ML question about which features are the best quality red wine indicators by using Regression. Based

on three metrics (R-squared, RMSE, and MAE), he found that Random Forest-based feature sets model performed best and determined the most influential features are volatile acidity, citric acid, sulphates, and alcohol.

Another example is from Nivyasree Avula who tested several models such like “Normalization” and “Neural Networks” to check the accuracy of the wine. And finally, he found that using the MinMaxScalar will give a higher accuracy which is 70%.

From these examples, I’d like to use different classification methods to see which yields the highest accuracy for the quality of wine and determine which feature is the most indicative of a good quality wine.

2. Exploratory Data Analysis

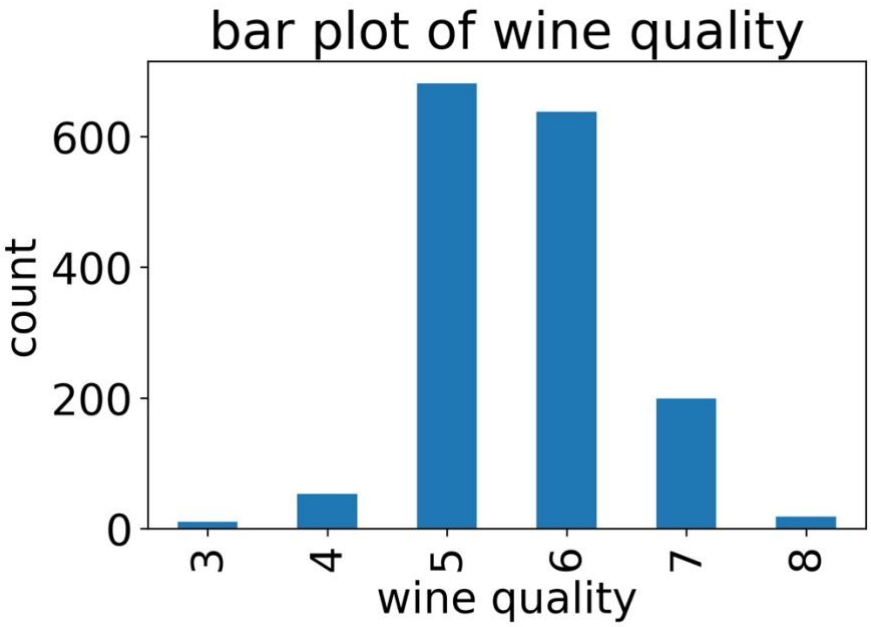

Figure 1: This figure gives an outlook of how the target feature distributed. Using bar graph to show the number of results for each wine quality. From the graph, it shows that most of the results are in wine quality (5,6) and very little in (8,3). Also, the distribution is approximately Normal distributed, centered around 5 and 6. Need to consider about the data which is imbalanced.

