STA141A Final Project

Aly Kapasi - 915223613:

Contributed towards most of the programming in the project and wrote the statistical objective and some explanations of the graphs

Songmin Li - 915214334:

Contributed towards most of the writing and explanations also made the frequency distributions, pairwise plots and boxplots.

Background

The purpose of research behind this project is to delve into the nature of red wine quality, specifically to investigate the variables affecting the quality of red wine and the relationship between those variables. Under this purpose, the data of physicochemical properties of 'Vinho Verde' red wine from a study conducted in 2009 by Cortez et al are targeted to make research.

The dataset consists of 12 variables containing the information about red vinho verde wine, and the data were collected from the UCI machine learning repository, which proves the reliability of the data. Of these 12 variables, 11 of them are input variables and they are fixed acidity, volatile acidity, citric acid, residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide, density, pH, sulphates, and alcohol; the last one, which is the output variable, is quality (score between 0 and 10). According to the information obtained, these properties (or some of them) can determine the quality of the wine in a broad sense and the price at which wine is introduced to the market also fluctuates based on its own quality. Therefore, from the perspective of both business and statistics, the value and significance of this research are valid.

The data were collected through May 2004 to February 2007 and it is original from the samples which were protected in an official certified Entity (CVRVV). Specifically, CVRVV is an inter-professional organization which aims to prove the professions of Vinho Verde in the area of both production and regional & national business. In detail, the number of 1599 red wine dataset was recorded by a computer system called iLab, which means that the collected data and information can be faithfully and automatically managed into computer while following the process of testing samples, and finally the whole database was exported to a complete file (.csv).

The experiments involved both analytical and sensory. In order to avoid missing samples and ensure the integrity of the data, only the type of most common physical and chemical tests were performed for analytical experiments. To test the sensory property, preference, accurately, there are at least three sensory evaluators who evaluated the red wine quality based on blind tasting. They rated the wine on a scale from 0(very bad) to 10(excellent), and good quality wine means a quality rating higher than 6. Based on their assessments, the final sensory scores were determined as the median.

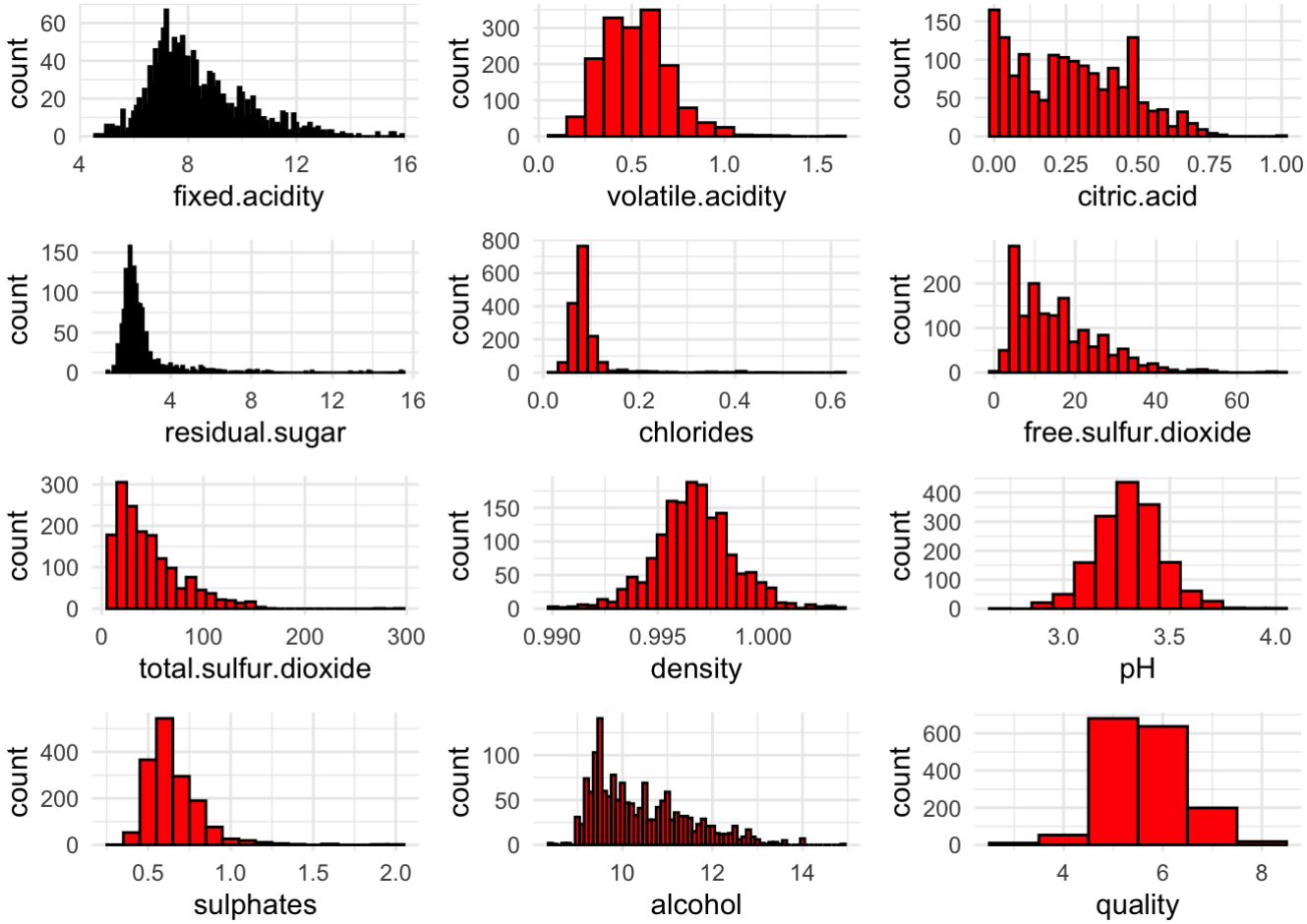
Vinho Verde comes from a small region in northern Portugal, which is known for its premium white, red and rose wines. It can produce about 15% of the wine production in Portguese. Plus, red wine is one of the most common varieties of wine. Thus, the results obtained by analyzing Vinho Verde"s data can be a good reference for wine companies to adjust their strategic approach, which is beneficial to the wine quality and commercial value of wine, which can also benefit the public. Overall they are the reasons why we choose this dataset to make research.

