Personal Project - Classfication Modeling

Celine Chiou(Yun-Chyau, Chiou) 300230747

December 17, 2021

Introduction

Nowadays,many companies are using the wine certification to prove the quality of their product. The price of the wine depends on various variables, for example, the acidity, residual sugar, chlorides, free.sulfur.dioxide... etc. Furthermore, As a result, this project aims to determine which features are the best quality wine indicators and generate insights into each of these factors to our model's wine quality. In the following project, we will use 7 different classification methods to find out which one has the most accurate model for our case.

Initial Exploratory Data Analysis

Data Dictionary

Table 1: Data dictionary

Variable Name	Variable Discription	Variable Type
fixed.acidity	Non-volatile acids that do not evaporate readily	0
volatile.acidity	High acetic acid in wine which leads to an unpleasant vinegar taste	0
citric.acid	Acts as a preservative to increase acidity	0
residual.sugar	Amount of sugar remaining after fermentation stops.	0
chlorides	The amount of salt in the wine	0
free.sulfur.dioxide	It prevents microbial growth and the oxidation of wine	0
total.sulfur.dioxide	Amount of free + bound forms of SO2	0
density	Sweeter wines have a higher density	0
pH	Level of acidity on a scale of 0–14	0
sulphates	Available in small quantities in wines makes the drinkers sociable	0
alcohol	Wine additive contributes to SO2 levels and acts as an antimicrobial	0
quality	Which is the output variable/predictor	1

Data Visualization

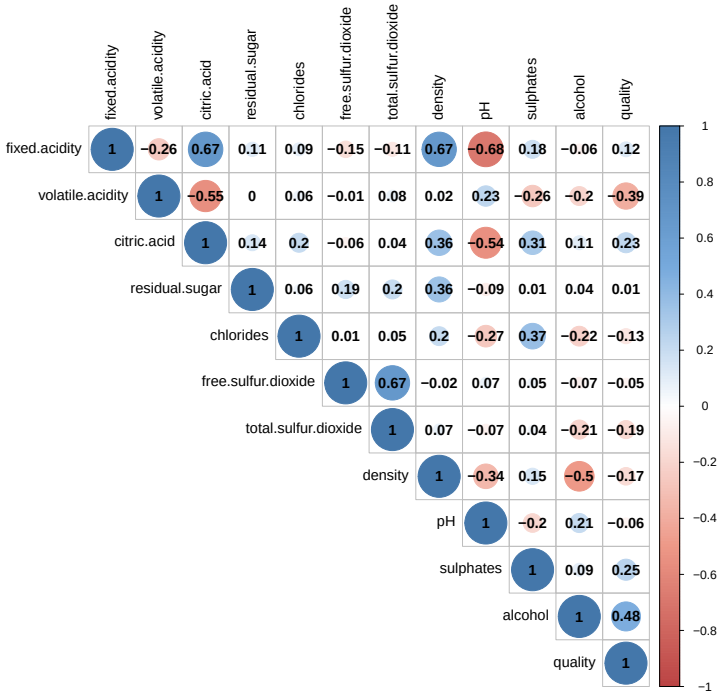

Figure 1: Correlation of each variables of red wine

Figure 1 We try to figure out the relationship between each variables. For the red wine quality, we could see that the variables of alcohol, sulphates, citric acid, and fixed acidity are positive correlated to the quality. On the contrary, volatile acidity, total sulfur dioxide, density, and chlorides are negative correlated to the quality. Among them, alcohol and volatile acidity are the most influential factor on the quality.