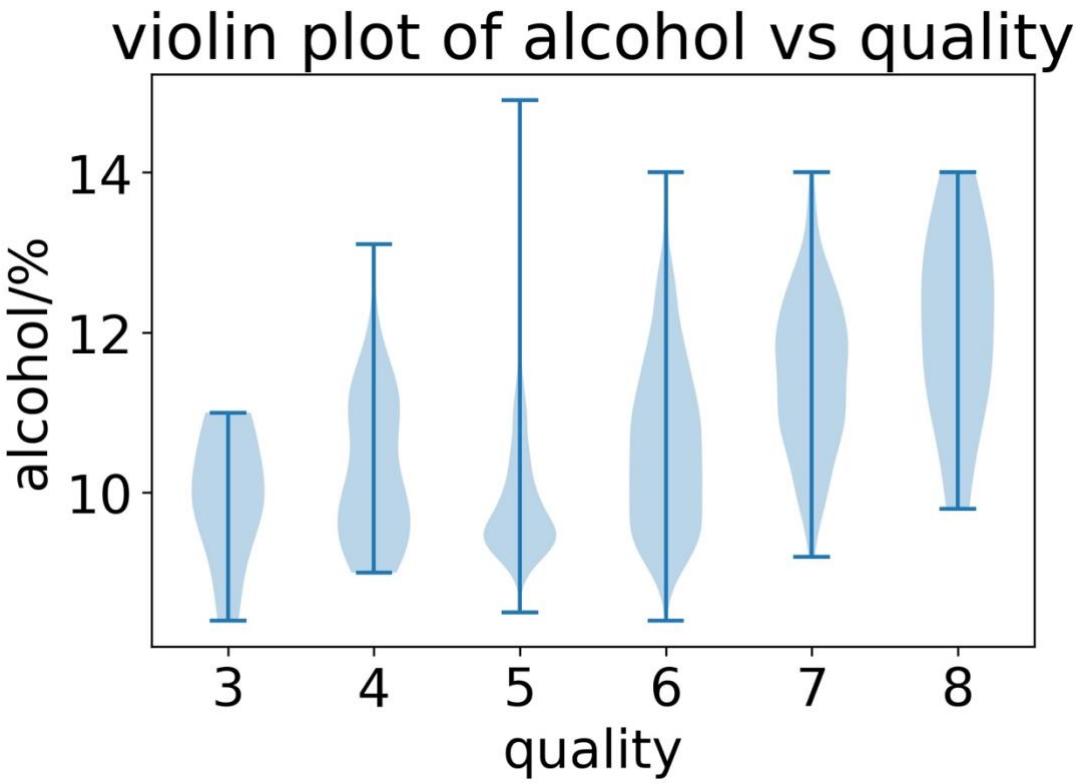

Figure 2 This figure displays the kernel density plots for the average quality of the wine of a given wine's feature. From the graph, it is evident that the higher quality wine has, the higher alcohol mean it has. Quality 5 is noticeable that has a high variance but low mean which is not very normal. At the same time, from the shapes of the violin plots, we can see except for quality 4 and 5, within the range the alcohol variability is roughly even but tapers at either end.

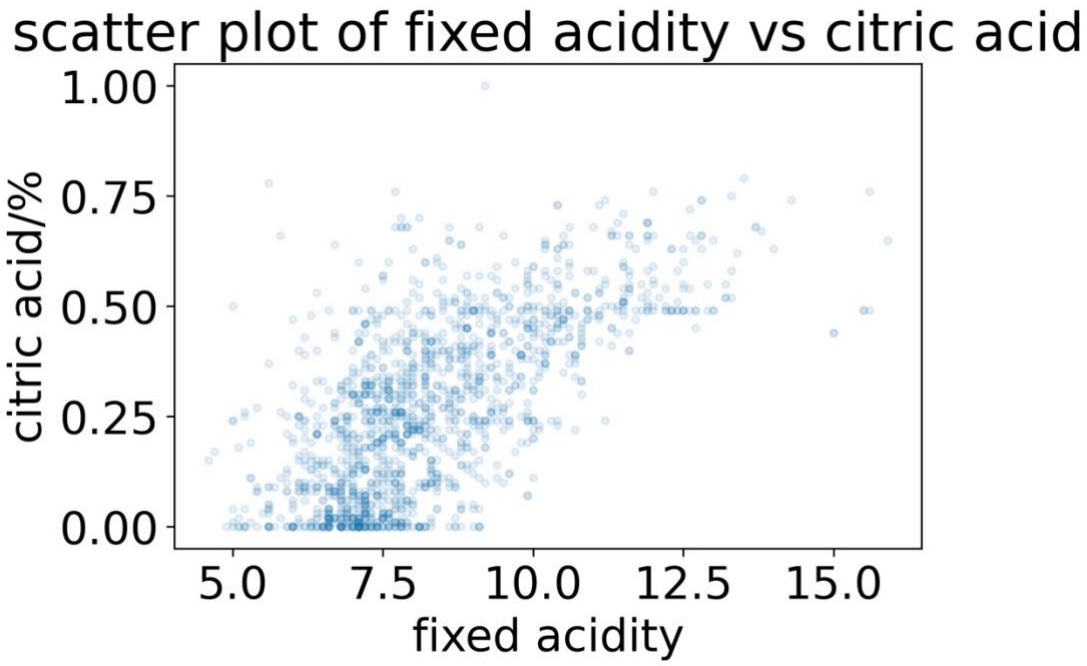

Figure 3 This figure displays the relationship between citric acid and fixed acidity by scatter plot.