Statistical Objective

The main goal of this project is to see if physicochemical factors can be used to determine whether the wine is of good quality or not. Good quality wine being any wine with a quality rating higher than 6. The methods we used to solve this question were Exploratory Data Analysis, Linear Regression, Logistic Regression, Stepwise Regression, Linear Discriminant Analysis and K-Nearest Neighbors Clustering. All the methods with exception to linear regression were to determine whether the wine was of good quality or not; the regression was used to see whether the variables could be used to predict the quality of the wine. The Exploratory Data Analysis was used to see relationships and distributions of the variables.

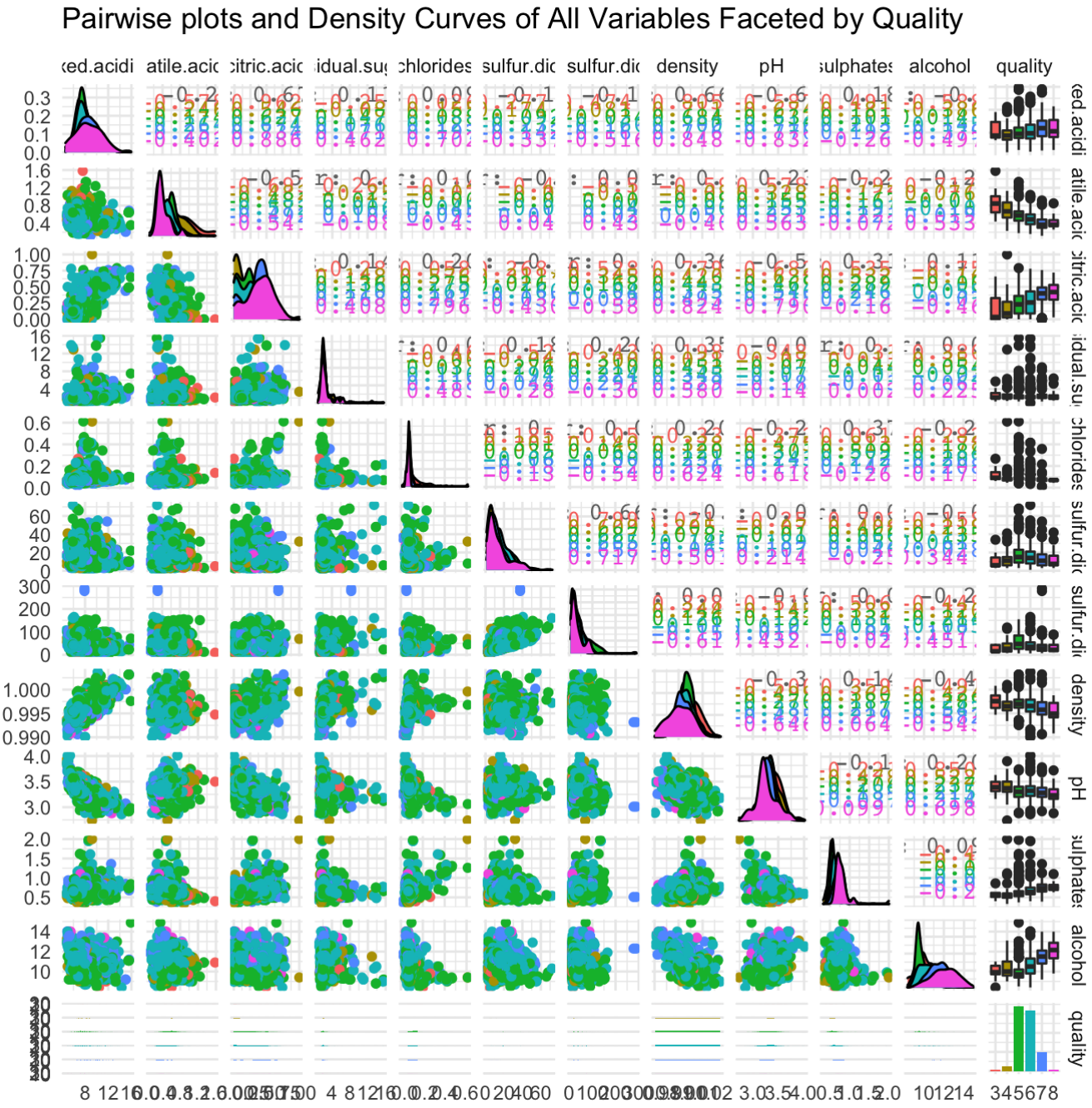
Diagnostics

Distributions of Variables



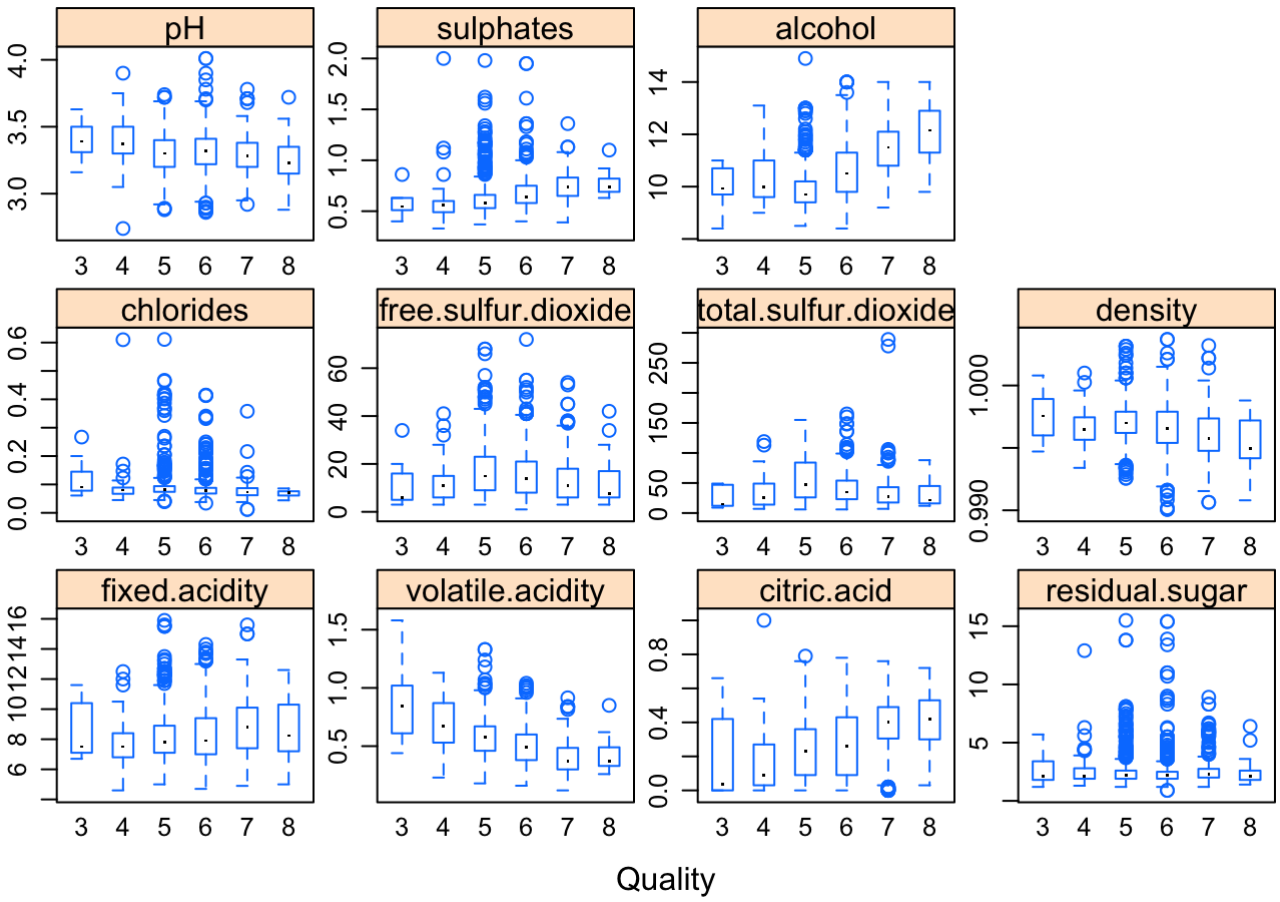
The histograms above show the distributions of all the variables in the dataset. Fixed acidity seems to have the most mass between 6 and 9; the volatile acidity seems to have most of it's mass between 0 and 1. The most common citric acid contents for the wine is 0, and then there is a steady decline in quantity of the citric acid content with exception to 0.5 where there is increased quantity. The residual sugar is in g/L and most of the wines are in the range of 1 to 3 g/L. Most of the wines have a chloride content between 0 and 0.2. Most of the wines have very little free sulfur dioxide as there is a negative slope for the distribution, similarly for total sulfur dioxide. Both density and pH for the wines seem to be normally distributed, with a mean of 0.997 and 3.2 respectively. Sulphates and alcohol content have a positive skew. The distribution of the quality shows that most of the wines are rated either 5 or 6, with some rated 7 and a few rated either 3, 4 or 8.