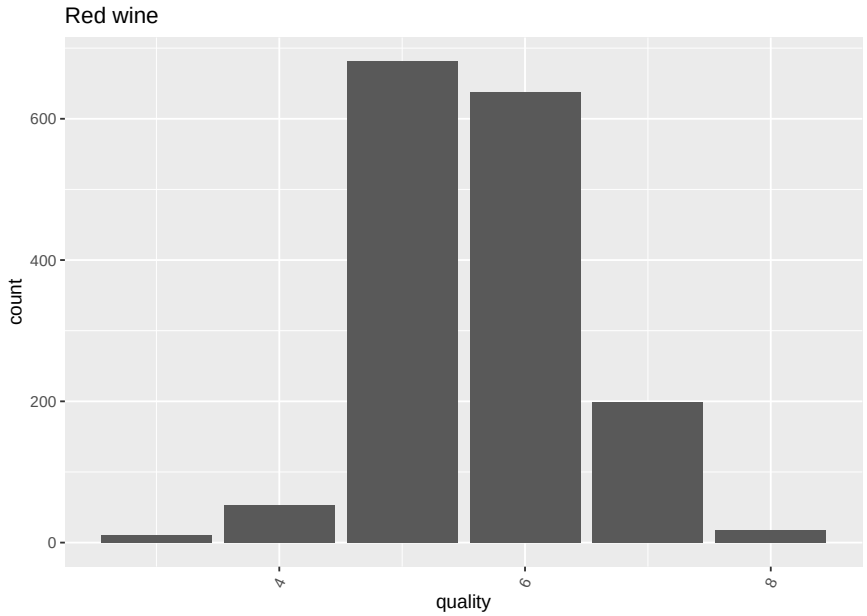

Figure 2: Bar chart of quality of red wine

Figure 2 According to the figure 2, we try to figure out the distribution of the red wine quality.The quality variable is normal distributed and we could see that quality 5 and 6 have the most points, and quality 7 has the third many points. Therefore, we decide to add dummy variables between 5 and 6 to distinguish the "good wine" from "bad wine".

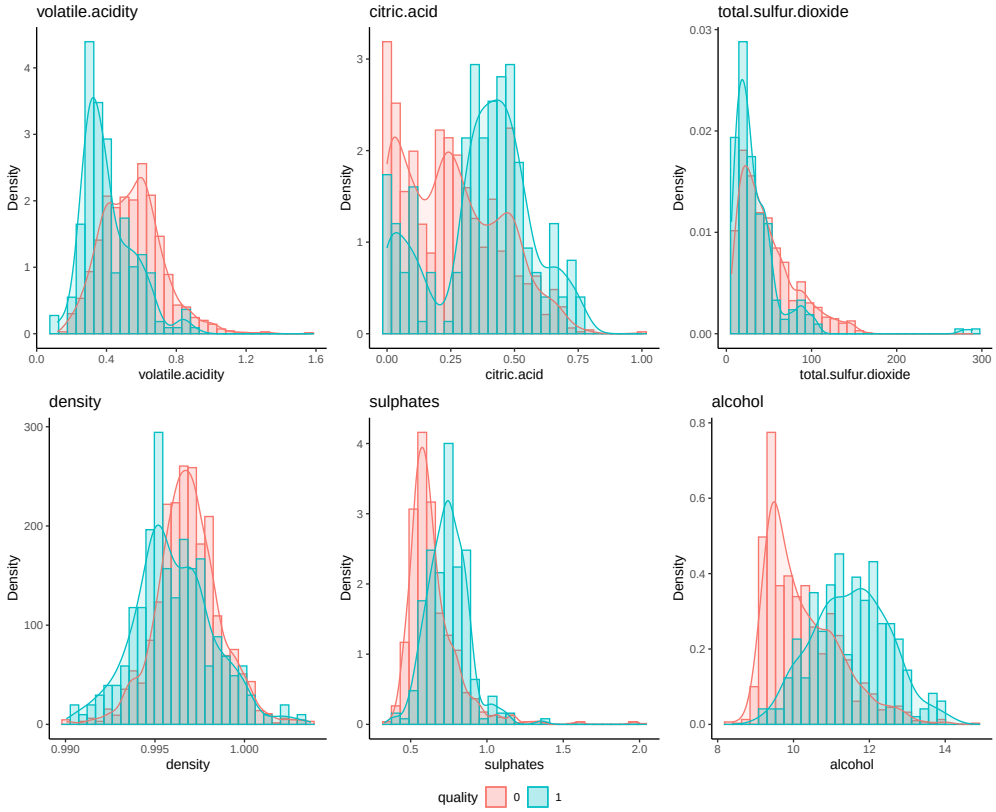

Figure 3: Distribution of correlated variables

Figure 3 According to the results we have above, we then plot the 6 variables we mentioned in the correlation plot to see more detail information about the classification. Furthermore, we separate the distribution from the quality >6 and the quality <6 as we mentioned above (1 represents the quality >6, 0 represents the quality<6). Based on the figure 3, we could see that its really clear that the distribution of volatile.acidity, citric.acid, density, sulphates, and alcohol separate really obviously from good wine to bad wine. volatile.acidity and density are negative correlative to this classification. On the other hand, citric.acid, sulphates, and alcohol are positive correlative to this classification.

