Red Wine Quality Prediction

STAT462 Final Report

Team: 5

Jiahong Lu	jfl5651@psu.edu
Yiqi Xiong	yfx5079@psu.edu
Zhongyuan Hua	zzh31@psu.edu
Kexin Yi	kfy5053@psu.edu
Keqian Zhang	kxz5115@psu.edu

Abstract

Dataset

Wine Quality Data Set (red wine only)

Linear regression and machine learning are two emerging areas of research. Many algorithms are used to analyze wine quality or class. The quality of wine is not only determined by alcohol but also determined by other 10 various chemicals. This report is aimed to explore how each chemical component influences the quality of wine by using linear regression and machine learning. Firstly, data processing is applied i.e. remove the duplicated data and fill the missing data. Secondly, moving to exploratory data analysis (EDA) which helps us know the influential factor of wine quality better. Then, linear regression model, Logistic model and Random Forest model are performed to predict test data individually. The accuracy of the linear model is not great, while the Logistic model outputs 73.52% accuracy and the Random Forest model has 90.6% accuracy.

Keywords: *Data cleaning in preprocessing, collinearity, plot and distribution in EDA, linear regression, logistic regression model, random forest model, prediction, analysis, accuracy.*

Introduction

In recent years, technology plays a significant role in the industrial field, and helps increase quality and productivity. Product quality is crucial for the food industry, which is not only good for people's health, but also easy for the sales promotion.(Gupta, 2018) However, food quality assurance is a time-consuming and expensive task, since food processing is complicated and will change over time, which means we do not know the final result until all processing steps finish. Therefore, people introduce several technological methods to predict food quality. This project targets red wine quality prediction. Since wine drinking has been proven to have a protective effect on heart diseases, increasing wine consumption brings the problem of how to produce good quality wine within a shorter time at lower cost. The data set, provided by UCI machine learning repository, is related to red variants of the Portuguese "Vinho Verde" wine.(Cortez. et al, 2009) In the data set, there are 1599 red wine sample observations and 13 variables. The "quality" is output variable, which listed from 0 to 10 corresponding with bad to excellent; the input variables include fixed acidity, volatile acidity, citric acid, residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide, density, pH, sulphates, and alcohol. Despite the "quality" variable, the rest of variables are associated with one key physicochemical characteristic that is important for red wine quality. We would like to perform linear regression model and logistic model to determine dependency of wine quality on other variables and in wine quality predictions. By examining the importance of each chemical parameters in the wine, we would be able to know the accuracy of the models that implement prediction in this project and the method to improve wine quality.

Methodology

Data Cleaning

We have checked duplicate and missing data. After we remove all repeated data, we leave the 1359 data. According to this data, we make an analysed report. We plot the Fixed acidity, Volatile Acidity, Citric Acid, Residual Sugar, Chlorides, Density, pH, Sulphates, Alcohol variable on histograms and on boxplots. The histograms and boxplots (Plot 1) show most of the variables skewed to the right, in which we will need to apply transformation and examine potential outliers. We will perform them in the model fitting section. We also split the dataset into testing set and training 25-75 respectively. So the following models will perform predictions based on training set and compare accuracy based on testing set.

Exploratory Data Analysis

Since the goal of this project is to find the factors that may influence the quality of red wine, the quality is treated as the response (dependent) variable and other variables as independent variables. Before fitting models, we use the scatterplots to make a preliminary analysis in order to find out which variables seem to potentially predict the quality of the red wine. To make the scatterplots clean and easier to read, we separate the variables based on categories. The acidity category includes the fixed acidity, volatile acidity, citric acidity, and the pH value. The element category includes the residual sugar, chlorides, sulphates, alcohol, and quality. The SO2 category contains the free sulfur dioxide and the total sulfur dioxide. The last category is the density. From these scatterplots, there is no obvious linear relationship between quality of red wine and any of the variables, but the plots do reveal that there may be collinearity issues within each category of the variables.

To identify the collinearity between variables, we use both the function “ggcorr” and “ggduo” in R to plot the correlation between every pair of independent variables as well as the correlation between every variable with the response variable quality. From the correlation check, it reveals that alcohol and volatile acidity are kind of strongly correlated with the quality of wine compared with the other variables. At the same time, it also reveals some collinearity between several groups of independent variables. The variables that are strongly correlated (>=0.6) to each other includes the density and fixed acidity, fixed acidity and citric acidity, fixed acidity and pH, and total sulfur dioxide and free sulfur dioxide.