Pairwise Distribution of Variables



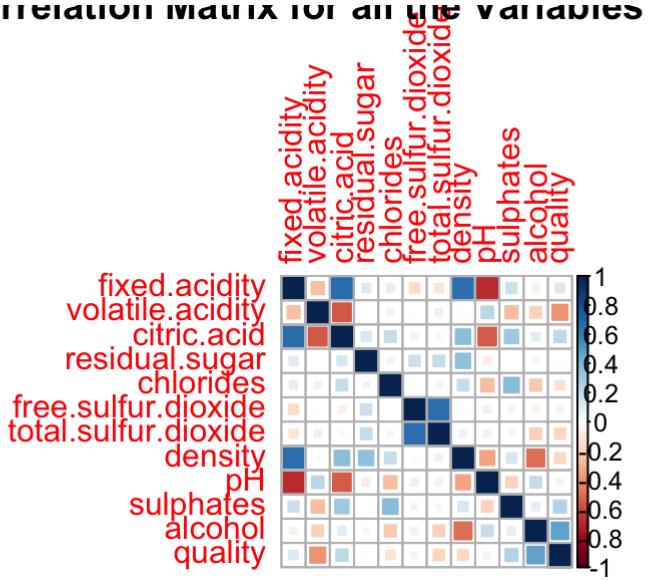
The ggpairs function plots all the variables afainst each other. As can be seen from the plots above, most of the variables do not have any linear relationship. Some variables that are positively related to each other are fixed acidity vs citric acid or density, citric acid vs density and total vs free sulfur dioxide. There are also a few relationships that are negatively correlated such as pH vs fixed acidity or citric acid or density or density vs alcohol. The plots also contain the distribution of data with respect to wine quality for each variable. It seems that most of the distributions seem to be positively skewed, where majority of the mass is behind the mean, though there are some different distributions too. Volatile acidity, citric acid and alcohol concentration all have uncommon distributions. Density and pH both seem to be roughly normally distributed as most of the mass is centered around the mean.

Boxplots of Variables, with respect to Quality



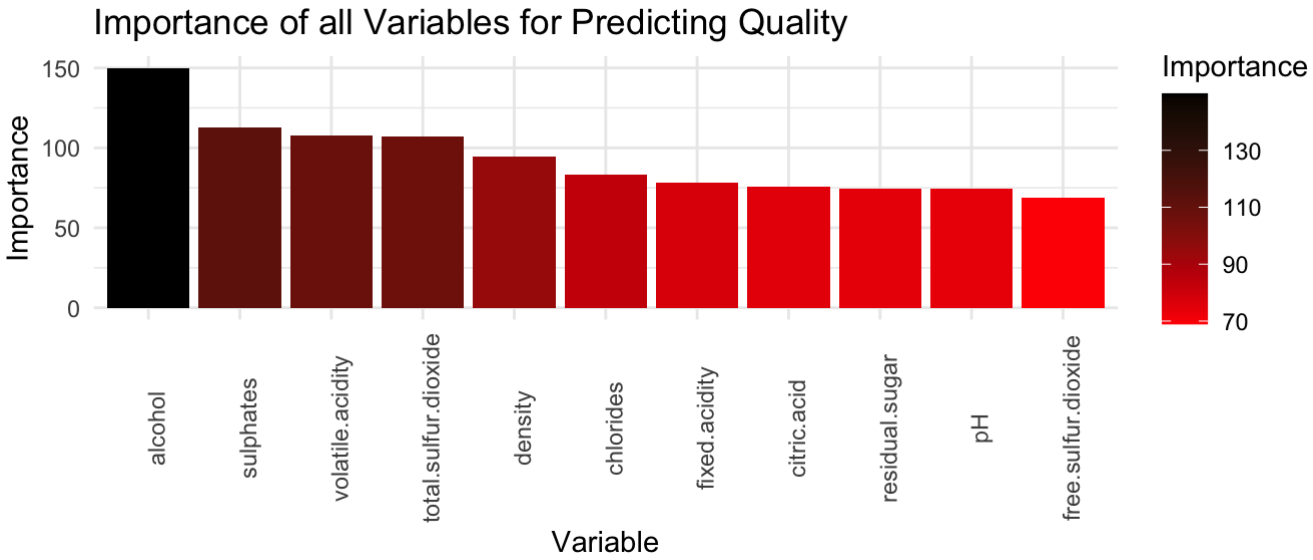
The boxplots show the relationship between the input variables and the output variable, quality. Most of the predictor variables seem to have similar values across the wine qualities, but there are some that have distint differences between wine qualities; the first such variable is the sulphates concentration, where it shows that the higher quality wines seem to, on average have higher sulphate levels compared to the lower quality wines. Alcohol also has differing values between wine qualities, as generally higher quality wines have higher alcohol concentrations. The boxplots also show that wines with lower volatile acidity are commonly better wines. Finally wines with higher citric acid amounts seem to be of better quality.

Correlation Matrix for all Variables



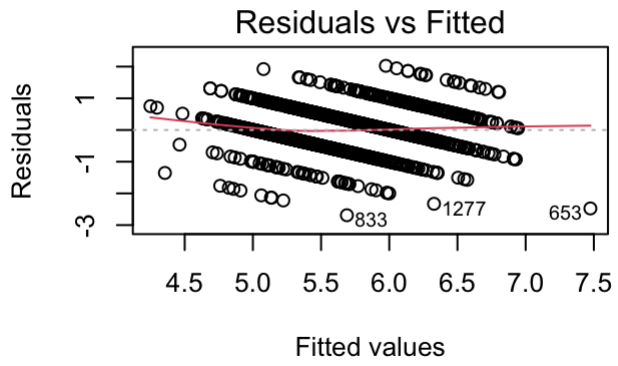
The correlogram shows that there are some variables that have strong relationships with each other. Some strong positive relationships are fixed acidity and citric acid, fixed acidity and density, free sulfur dioxide and free sulfur dioxide, while pH and fixed acidity, pH and citric acid, alcohol and density are the variables with strong negative relationships.