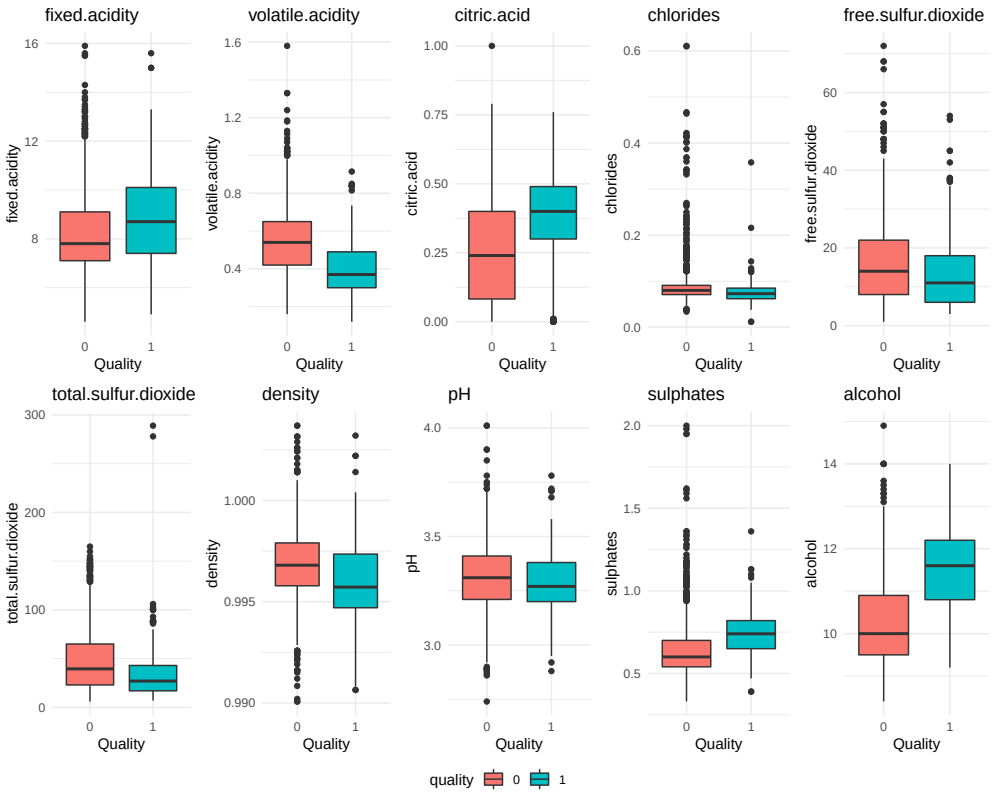

Figure 4: Box plots for quality verses different variables

Figure 4 The relationship of different variables we get from this plots is the similar as the plots above. The only thing we would like to mention is that these independent variables show no significant relationship with quality: residual.sugar, chlorides, and total.sulfur.dioxide.

Classification Modeling

Creating Dummy variable and Training/Testing Dataset

Because this is a binary classification problem. After the initial exploratory data analysis, we are now confident to create a dummy variable between quality 5 and quality 6 to identify good wine and bad wine. Besides that, we are going to use random sampling to create the training set and the testing set (choose a 70/30 split for training dataset and testing data set).

```
red <- read.csv("winequality-red.csv")
red <- red%>%mutate(
  quality = ifelse(quality>6, 1, 0)
)
red$quality <- as.factor(red$quality)

set.seed(202112)
index <- sample(1:nrow(red))
train_index <- index[1:round(0.7*nrow(red))]
test_index <- index[(round(0.7*nrow(red))+1):nrow(red)]
```

```
train <- red[train_index, ]
test <- red[test_index, ]
```