The next step of our exploratory data analysis is to plot all the variables to view their distributions. From the histograms, they show that almost all of the variables are not normally distributed. In order to quantify the skewness of each of the variables, we apply the “skewness” function. To interpret the results, there is a rule to follow. If the value of skewness is 0, the data are perfectly symmetrical. If the value of skewness is less than -1 or greater than 1, the distribution is highly skewed. If the value of skewness is between -0.5 and 0.5, the distribution is approximately symmetric. Based on the histograms and the rule for skewness, the approximately normally distributed variables are: citric acid, density, and pH value. However, if we further use the Shapiro test, they are still skewed since the P-value is less than 0.05.

The final part we have examined is the response variable quality of red wine. For the response variable, we want to determine whether it is normally distributed. After plotting a histogram and performing the Shapiro test, we conclude that the response variable is not normally distributed.

Linear Regression

To start with modeling fitting, since we examine the skewness of data and outliers in the previous section, we decide to apply Box-Cox transformation in order to transform our dataset into a normal shape. When we check the boxplots, there exist several outliers in these variables, and we need to

delete outliers, so that we can eliminate their violation toward regression and analysis of variance (ANOVA) analysis. We delete sample values with more than 3 standard deviations. After processing previous procedures, we fit a full model into the training set with quality as response and all other variables as predictors to determine several potentially useful subsets of explanatory variables. As we mentioned before, there exists collinearity issue, and it has been proven that when we check the variance inflation factor (VIF) of variables.(Plot 2) We also check the assumption for the linear regression analysis, based on the (Plot 3), the assumption for linearity, independence, and equal variance are satisfied. The normality assumption is also satisfied from the Quantile-Quantile (QQ) plot while the data do not necessarily follow the normal distribution from the Shapiro-Wilks tests. We drop the insignificant variables, and fit the linear regression model again, then check the VIF and ANOVA result to make sure the new model is significant. We also would like to improve the Multiple R-squared (R^2), so we remove influential points during the model diagnostic. Moreover, since we are interested in interactions between these predictors. We fit an interaction regression based on significant predictors and significant interaction between them. R^2 and normalized root mean square error (NRMSE) are used to determine the accuracy of prediction.

Analysis

From the summary of the full linear model, the p-value of F test is less than significant levels 0.05, so we do reject the null hypothesis and conclude that the coefficients of these predictors are not all zero. From the p value of t test, we can get that the significant variables are volatile.acidity, chlorides, total.sulfur.dioxide, sulphates and alcohol. pH is also significant but not as significant as the 5 predictors above. In order to make all predictors significant to model, we drop those predictors with p value larger than 0.05.

$$\text{quality} = \beta_0 + \beta_1 \text{volatile.acidity} + \beta_2 \text{ chlorides} +$$
$$\beta_3 \text{ total.sulfur.dioxide} + \beta_4 \text{sulphates} + \beta_5 \text{alcohol} + \beta_6 \text{pH}$$
$$\beta_0 = 4.1133296;\ \beta_1 = -1.0333054;\ \beta_2 = -1.6945589;$$
$$\beta_3 = -0.0021878;\ \beta_4 = 0.8548460;$$
$$\beta_5 = 0.3065429;\ \beta_6 = -0.441403$$

In this new model, according to the rule of thumb, since the values of the Cook's distance are all below 1, then we don't have outliers. But there are still many points sticking out compared to the other points, so we need more investigations. From the VIF output, there is no collinearity issue. Then check the plots of the new model. From the plot of residuals vs. fitted values, the distribution of the residuals is random around the zero line without funnel-shaped pattern, so the linearity and the equal variance assumptions are satisfied. The independence assumption is satisfied since the observations are independent. From the Shapiro-Wilks test, the p-value is 5.728e-07, which shows that the normality assumption is not satisfied. The value of R^2 is 0.3441, which means the model explains 34.41% of the data. Therefore, this model doesn't fit very well and we need further investigation to have a higher R^2.

We choose to remove some influential points manually and then fit the model again. In this model, we get R^2 with 35.18%. Points sticking out compared to the others still exist but they don't have great influence. Collinearity issue doesn't exist from the VIF output. Similar to the first sub model, assumptions are satisfied besides normality.