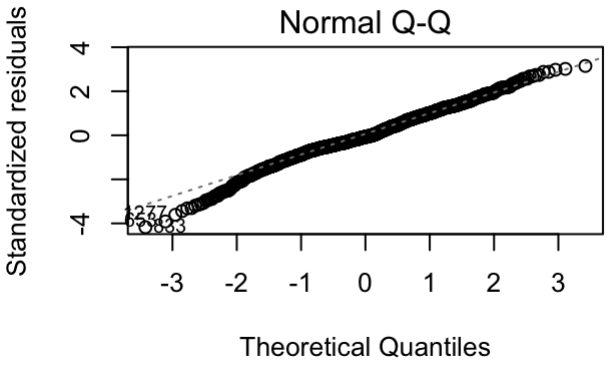
Ranking Importance of all Variables

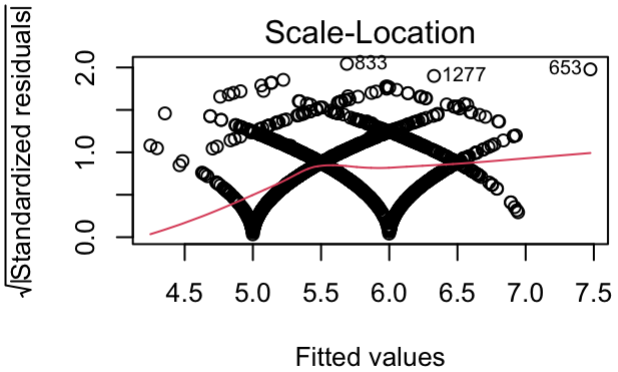


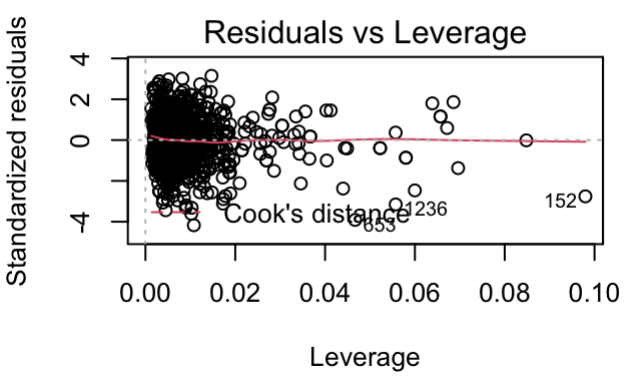
Using the randomForest library's importance function, we found the importance of each of the variables in determining wine quality. The plot above shows that the most important predictor is the alcohol content. Sulphate amount, volatile acidity and total sulfur dioxide being the next most important predictors, while free sulfur dioxide was the lease significant predictor.

Diagnostics for Multiple Linear Regression










```
## [1] 0.3605517
```

The Residuals vs Fitted plot, has a trend line which is approximately horizontal on x = 0, meaning the residuals don't have a fitted pattern. A pattern would indicate to an error in a part of the linear regression model. The Normal QQ Plot follows the dotted line fairly well save for the lowest quantiles and residuals, which seem to stray slightly off the line, but majority of the points are on the line enough to assume normality of the residuals. The Scale-Location plot shows that there may be a slight heteroscedaticity problem with the data, as the line is fairly horizontal after 5.5, but before that it is sharply inclined which alludes to a heteroscedaticity problem with the residuals. This means that the residuals before 5.5 are not homogenous. Finally the Cook's Distance plot shows that none of the points are outliers as none of them are close to the dotted red line, meaning removing any data point will not improve the accuracy of the model.