Logistic Regrssion Model

```
##
## Call:
## glm(formula = quality ~ fixed.acidity + volatile.acidity + residual.sugar +
##     chlorides + total.sulfur.dioxide + density + sulphates +
##     alcohol, family = binomial, data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.9173  -0.4276  -0.2264  -0.1350   2.9483
##
## Coefficients:
##                        Estimate Std. Error z value Pr(>|z|)
## (Intercept)           2.512e+02  1.066e+02   2.357 0.018434 *
## fixed.acidity         2.443e-01  9.558e-02   2.556 0.010595 *
## volatile.acidity     -2.956e+00  7.764e-01  -3.807 0.000141 ***
## residual.sugar        2.871e-01  8.383e-02   3.425 0.000614 ***
## chlorides            -7.398e+00  3.356e+00  -2.204 0.027493 *
## total.sulfur.dioxide -1.321e-02  3.862e-03  -3.421 0.000623 ***
## density              -2.646e+02  1.071e+02  -2.471 0.013465 *
## sulphates             3.627e+00  6.506e-01   5.575 2.47e-08 ***
## alcohol               7.072e-01  1.312e-01   5.389 7.09e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 862.94  on 1118  degrees of freedom
## Residual deviance: 602.12  on 1110  degrees of freedom
## AIC: 620.12
##
## Number of Fisher Scoring iterations: 6
```

Logistic regression is used to model the binary dependent variable, so it is suitable for our case.General speaking, logistic regression should be more suitable while comparing to linear regression model, because logistic model is a S curve to predict the categorical dataset and linear regression is to predict the continuous dataset.

For the logistic regression, first of all, we start with the model with all variables. Apparently, the result us not really well, since the AIC is 624.47 and residual deviance is 604.47. Therefore, we decide to create model 2 which deletes the variables that are not significant. And we could see that the remaining feature variables are the same as our EDA. After deleting all variables that are not significant, we could see that all variables in model 2 become significant. Furthermore, the AIC become 620.12 and residual deviance is 602.12, which means that the model is more fitted.

The following is the predict table and the accuracy rate of the linear regression of model 2. The accuracy rate for the linear regression model is 87.7%

```
##
```

```
## pred.prob2   0   1
##          0 397  48
##          1  11  24
```

```
## [1] 0.8770833
```

Linear Discriminant Analysis

```
#fit model
lda.fit  <- lda(quality  ~ fixed.acidity+ volatile.acidity+ residual.sugar+ chlorides+
                    total.sulfur.dioxide+ density+ sulphates+ alcohol,
                data = train)
lda.pred <- predict(lda.fit, test)
lda.class <- lda.pred$class
```

LDA is a analysis which is similar to logistic analysis. Instead of collecting directly calculating the conditional probability of classes based on the parameters, it creates separate distributions of the parameters for each class and uses Bayes Theorem to convert those distributions into conditional probabilities. We use the same feature variables as the the logistic regression model, and the following is the predict table, the accuracy is 88.54%

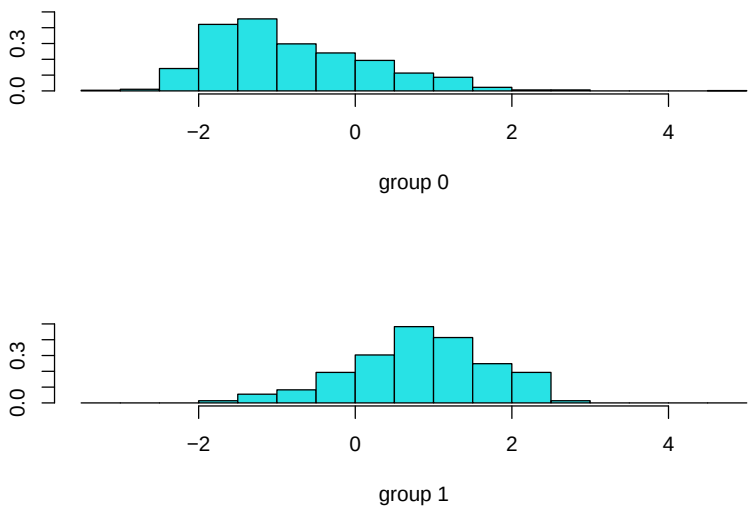

Figure 5: LDA plots

```
##
## lda.class   0   1
##         0 393  40
##         1  15  32
```

```
## [1] 0.8854167
```

Figure 5 Describe the LDA model for group 0(quality <6) and group 1(quality >6), we could observe that group 0 is a left skewed distribution and group 1 is a right skewed distribution.