The interaction model has R^2 as high as 0.3634, however there is a collinearity between the interacted variables by exploring the interaction model. There are significant interactions between alcohol and sulphates with wine quality(positive correlation);total sulfur dioxide and sulphate with wine quality(negative correlation). There also exists an outlier from the Cook's distance plot. Through checking, this model meets all the assumptions besides normality. When we use the testing set to predict the quality, the R^2s for testing are 0.4075342 for linear regression and 0.4235409 for

interaction and the NRMSEs are 0.882 and 0.883 respectively, which implies a restricted level of fit of the models. Therefore, in order to gain higher accuracy, we need to consider applying other models.

Binomial Logistic Regression Model

After we transform the response variable quality to binary format: 0 represents the value of wine quality less or equal to 5 and 1 represents the value of quality greater to 5.(see Appendix A) At first, we fit all predictors in the regression model. Then, we check the model summary and find out there are some insignificant predictors. We use stepwise regression with backward elimination approach to accomplish variable selection. Based on stepwise regression results, we drop residual.sugar, density, pH, and chlorides. We also set our prediction on a training dataset based on log-odds, which is the default prediction for binomial logistic regression. We analyze model performance from accuracy of the testing set.

Analysis

From the logistic regression with all predictors, the predictors that are statistically significant are almost identical to the full multilinear regression model, but the variable pH is very different. Since pH is statistically significant in the full linear regression model, but not in the logistic regression model. After we apply backward elimination to selecting variables, we get quite similar composition of these variables with linear model after dropping insignificant variables. (Plot 4) Also, because logistic regression has little tolerance to multicollinearity issue, which means the independent variables should not have high correlation with each other. The backward elimination generates a similar result of section with previous model. From the VIF result for the previous linear model, we can see there is no collinearity problem in logistic regression. Approximate R^2 value is 25.20522 %. The training set accuracy is 74.68106%, and the testing set accuracy is 73.52941%. (see Appendix A)Therefore, the logistic regression can generate higher prediction accuracy compared with linear regression.

Random Forest Model

Random Forest is one way to improve the performance of decision trees. The starting algorithm of building trees is the same as the algorithm used in the decision tree. However, Random Forest only uses a small random subset of features to make a split instead of the full set of the features.

In our Random Forest model, at first, we create a new feature “taste” and classify the wines ranked by “5” and “6” as normal, the higher wines as “good”, and lower wines as “bad”. Next, splitting the data into 80% of the training set and 20% of the testing set. To fit in the model, we apply the Random Forest library. In the model, taste is the response variable and the predictors could be anything excluding the quality. The rest of the steps are prediction and the accuracy calculation.

Analysis

From the model summary (see Appendix A), we could see 500 trees are built and the model randomly selects 3 features at each split. The OBB estimate of error rate is 0.1367 which is not high, such that our prediction goes well. The bottom part of the model summary is a confusion matrix containing prediction versus actual as well as the corresponding class error rate shown on the right side. As we calculate the accuray, we need to get the results of prediction into a matrix, so we use *as.matrix()* to convert it. Then the accuracy could be successfully calculated by the basic rule.

In the end, the Random Forest model predicts accuracy at a rate of approximate 90.6%.

Conclusion

To sum up, in our experiment, we preprocessed the data at the first. Removing missing data and duplicated data. In the EDA part, we find out collinearity, plots showing with relationship, and the distribution. Based on the conclusion derived from EDA, we fit a linear model, logistic model, and random forest model, which are extended trials in our project.

From these models we apply, here are some finalized results. The linear regression models generally provide small R squared values. The variable alcohol contributes 22% to the response variable wine quality. Most of all the factors converge to the average category. It is possible to reach more advanced results if more data are available to us. We have also applied some models with interaction terms that are significant. The model with interaction terms generates a relatively better R squared value, but meanwhile brings serious collinearity issues. We also apply the backward elimination approach with the stepwise functions to drop variables with high AIC.

Next, we will determine how well our models can predict the true outcomes. According to the log-odds, the prediction of the models on the training data has an accuracy of 74.68%. For the prediction on the testing data, the accuracy is 73.53% which is slightly lower. The best accuracy on prediction comes from the random forest model. The accuracy is about 90.6%, which is a big improvement from the previous two models. In summary, the model with the highest prediction accuracy that we perform among this project is the random forest model. With this model, we have red wine quality prediction accuracy as high as 90.6%.