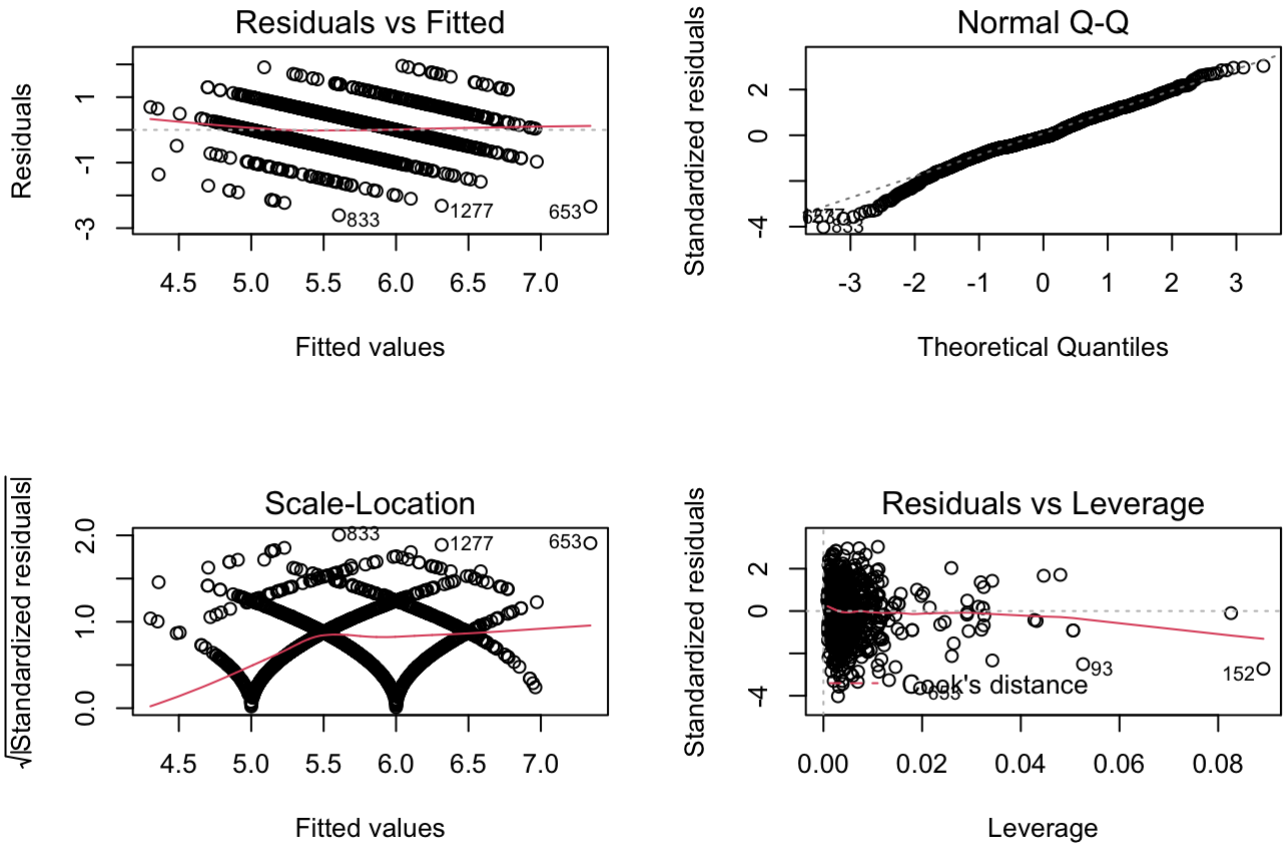
Analysis

Stepwise Multiple Linear Regression

```
##   adjr2 mallows bic
## 1     8       7   6
```

```
##          fixed.acidity volatile.acidity citric.acid residual.sugar chlorides
## 1  ( 1 ) " "           " "              " "         " "            " "
## 2  ( 1 ) " "           "*"              " "         " "            " "
## 3  ( 1 ) " "           "*"              " "         " "            " "
## 4  ( 1 ) " "           "*"              " "         " "            " "
## 5  ( 1 ) " "           "*"              " "         " "            "*"
## 6  ( 1 ) " "           "*"              " "         " "            "*"
##          free.sulfur.dioxide total.sulfur.dioxide density pH  sulphates alcohol
## 1  ( 1 ) " "                 " "                  " "     " " " "       "*"
## 2  ( 1 ) " "                 " "                  " "     " " " "       "*"
## 3  ( 1 ) " "                 " "                  " "     " " "*"       "*"
## 4  ( 1 ) " "                 "*"                  " "     " " "*"       "*"
## 5  ( 1 ) " "                 "*"                  " "     " " "*"       "*"
## 6  ( 1 ) " "                 "*"                  " "     "*" "*"       "*"
```

```
## [1] 0.3571736
```



We used both normal and stepwise regression to compare the models, and compared their effectiveness. The r^2 for the full linear regression model was 0.3605517, while the stepwise regression model had a r^2 of 0.3571736, considereing the stepwise model had 5 less variables the discrepency in r^2 values is very good. The stepwise regression was done by the data through the regsubsets function which is a stepwise regression function, and then using the summary choosing the model with the best model selection criteria. The three criteria used were adjusted r^2, Mallow's CP and BIC. For the sake of simplicity the smallest model was the BIC criteria, so that is

what was used. The stepwise regression summary also showed which variables should be included in the final model. Regardless of the imporvement using the stepwise regression a r^2 value of 0.36 is not good enough for prediction.

Linear Discriminant Analysis

```
##   fixed.acidity volatile.acidity citric.acid residual.sugar  chlorides
## 0      8.234069        0.5499904   0.2547697       2.510605 0.09023417
## 1      8.724204        0.4041083   0.3710828       2.735669 0.07454140
##   free.sulfur.dioxide total.sulfur.dioxide   density       pH sulphates
## 0            15.99952             47.76679 0.9968349 3.313292 0.6422937
## 1            14.03822             35.94904 0.9959173 3.294013 0.7400637
##    alcohol
## 0 10.26823
## 1 11.56401
```

```
##     predicted
## true   0   1
##    0 327  13
##    1  38  22
```

The next method used was Linear Discriminant Analysis, but instead of using quality as the output variable, we used whether the wine was good or not, as it is derived from the quality variable and that may also help the model be more accurate. As seen from the table most of the means are close together, so there is no particular mean that can be significantly more important in helping classify the wine as good or bad.

The table shows that it is fairly accurate at classifying whether the wine would be good or not, as the misclassification error rate is 0.1275.

