Wine Quality Classification

Peyton Gillenwater, Shuai Huang, Ruoyang Qin, Culley Wilson

May 14, 2021

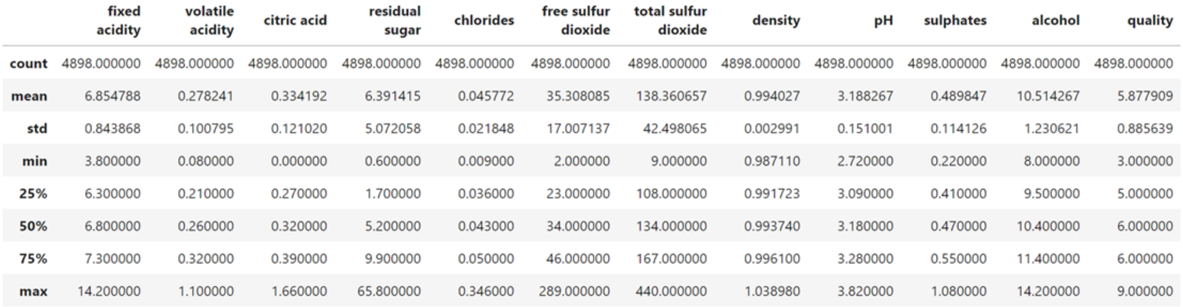
1 Introduction

We chose to analyze wine quality data from the UCI database. (The source data can be found here). This project deals with a classification problem, mainly focusing on classifying the wine quality as good or bad. 11 attributes were used to do the prediction. In this project, we compared several different methods, including regression classification models.

2 Data Description

2.1 Variable Analysis

As in Figure 1, the number of good quality wines is close to the number of bad quality wines. The 11 predictors are all continuous, and there are no missing values in the data set. To detect the extreme values, we calculated the mean, standard error, 25%, 50%, and 75% percentiles, minimum and maximum. We also drew a histogram for each variable, which is not displayed here. Most of the values are close to the mean.



	fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality
count	4898.000000	4898.000000	4898.000000	4898.000000	4898.000000	4898.000000	4898.000000	4898.000000	4898.000000	4898.000000	4898.000000	4898.000000
mean	6.854788	0.278241	0.334192	6.391415	0.045772	35.308085	138.360657	0.994027	3.188267	0.489847	10.514267	5.877909
std	0.843868	0.100795	0.121020	5.072058	0.021848	17.007137	42.498065	0.002991	0.151001	0.114126	1.230621	0.885639
min	3.800000	0.080000	0.000000	0.600000	0.009000	2.000000	9.000000	0.987110	2.720000	0.220000	8.000000	3.000000
25%	6.300000	0.210000	0.270000	1.700000	0.036000	23.000000	108.000000	0.991723	3.090000	0.410000	9.500000	5.000000
50%	6.800000	0.260000	0.320000	5.200000	0.043000	34.000000	134.000000	0.993740	3.180000	0.470000	10.400000	6.000000
75%	7.300000	0.320000	0.390000	9.900000	0.050000	46.000000	167.000000	0.996100	3.280000	0.550000	11.400000	6.000000
max	14.200000	1.100000	1.660000	65.800000	0.346000	289.000000	440.000000	1.038980	3.820000	1.080000	14.200000	9.000000

Figure 1: Basic Statistics of All Variables.

2.2 Correlation Analysis

We called a covariance matrix (as in Figure 2), violin plots and hive plots (as in Figure 3)to see the relationship between variables. From the Covariance Matrix, density, and residual, sugar has a strong positive relationship, and there is a very strong negative relationship between alcohol and density. Violin plots or hive plots can help us detect the relationship between a single predictor variable and the response variable, one of which is displayed here. From the plot, we see a positive relationship between quality and alcohol.