In the future, since we use the middle-size data in this project, we can apply the random forest model in a larger wine data set, for example, the combination of red and wine data from UCL Lab. Moreover, there are still potential improvements that we can do, like minimize the error and maximize prediction accuracy. On the other hand, we also can implement similar analysis to another type of food quality assurance, not only in the red wine industry.

Referrence

Cortez, P., Cerdeira, A., Almeida, F., Matos, T., & Reis, J. (2009). Modeling wine preferences by data mining from physicochemical properties. *Decision Support Systems*, *47*(4), 547-553.

Gupta, Y. (2018). Selection of important features and predicting wine quality using machine learning techniques. *Procedia Computer Science*, *125*, 305-312.

Kodali, T. (2016, February 4). Predicting wine quality using Random Forests. Retrieved from https://www.r-bloggers.com/predicting-wine-quality-using-random-forests/

Appendices

Appendix A

Binomial Logistic Regression:

This code is to transform the response variable quality to binary format.

```
red$category[red$quality <= 5] <- 0
red$category[red$quality > 5] <- 1
red$category <- as.factor(red$category)
head(red)
```

free.sulfur.dioxide <dbl>	total.sulfur.dioxide <dbl>	density <dbl>	pH <dbl>	sulphates <dbl>	alcohol <dbl>	quality <int>	category <fctr>
11	34	0.9978	3.51	0.56	9.4	5	0
25	67	0.9968	3.20	0.68	9.8	5	0
15	54	0.9970	3.26	0.65	9.8	5	0
17	60	0.9980	3.16	0.58	9.8	6	1
13	40	0.9978	3.51	0.56	9.4	5	0
15	59	0.9964	3.30	0.46	9.4	5	0

6 rows | 7-14 of 14 columns

Here is our backward selection result.

```
model_gl <- step(model_glm,direction="backward",trace=0)
model_gl$anova
```

Step <S3: Asis>	Df <dbl>	Deviance <dbl>	Resid. Df <dbl>	Resid. Dev <dbl>	AIC <dbl>
	NA	NA	1007	1053.508	1077.508
- residual.sugar	1	0.07502818	1008	1053.583	1075.583
- density	1	0.02213543	1009	1053.605	1073.605
- pH	1	0.08653277	1010	1053.691	1071.691
- chlorides	1	0.32894033	1011	1054.020	1070.020

This is prediction accuracy based on training data and testing data.

```
train_table <- table(predicted = train_pred, actual = red_train$category)
train_table
```

```
##            actual
## predicted     0   1
##   Bad Wine  361 139
##   Good Wine 119 400
```

```
sum(diag(train_table))/length(red_train$category)
```

```
## [1] 0.7468106
```

```
test_pred <- ifelse(predict(model_gl, newdata = red_test, type = "response") > 0.5, "Good Wine", "Bad Wine")
test_table <- table(predicted = test_pred, actual = red_test$category)
test_table
```

```
##            actual
## predicted     0   1
##   Bad Wine  121  51
##   Good Wine  39 129
```

```
sum(diag(test_table))/length(red_test$category)
```

```
## [1] 0.7352941
```

This is the code of creating a “taste” variable in the Random Forest model.

```
> wine$taste <- ifelse(wine$quality < 5, "bad", "good")
> wine$taste[wine$quality == 5] <- "normal"
> wine$taste[wine$quality == 6] <- "normal"
> wine$taste <- as.factor(wine$taste)
> str(wine$taste)
 Factor w/ 3 levels "bad","good","normal": 3 3 3 3 3 3 3 2 2 3 ...
```

This is the code of splitting data processes in the Random Forest model.

```
> samp2<-sample(1599, 1280)
> wine_train3 <- wine[samp2, ]
> wine_test3 <- wine[-samp2, ]
> dim(wine_train3)
[1] 1280   13
> dim(wine_test3)
[1] 319  13
```