Logistic Regression

```
##          (Intercept)        fixed.acidity     volatile.acidity
##         191.14300846           0.22476216          -3.43198682
##       residual.sugar            chlorides total.sulfur.dioxide
##           0.22707815         -10.50465580          -0.01182277
##              density            sulphates              alcohol
##        -204.27917823           3.76643986           0.76500295
```

```
##     predicted
## true   0   1
##    0 330  10
##    1  40  20
```

```
##     predicted
## true   0   1
##    0 330  10
##    1  41  19
```

The logistic regression was also done stepwise. The first model is a simple logistic regression model, but the second model is stepwise logisitic regression. The simple regression model is used to create the stepwise model, a stepAIC is run on the model and then the model chosen by the function is then run through glm to get the

stepwise model. Then both model run simulations against the test data, and the results are charted on a confusion table. The confusion tables show that the second model is almost as good as the full model at predicting whether the wine is good(1) or bad(0). The misclassification error rate for the full logisitic regression model is 0.125, while the misclassification error rate for the stepwise model is 0.1275.

K-Nearest Neighbors

```
##     predicted
## true   0   1
##    0 319  21
##    1  49  11
```

For the K-Nearest Neighbors Model, we used k=5 clusters, this value was decided by trial and error, and it had the lowest misclassification error rate of the tried clusters. To fit the report in the required word and page limit, we will only show the model with the final chosen cluster size. TheK-Nearest Neighbors Model uses all the predictor variables and is used to see whether those variables are sufficient at predicting whether a wine can be classified as good or not. Compared to the other models, K-Nearest Neighbors has a misclassification error rate of 0.175.

Conclusion

Combining all the results we found, a relatively effective prediction model can be established that uses physicochemical properties and values to judge the quality of red wine. This model is only obtained from 11 limited variables of the data from Vinho Verde' red wine collected by a study conducted in 2009 by Cortez et al . In general, the amount of alcohol, sulphates, and volatile acidity are factors that most affect the quality of red wine, and the relationship between them is linear with the proof from the Correlation and Pairwise Plots. In addition, the variable of pH should be removed since it does not have a relevant effect on the red wine quality (with p-value = 0.000183 in Stepwise Regression), and this action can also help to save time for further research. Plus, with the comparison between all the error rates we calculated, the misclassification error rates for LDA, Full Logistic regression, Stepwise Logistic regression, and K-Nearest Neighbors clustering are 0.1275, 0.125, 0.1275, and 0.1275 respectively, and this shows that the reliability of all these methods are fairly similar.

The result could be a realizable benefit for the wine industry. Red wine producers can pay more attention to control the value of red wine on these variables, so as to make the quality of red wine better with the higher price, which facilitates them to obtain a higher commercial value. Nevertheless, there are some deviations and limitations in this study. There are only a few physicochemical properties that are included in the study, and more data and variables need to be introduced if the models are to be more precise and complex. At the same time, it is hoped that future research can give a more detailed, accurate and standard answer.

Works Cited

https://archive.ics.uci.edu/ml/datasets/wine+quality (https://archive.ics.uci.edu/ml/datasets/wine+quality)

https://winefolly.com/deep-dive/vinho-verde-the-perfect-poolside-wine-from-portugal/ (https://winefolly.com/deep-dive/vinho-verde-the-perfect-poolside-wine-from-portugal/)

https://www.sciencedirect.com/science/article/abs/pii/S0167923609001377?via%3Dihub (https://www.sciencedirect.com/science/article/abs/pii/S0167923609001377?via%3Dihub)

http://www.sthda.com/english/articles/37-model-selection-essentials-in-r/154-stepwise-regression-essentials-in-r/ (http://www.sthda.com/english/articles/37-model-selection-essentials-in-r/154-stepwise-regression-essentials-in-r/)

https://topepo.github.io/caret/variable-importance.html (https://topepo.github.io/caret/variable-importance.html)

http://www.sthda.com/english/articles/36-classification-methods-essentials/150-stepwise-logistic-regression-essentials-in-r/ (http://www.sthda.com/english/articles/36-classification-methods-essentials/150-stepwise-logistic-regression-essentials-in-r/)