It shows the strong correlation between these two features. As fixed acidity increases, citric acid also increases. And most of the wine has fixed acidity between 5.0 and 10.0; most of the wine has citric acid between 0.0 to 0.5.

3. Methods

3.1 Data Splitting and Preprocessing

Since all the features are one of the characteristics of a unique type/brand of wine, the data is iid and doesn't have a group structure or time-series.

In the data preprocessing step, since the datasets is very small with only 1599 data, and it is also imbalanced data: only a small fraction of the points is in classes (3, 8). That's why I choose StratifiedKfold to split the dataset. the splitting step allocated 20% of the observations to testing, and the other 80% to 5 folds cross-validation. In each instance of cross-validation, the preprocessor fit and transformed the four training folds before the transforming validation and testing sets. The model trained using a cross-validation split to account for the small number of observations in the set and the variability between random splits that may occur.

There are altogether 11 features in the preprocessed data. And since they are all continuous data, and follow a tailed distribution, the preprocessor applied the StandardScaler for each feature. As a result, the final preprocessed dataset doesn't have change and has 11 features.

3.2 Model Selection

Using the splitting and preprocessing methodology, six different machine learning models were trained and compared: a logistic regression model (which is represented as the baseline), a logistic regression model with l1 regularization, a logistic regression model with l2 regularization, a logistic regression model with an elastic net, a Random Forest classifier, a Support Vector Machine classifier, and a K-nearest neighbors classifier. All models were hyperparameter tuned. The parameter grid for each of these methods are determined by the best hyperparameters in each run. At last, the best hyperparameters should not be on the boundary of the parameter grid, and then it can be said that overfitting and underfitting scenario has been considered. Accuracy was used as the evaluation metric because this is a classification problem, and it is cheap to act compared with other metrics such like F-1 score. This process was repeated 10 times for 10 different random states, and 10 best models and the 10 test scores were returned. Below are the parameters tuned and values tried for each model:

Model	Hyper-Parameter	Search Space	Optimal Value
L1	C: Inverse of the regularization strength	[1e-4, 1e4]	1e-1
RF	Max_depth Max_features	[1,3,10,30,100] [0.1, 0.2, 0.3, 0.5, 0.75, 1.0]	30 0.2
SVC	Gamma: Kernel coefficient C: Inverse of the regularization strength	[1e-2,1e2] (-1, 1, 5)	3.162 0.1
KNN	N_neighbors Weights	[1,2,3,5,10,30,100,200] ['Uniform','distance']	100 distance

Figure 4 Parameters used for tuning of each model

After tuning, each grid search's best model parameters were extracted and used for comparison on Accuracy on a holdout set. Below are the average accuracy scores for the best of each model across the ten random states:

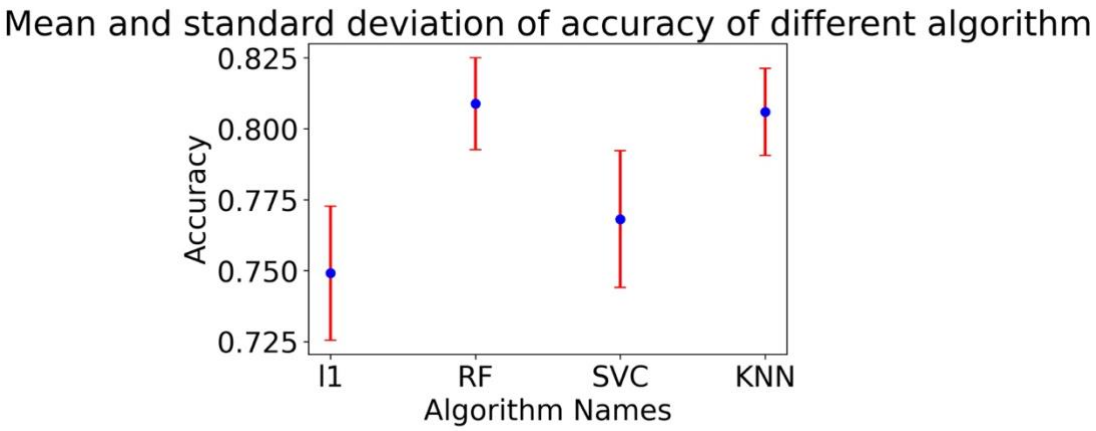

Figure 5 Average RMSE scores for the best model over ten random data splits

The Random Forest model has the best performance on the test level and so was chosen as the model of choice. For each random state, the best value of the random forest model was checked to get the best parameter.