Quadratic discriminant analysis

```
qda.model = qda (quality  ~ fixed.acidity+ volatile.acidity+ residual.sugar+ chlorides+
                      total.sulfur.dioxide+ density+ sulphates+ alcohol,
                data = train
                 )
```

QDA is a modified model of LDA, but bases on different assumption. Both QDA and LDA assume that the observation is drawn from a normal distribution; however, the QDA assumes that each class has its own covariance matrix. The higher the dimension of the dataset, the larger the parameter we have to estimate in QDA. And in our case, we use the same feature variables in the logistic model, which only have 8 dimensions, as a reult, we could expect that the accuracy may be really same as LDA. The accuracy of the QDA is 86.88

```
##
## qda.class   0   1
##         0 376  31
##         1  32  41
```

```
## [1] 0.86875
```

K-Nearest Neighbors

```
#fit model

train.x<- subset(train, select= c(fixed.acidity, volatile.acidity, residual.sugar, chlorides,
                      total.sulfur.dioxide, density, sulphates, alcohol))
test.x<- subset(test, select= c(fixed.acidity, volatile.acidity, residual.sugar, chlorides,
                      total.sulfur.dioxide, density, sulphates, alcohol))

train.y<- train$quality
test.y<- test$quality
```

K-nearest works different way from all of the methods above. The K-Nearest neighbors determine whether or not a point lies in a sparse region of the feature space by computing the average of the distances to the k-nearest neighbors of the point. We refer to this quantity as the kNN score for a point. Intuitively, the points in dense regions will have many points near them and will have a small kNN score. The following tables show when k=1 and k=3 respectively. we could conclude that the accuracy for k=1 is 86.25% and when k=3, the accuracy is 86.40%

k=3

```
##           test.y
## knn.pred3   0   1
##         0 384  49
##         1  24  23
```

```
## [1] 0.8625
```

k=1

```
##          test.y
## knn.pred1   0   1
##         0 368  35
##         1  40  37
```

```
## [1] 0.8604167
```

Random forest

Random forest is a classification for constructing a multitude of decision trees at training time. For classification tasks, the output of the random forest is the class selected by most trees. To be more specific, we do not need to select feature variables since Random Forest automatically calculates variable's importance.

The following table show the importance of all variables, and we could see that the importance is correspond to the feature variables we choose in the above models. Moreover, the accuracy of the random forest is 92.92%

```
##                       MeanDecreaseGini
## fixed.acidity                 19.04464
## volatile.acidity              28.55309
## citric.acid                   23.71643
## residual.sugar                18.48477
## chlorides                     19.25481
## free.sulfur.dioxide           14.37677
## total.sulfur.dioxide          19.45325
## density                       25.05448
## pH                            15.28070
## sulphates                     28.06619
## alcohol                       40.74144
```

```
##
## pred   0   1
##    0 402  28
##    1   6  44
```

```
## [1] 0.9291667
```

XGBoost

XGBoost is a supervised-learning algorithm used for regression and classification on large datasets.Boosting is considered to be more effective, since it controls both the bais and variance. Also, it is an ensemble learning technique to build a strong classifier from several weak classifiers in series. The following is the table of the prediction, and the accuracy is 91.46%

```
##
## xg_pred_class   0   1
##             0 390  23
##             1  18  49
```

```
## [1] 0.9145833
```