3 Model Selection

3.1 Criteria for Evaluating Models

When evaluating regression models, we calculated the accuracy on the train data set and test data set. 10-fold cross-validation and F1 score were used and 5-fold cross-validation was used to evaluate

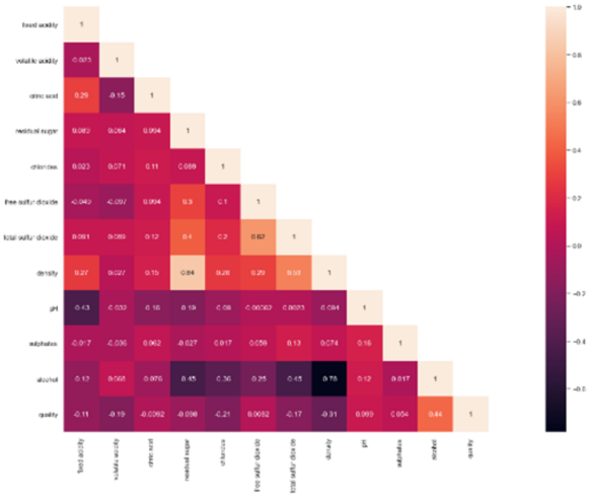


Figure 2: Covariance Matrix

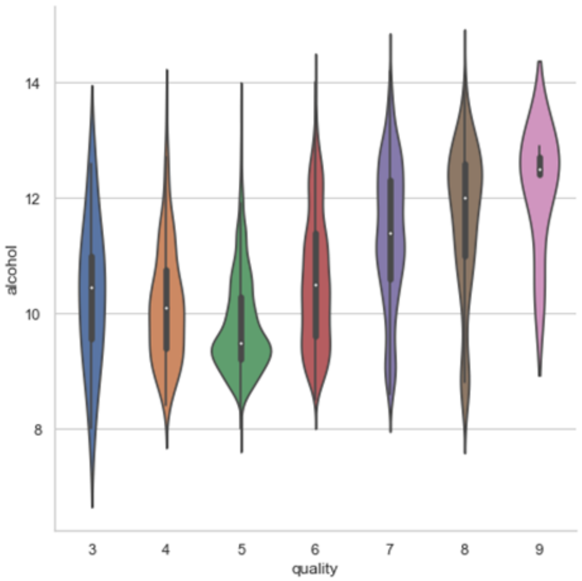


Figure 3: Violin Plot for Alcohol and Quality

other models.

3.2 Models to be Compared

There are two kinds of models used in our project, regression models and classification models. When one of the regression models was performed, we first calculated the value of quality and then used that value to do the classification. Regression models we used include Multiple Linear Regression Model, Ridge Regression Model, and Lasso Regression Model. The classification models we performed include Logistic Regression model, Naïve Bayes, k-Nearest Neighbors Algorithm (KNN), Linear Discriminant Analysis (LDA), Quadratic Discriminant Analysis (QDA), Decision Trees, Bagging, Boosting, Random Forest, Linear Support Vector Machine, Radial Basis Function Kernel SVM, Neural Network with 5-layer perceptron and Neural Network with 10-layer perceptron (MLP=5 and MLP=10).

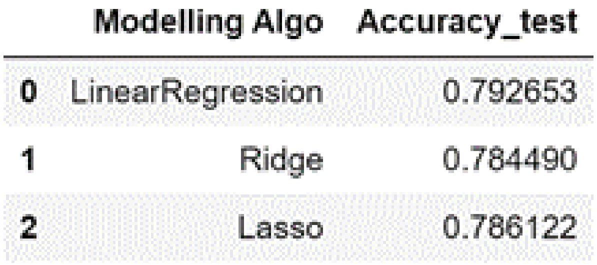
3.3 Model Comparisons

1. Regression Models

   Three linear regression models were used and the accuracy on the test data set were calculated and displayed below in Figure 4. The performances of all three models are not bad, with the accuracy on the test data set close to 0.80. This is probably because the continuous variables were used to train the regression model instead of categorical variables. We can also notice that the Ridge Regression Model and Lasso Regression Model which are methods of regularization would not do much help to improve the classification, which is reasonable since we use the models to do classification. Thus, unless there are big differences, whether or not we use regularization methods does not affect the results.

2. Classification Models

   Different evaluating criteria might produce different results, especially the difference between the train data set and the test data set. Therefore, it's necessary for us to refer to k-fold cross-validation to evaluate the models. As shown in Figure 5, after we conducted the classification models we listed in the previous section, we discovered that the performance of Linear Support Vector Machine is apparently worse than other methods, which is probably because it's hard to find a hyperplane to separate the data. Some improved methods based on Decision Tree, like Bagging, Boosting, and Random Forest, are promising approaches.