4. Results

4.1 Evaluation of Models

Based on the results above, I choose the best model and hyperparameter of random forest model which is 30 for max_depth and 0.2 for max_features. The model was retrained on new splits over new random states 100 times. For each split, 80% of the data was allocated to training and 20% was allocated to testing. For each random state, the test accuracy score, baseline accuracy score and model were recorded.

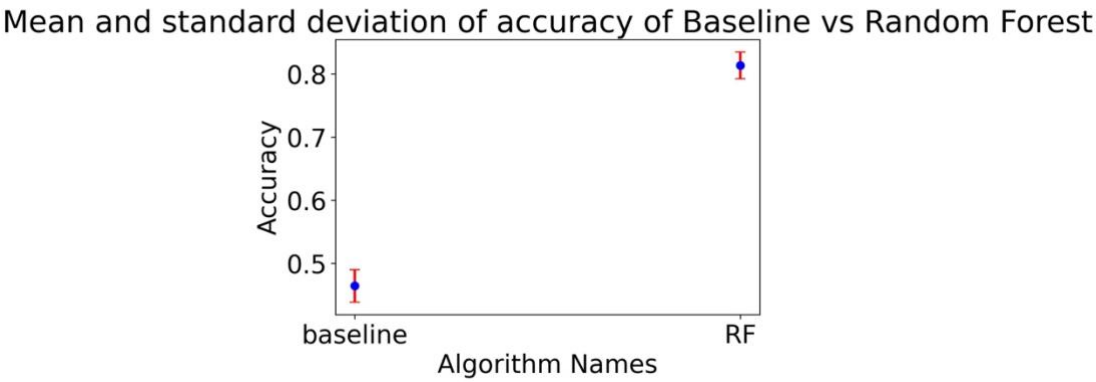

Figure 6 Best RF model Vs Baseline

The baseline model returns an average accuracy 0.464 with a standard deviation 0.026. In comparison, the random forest model returns an average accuracy 0.813 with a standard deviation

0.021. The random forest model achieves an accuracy that is 13 standard deviations above baseline. Similarly, the baseline model's accuracy was 16 standard deviations below the average of the trained models.

4.2 SHAP

SHAP is one way to calculate local feature importance which is based on Shapely values from game theory. The Shapely values are computed for each point in the demonstrations. Figure 6 compares Shapely values for each feature which shows the local feature importance. From the graph, we can see that the alcohol has the greatest affect, sulphates is the second one, and the volatile acidity is the third one.