Compraison and Discussion

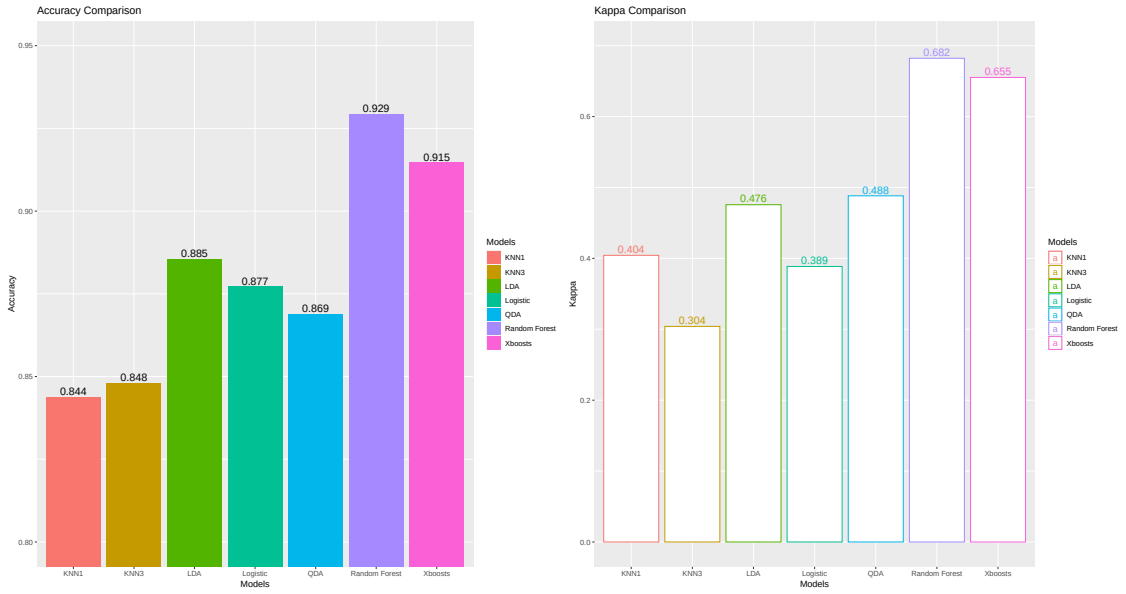

Figure 6: Red Wine Comparison plots

Figure 6

According to the accuracy plot, random forest was the highest performing model, which could correctly classified about 91.2%.The second best performing model is XGboosts. The logistic model, LDA, and QDA method perform almost equally good, which was about 86%-90%. On the contrary, the worst classification method is K-nearst, which is only 86%. The reason may be that the points are too close to each other, so K-nearst could not identified different quality group.

Kappa measures the degree of agreement of the nominal or ordinal assessments made by multiple appraisers when assessing the same samples.Kappa values from +1 to -1, the stronger the agreement. In the Kappa comparison random forest is also the best performing model, and the second best is also XGboosts classification method. However, the comparison between knn1 and knn3 comes really different while comparing to the accuracy plots.

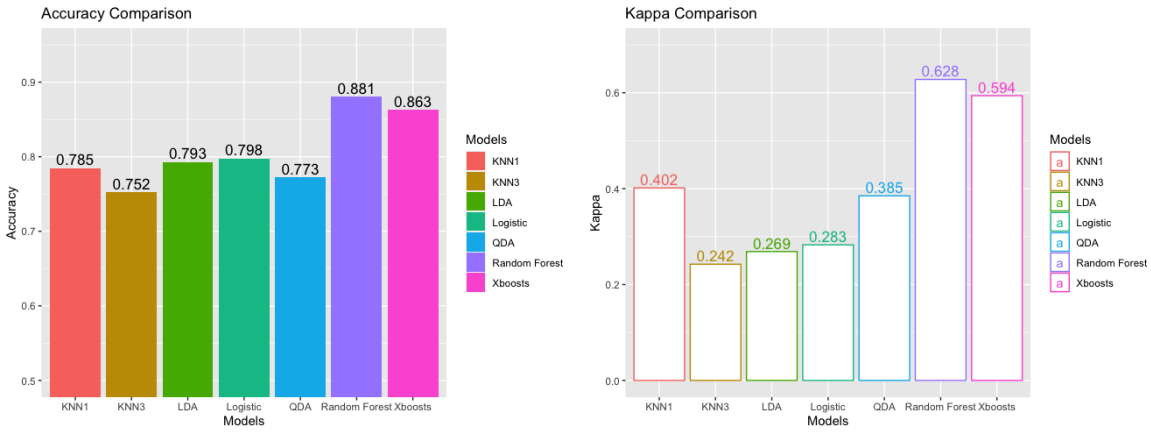

Figure 7: White Wine Comparison Plot

Figure 7

In order to check if the model predict ability, we use the white wine dataset to do the comparison again. In the white wine model, the feature variables that are significant are fixed.acidity, volatile.acidity, residual.sugar, chlorides, free.sulfur.dioxide, density, pH, and sulphates, which is really different from the red wine. The plots shows that Random forest and XGboosts are still the first two best classification method. However, all of the accuracy is lower than the red wine ones. The reason may be that there is some collinearity problem exists the data set, we could deal with the LASSO regularization technique. To improve the model, we could delete the outliers and use cross validation.