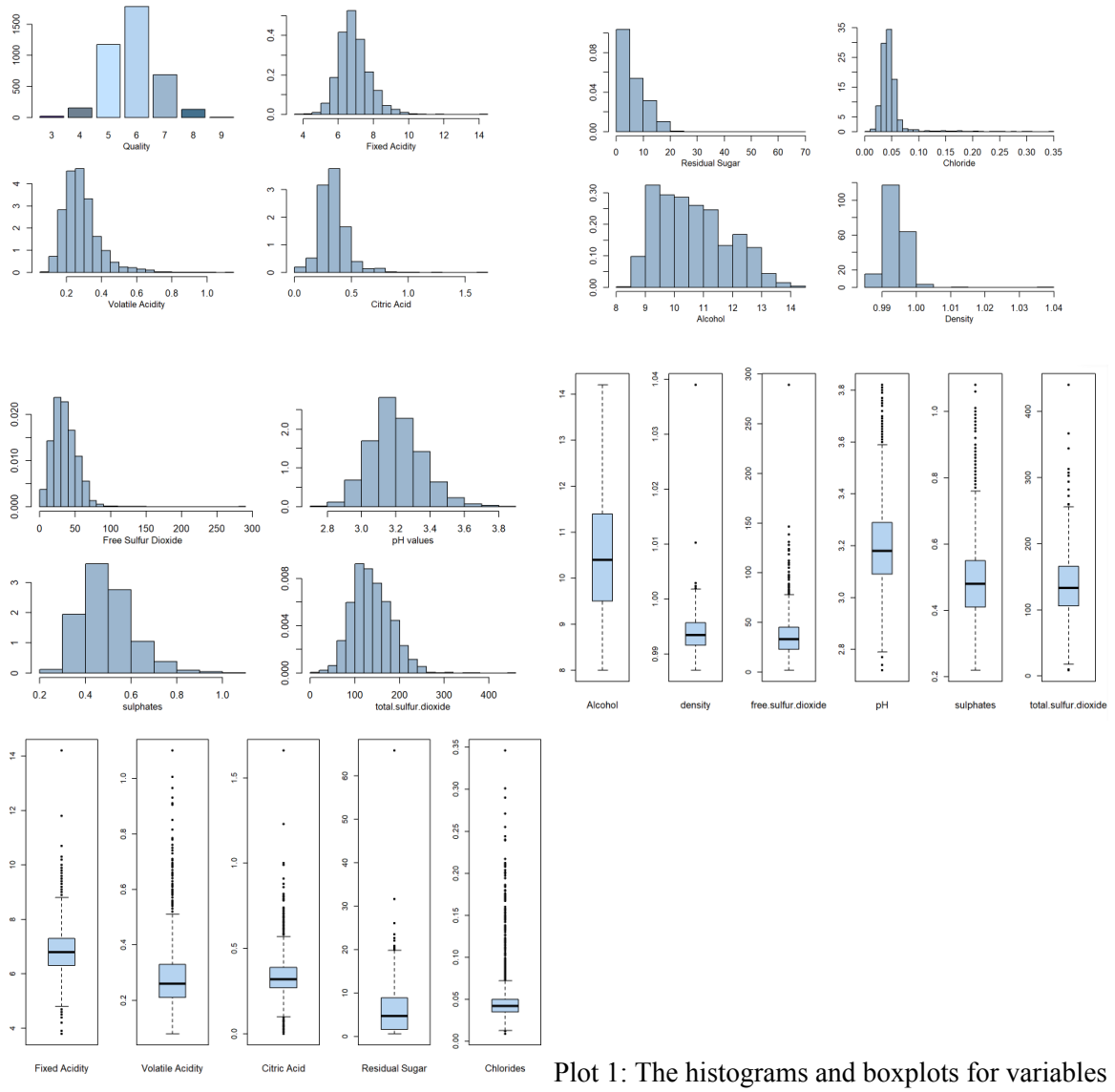
This is the Random Forest model and its model summary

```
> model<-randomForest(taste ~ . - quality, data = wine_train3)
> prediction<-predict(model, newdata = wine_test3)
> cm=as.matrix(table(prediction, wine_test3$taste))
> sum(diag(cm))/sum(cm)
[1] 0.9059561
```

```
> model

Call:
 randomForest(formula = taste ~ . - quality, data = wine_train3)
               Type of random forest: classification
                     Number of trees: 500
No. of variables tried at each split: 3

        OOB estimate of  error rate: 13.67%
Confusion matrix:
       bad good normal class.error
bad      1    1     50  0.98076923
good     0   85     92  0.51977401
normal   2   30   1019  0.03044719
```