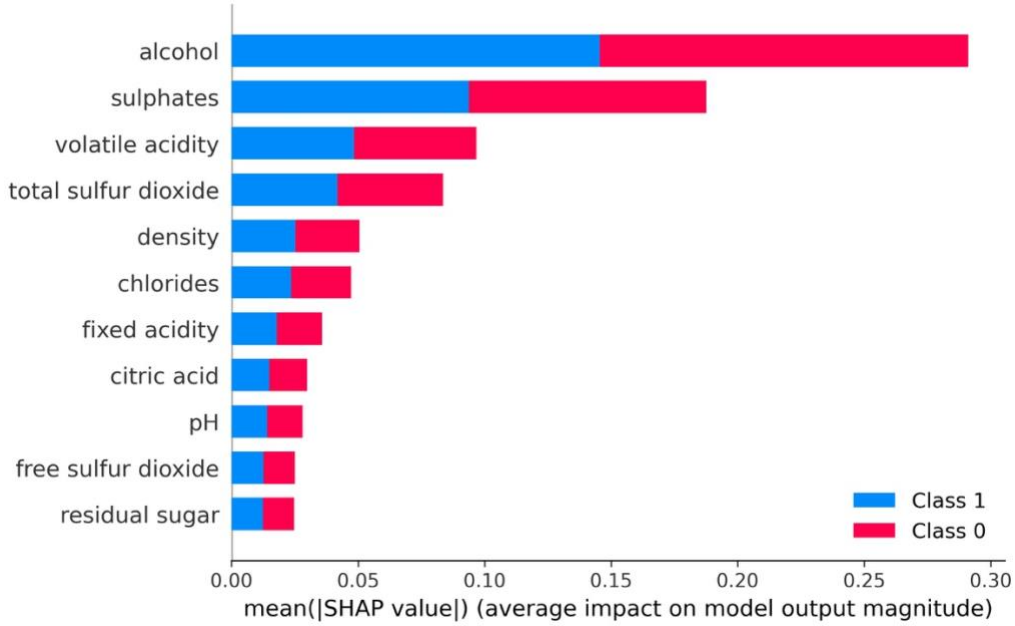

Figure 6 SHAP value

4.3 Global Feature Importance

By performing three different global feature importance methods: permutation feature importance calculation, mean decrease in impurity, and the Random Forest Feature Importance, the importance of each feature is shown.