	Modelling Algo	Accuracy_test
0	LinearRegression	0.792653
1	Ridge	0.784490
2	Lasso	0.786122

Figure 4: Accuracy of Regression Models

(a) Naive Bayes:
We know from the covariance matrix that some predictors are correlated, which means the data violates the assumption that predictors are independent of each other. Mainly, for this reason, the performance of the Naïve Bayes Model was not satisfying.

(b) K-Nearest Neighbors Algorithm (KNN):
When k is small, the model cannot make good predictions, although the accuracy on the train data set is high. When the k is larger, the model performs better. This is probably because of the overfitting problem when k is small. However, if k is too large, something called the curse of dimensionality would happen, leading to a very poor prediction and hence a poor fit.

(c) Discriminant Analysis:
Apparently, Linear Discriminant Analysis is better than Quadratic Discriminant Analysis. The Quadratic Discriminant Non-linear Model could have the problem of curse of dimensionality like KNN when the observations are not large enough.

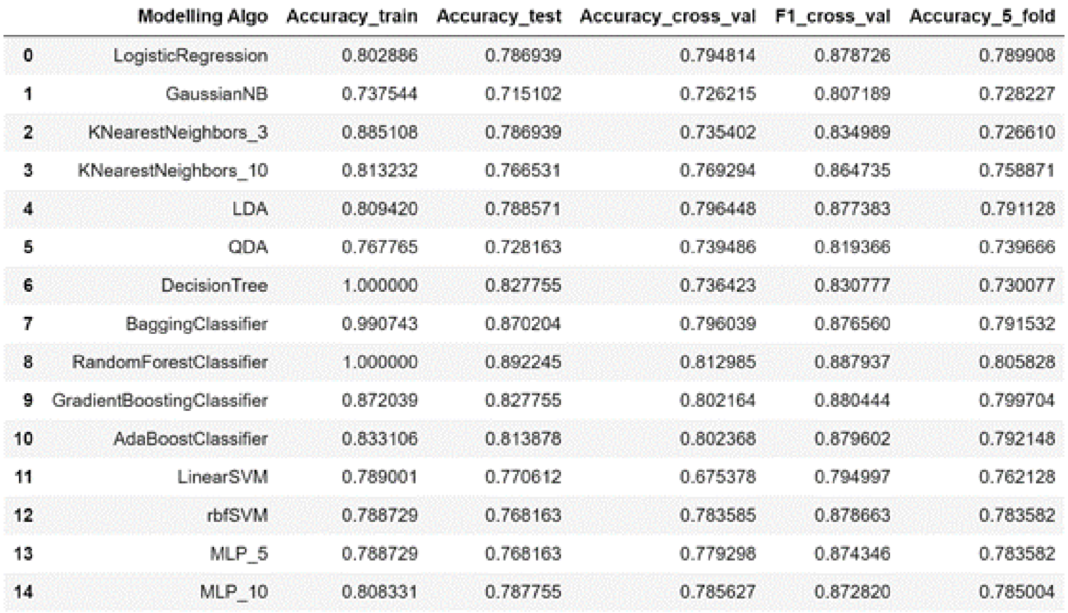
(d) Tree-Based Models:
All the tree-based methods we used in this analysis perform well. However, we discovered that the accuracy of the Decision Tree Model on the train data set was 100%, but the accuracy on the test data set was low, which implied that the model might be overfitting and we need to do tree pruning, which we will discuss later. Other tree-based methods are doing regularizations on the Decision Tree. We can see that although the accuracies on the train data set become worse, the performance on the test data set become better.

(e) Support Vector Machine:
As we stated above, for this data set, it's probably not easy to find a hyperplane, thus the performance of Linear Support Vector Machine is worse than Radial Basis Function Kernel SVM (RBFSVM), especially on high dimensional data.

(f) Neural Network with Multiple-layer Perceptron:
We applied Neural Networks, with different perceptron layers. Finally, we discover that the predictive ability will be the best when the perceptron layers are between 5 and 10. When it's lower than 5, the estimate will be poor, but when the layers are too larger, it can lead to overfitting.