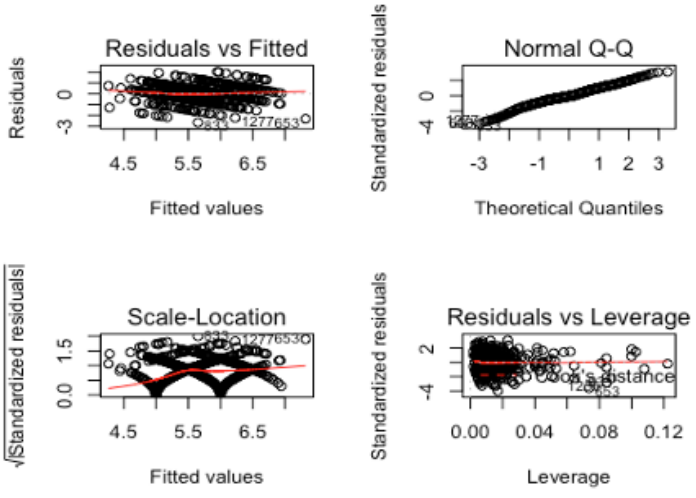
Plot 1: The histograms and boxplots for variables

```
## Call:
## lm(formula = quality ~ ., data = red_train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.64972 -0.37075 -0.04381  0.45779  2.05049
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          7.023361  27.964986   0.251  0.80175
## fixed.acidity        0.003500  0.034381   0.102  0.91895
## volatile.acidity    -1.152120  0.152730  -7.543 1.02e-13 ***
## citric.acid         -0.190800  0.191075  -0.999  0.31825
## residual.sugar       -0.005372  0.019194  -0.280  0.77962
## chlorides           -1.525585  0.524762  -2.907  0.00373 **
## free.sulfur.dioxide  0.004739  0.002772   1.710  0.08766 .
## total.sulfur.dioxide -0.002978  0.000949  -3.138  0.00175 **
## density             -2.533896  28.543118  -0.089  0.92928
## pH                  -0.509496  0.251497  -2.026  0.04304 *
## sulphates            0.790526  0.144989   5.452 6.25e-08 ***
## alcohol              0.298990  0.034581   8.646  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6691 on 1007 degrees of freedom
## Multiple R-squared:  0.3476, Adjusted R-squared:  0.3405
## F-statistic: 48.78 on 11 and 1007 DF,  p-value: < 2.2e-16
```

```
vif(linear_full)

##        fixed.acidity     volatile.acidity          citric.acid
##             8.063044             1.763059             3.090034
##       residual.sugar           chlorides      free.sulfur.dioxide
##             1.687716             1.497952             1.969234
## total.sulfur.dioxide              density                   pH
##             2.267666             6.208979             3.467831
##            sulphates              alcohol
##             1.443301             3.116789
```

Plot 2: Full linear model summary and VIF output



Plot 3: Diagnostic Plots for Full Linear Model

```
summary(model_gl)
```

```
##
## Call:
## glm(formula = category ~ fixed.acidity + volatile.acidity + citric.acid +
##     free.sulfur.dioxide + total.sulfur.dioxide + sulphates +
##     alcohol, family = binomial(link = "logit"), data = red_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.2997  -0.8318   0.3141   0.8023   2.4009
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)          -9.270030   1.139541  -8.135 4.12e-16 ***
## fixed.acidity         0.112848   0.062910   1.794   0.0728 .
## volatile.acidity     -3.547276   0.580535  -6.110 9.94e-10 ***
## citric.acid          -1.419655   0.674957  -2.103   0.0354 *
## free.sulfur.dioxide   0.017295   0.010046   1.722   0.0851 .
## total.sulfur.dioxide -0.014993   0.003425  -4.377 1.20e-05 ***
## sulphates             2.013660   0.476890   4.222 2.42e-05 ***
## alcohol               0.946829   0.088196  10.735  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1409.2  on 1018  degrees of freedom
## Residual deviance: 1054.0  on 1011  degrees of freedom
## AIC: 1070
##
## Number of Fisher Scoring iterations: 4
```

Plot 4: Binomial Logistic Model Summary

Appendix B

Team Member	**Responsibilities**
Jiahong Lu	Abstract, Keywords, Random Forest, Conclusion, Proofread
Yiqi Xiong	Introduction, Linear Model, Logistic Model, Proofread
Keqian Zhang	Data Prepocressing, EDA
Kexin Yi	Analysis and diagnostic
Zhongyuan Hua	EDA, Conclusion