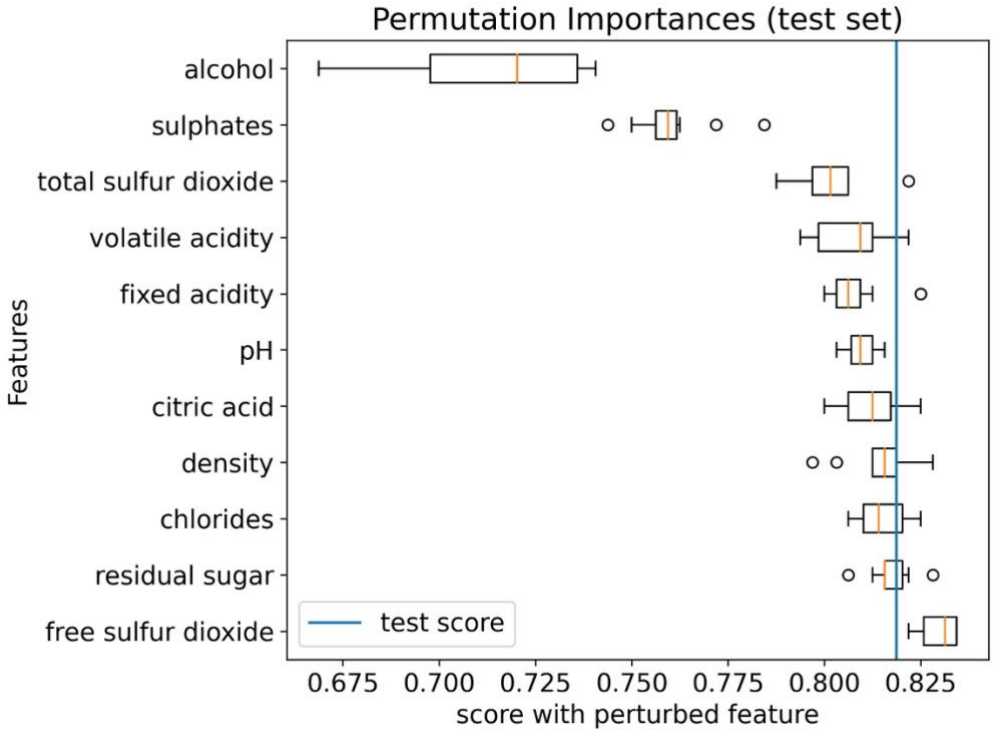

Figure 7 Permutation importance

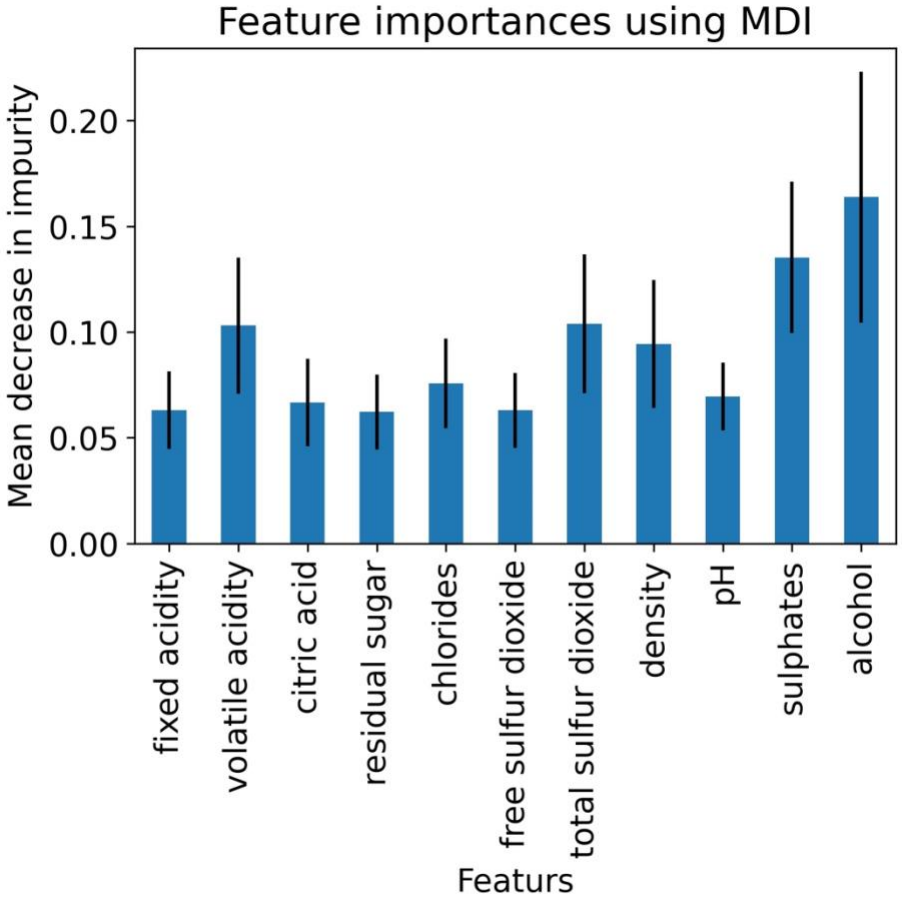

Figure 8 MDI Importance

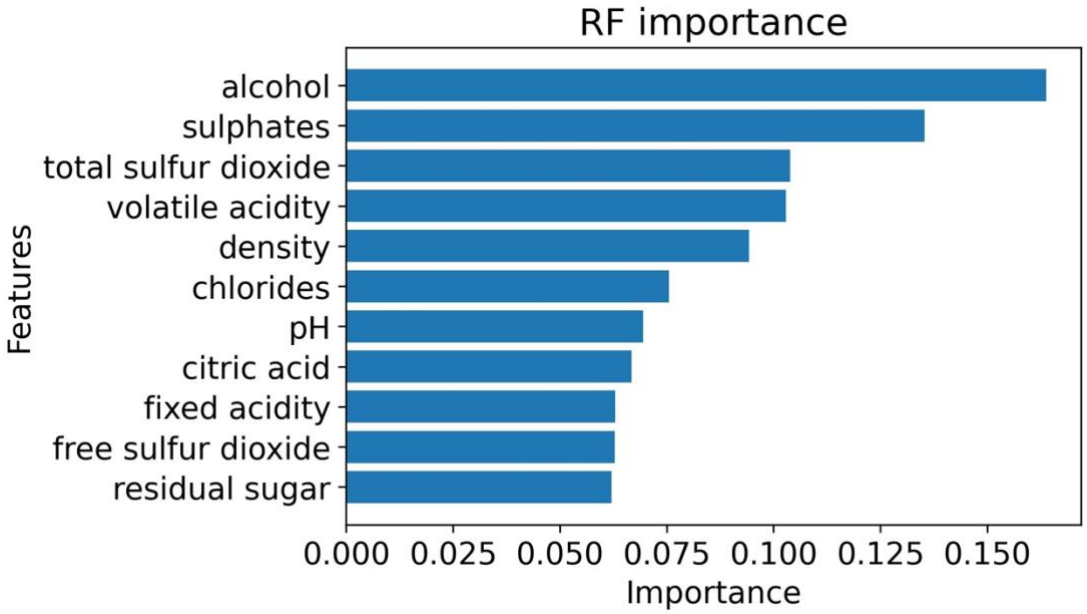

Figure 9 RF Importance

Because of the relationship between feature and target is broken by the shuffle, model score worsens. From the figures above, we can see that the top two which causes the most difference is alcohol, and sulphates. And volatile acidity and total sulfur dioxide has the similar effect behind alcohol and sulphates. And very surprisingly, in permutation, when we shuffle the citric acid data, the test score even is improved.