3.4 Model Improvement

As we discussed above, the Decision Tree was overfitting, therefore we did tree pruning on the original tree to find the optimal pruning parameter to select a subtree that leads to the lowest test error rate. We call the GridSearchCV method to help us automatically find the optimal parameter from the parameter list. Cross-validation was used to implement this method. With the help of this automatic searching method, we found that the max depth would be 10, the max leaf nodes would be 100 and the minimum samples leaf would be 1. The accuracy on the train data set becomes lower, but the cross-validation accuracy on the test data set has indeed improved.



	Modelling Algo	Accuracy_train	Accuracy_test	Accuracy_cross_val	F1_cross_val	Accuracy_5_fold
0	LogisticRegression	0.802886	0.786939	0.794814	0.878726	0.789908
1	GaussianNB	0.737544	0.715102	0.726215	0.807189	0.728227
2	KNearestNeighbors_3	0.885108	0.786939	0.735402	0.834989	0.726610
3	KNearestNeighbors_10	0.813232	0.766531	0.769294	0.864735	0.758871
4	LDA	0.809420	0.788571	0.796448	0.877383	0.791128
5	QDA	0.767765	0.728163	0.739486	0.819366	0.739666
6	DecisionTree	1.000000	0.827755	0.736423	0.830777	0.730077
7	BaggingClassifier	0.990743	0.870204	0.796039	0.876560	0.791532
8	RandomForestClassifier	1.000000	0.892245	0.812985	0.887937	0.805828
9	GradientBoostingClassifier	0.872039	0.827755	0.802164	0.880444	0.799704
10	AdaBoostClassifier	0.833106	0.813878	0.802368	0.879602	0.792148
11	LinearSVM	0.789001	0.770612	0.675378	0.794997	0.762128
12	rbfSVM	0.788729	0.768163	0.783585	0.878663	0.783582
13	MLP_5	0.788729	0.768163	0.779298	0.874346	0.783582
14	MLP_10	0.808331	0.787755	0.785627	0.872820	0.785004

Figure 5: Model Comparisons

4 Results Visualization

In order to show the Decision Tree has improved, we used the ROC curve (Figure 6) and the Confusion Matrix (Figure 7) to visualize our result. From the ROC curve, we discover that the area under the curve is larger, which means the prediction is better. The confusion matrix shows that the true positive rate is larger than the false positive and false negative rate, which indicates that the prediction is good.

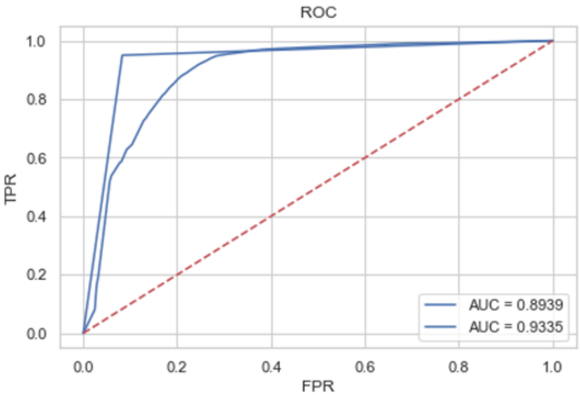


Figure 6: ROC curves

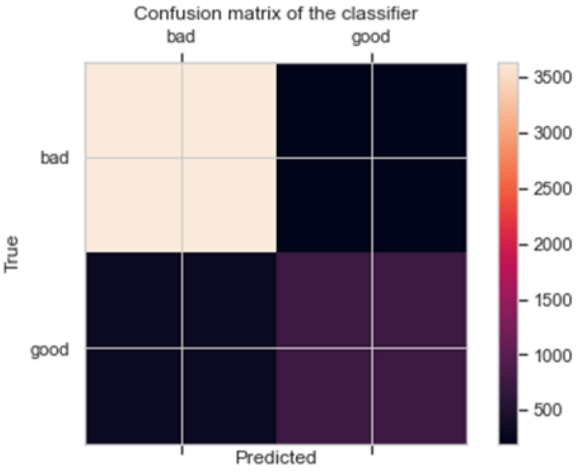


Figure 7: Confusion Matrix