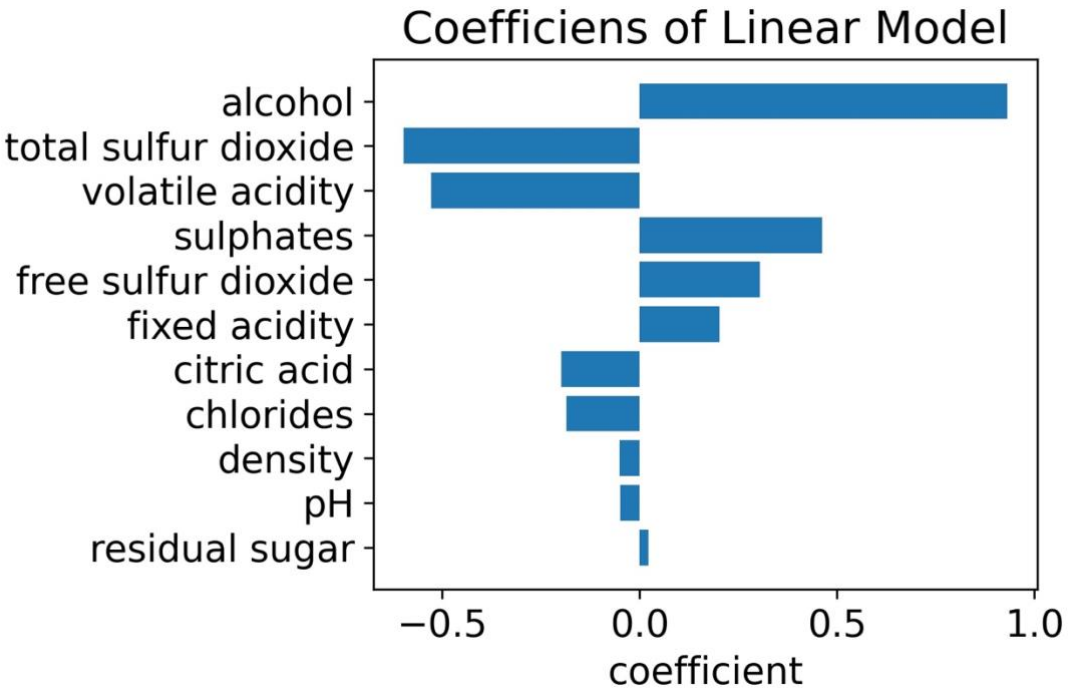

Figure 10 Coefficients of Linear Model

There is another way to find the global feature importance which is to change the coefficients of the linear models. But since it is only worked for linear models, and here random forest model gives the best result, the results of the method to find the global feature importance doesn't have strong accuracy. From the graph, we can see the top two important features are alcohol and total sulfur dioxide which has a difference from the methods above.

5. Outlook

There are two main weakness for this analysis. The first one is the unbalanced data. The main data of quality is dominant around 5 and 6, which had little contribution to create an optimal model. And there is sparsity of samples for the really low quality and high quality wines, so we don't get to really see what makes a good quality wine. The unbalanced data made it hard to identify other quality wine. To improve this part of the model, more balanced data is required (other quality wine data). And another problem is the data set has too few features although they are all being very descriptive. We could explore more features and develop better sense of a prior for which feature might have higher correlation, such like the location of the wine, the type of the wine, the harvest season, etc. For better prediction, we can use some techniques to combine different models such like bagging or boosting, some PCA or exploring different methods for feature selection which can make the model be more efficient and accurate.

Reference

https://www.kaggle.com/uciml/red-wine-quality-cortez-et-al-2009
https://towardsdatascience.com/red-wine-quality-prediction-using-regression-modeling-and-machine-learning-7a3e2c3e1f46
https://medium.com/analytics-vidhya/predicting-red-wine-quality-using-machine-learning-model-34e2b1b8d498

Github link

https://github.com/kelsier-wang/data-1030-project.git