Session Information

```
sessionInfo()
```

```
## R version 4.0.0 (2020-04-24)
## Platform: x86_64-apple-darwin17.0 (64-bit)
## Running under: macOS Mojave 10.14.6
##
## Matrix products: default
## BLAS:   /Library/Frameworks/R.framework/Versions/4.0/Resources/lib/libRblas.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/4.0/Resources/lib/libRlapack.dylib
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] gridExtra_2.3       rpart_4.1-15        class_7.3-16
##  [4] leaps_3.1           randomForest_4.6-14 corrplot_0.84
##  [7] caret_6.0-86        lattice_0.20-41     GGally_2.0.0
## [10] MASS_7.3-51.5       forcats_0.5.0       stringr_1.4.0
## [13] dplyr_0.8.5         purrr_0.3.4         readr_1.3.1
## [16] tidyr_1.0.3         tibble_3.0.1        ggplot2_3.3.0
## [19] tidyverse_1.3.0
##
## loaded via a namespace (and not attached):
##  [1] httr_1.4.1           jsonlite_1.6.1       splines_4.0.0
##  [4] foreach_1.5.0        prodlim_2019.11.13   modelr_0.1.7
##  [7] assertthat_0.2.1     stats4_4.0.0         cellranger_1.1.0
## [10] ipred_0.9-9          pillar_1.4.4         backports_1.1.7
## [13] glue_1.4.1           pROC_1.16.2          digest_0.6.25
## [16] RColorBrewer_1.1-2   rvest_0.3.5          colorspace_1.4-1
## [19] recipes_0.1.13       htmltools_0.4.0      Matrix_1.2-18
## [22] plyr_1.8.6           timeDate_3043.102    pkgconfig_2.0.3
## [25] broom_0.5.6          haven_2.2.0          scales_1.1.1
## [28] gower_0.2.2          lava_1.6.7           farver_2.0.3
## [31] generics_0.0.2       ellipsis_0.3.0       withr_2.2.0
## [34] nnet_7.3-13          cli_2.0.2            survival_3.1-12
## [37] magrittr_1.5         crayon_1.3.4         readxl_1.3.1
## [40] evaluate_0.14        fs_1.4.1             fansi_0.4.1
## [43] nlme_3.1-147         xml2_1.3.2           tools_4.0.0
## [46] data.table_1.12.8    hms_0.5.3            lifecycle_0.2.0
## [49] munsell_0.5.0        reprex_0.3.0         compiler_4.0.0
## [52] rlang_0.4.6          grid_4.0.0           iterators_1.0.12
## [55] rstudioapi_0.11      labeling_0.3         rmarkdown_2.1
## [58] gtable_0.3.0         ModelMetrics_1.2.2.2 codetools_0.2-16
## [61] DBI_1.1.0            reshape_0.8.8        reshape2_1.4.4
## [64] R6_2.4.1             lubridate_1.7.8      knitr_1.28
## [67] stringi_1.4.6        Rcpp_1.0.4.6         vctrs_0.3.0
## [70] dbplyr_1.4.3         tidyselect_1.1.0     xfun_0.13
```

Appendix

```
options(warn=-1)
suppressMessages(library('tidyverse'))
suppressMessages(library('MASS'))
suppressMessages(library('GGally'))
suppressMessages(library('caret'))
suppressMessages(library('corrplot'))
suppressMessages(library('randomForest'))
suppressMessages(library('leaps'))
suppressMessages(library('class'))
suppressMessages(library('rpart'))
suppressMessages(library('utils'))
suppressMessages(library('gridExtra'))

set.seed(0)

wine = read.csv("~/Downloads/winequality-red.csv")
par(mfrow=c(3,4))
wine_data = wine
wine_data$quality = as.factor(wine_data$quality)
wine_data$good.wine = ifelse(wine$quality > 6, 1, 0)

gg = ggplot(data=wine) + theme_minimal()
g1 = gg + geom_histogram(aes(x=fixed.acidity), binwidth=0.1, fill='red', color='black')
g2 = gg + geom_histogram(aes(x=volatile.acidity), binwidth=0.1, fill='red', color='blac
k')
g3 = gg + geom_histogram(aes(x=citric.acid), fill='red', color='black')
g4 = gg + geom_histogram(aes(x=residual.sugar), binwidth=0.1, fill='red', color='black')
g5 = gg + geom_histogram(aes(x=chlorides), fill='red', color='black')
g6 = gg + geom_histogram(aes(x=free.sulfur.dioxide), fill='red', color='black')
g7 = gg + geom_histogram(aes(x=total.sulfur.dioxide), fill='red', color='black')
g8 = gg + geom_histogram(aes(x=density), fill='red', color='black')
g9 = gg + geom_histogram(aes(x=pH), binwidth=0.1, fill='red', color='black')
g10 = gg + geom_histogram(aes(x=sulphates), binwidth=0.1, fill='red', color='black')
g11 = gg + geom_histogram(aes(x=alcohol), binwidth=0.1, fill='red', color='black')
g12 = gg + geom_histogram(aes(x=quality), binwidth=1, fill='red', color='black')

suppressMessages(grid.arrange(g1,g2,g3,g4,g5,g6,g7,g8,g9,g10,g11,g12, nrow=4))
ggpairs(wine_data[,1:12], aes(color = quality), lower=list(combo=wrap("facethist",binwid
th=0.05)), progress=FALSE,
        title='Pairwise plots and Density Curves of All Variables Faceted by Quality') +
theme_minimal()
featurePlot(x=wine_data[,1:11], y=wine_data[,12], plot ='box',
            scales = list(y = list(relation='free'), x = list(relation='free')), pch =
'.',
            auto.key = list(columns = 3), labels = c('Quality', ''), layout = c(4,3))
corrplot(cor(wine), method='square', title='Correlation Matrix for all the Variables')
wine_random_forest = randomForest(factor(quality)~.-good.wine, data=wine_data, ntree=160
)
var_importance = importance(wine_random_forest)
importance_data = data.frame(variable = row.names(var_importance),Importance = round(var
_importance[ ,1],2))

importance_data$rank = paste('#',dense_rank(desc(importance_data$Importance)))
```

```
ggplot(importance_data, aes(x = reorder(variable, -Importance), y = Importance, fill = I
mportance)) +
       geom_bar(stat='identity') + labs(x='Variable', y='Importance', legend='Importanc
e',
                                        title='Importance of all Variables for Predictin
g Quality') + theme_minimal() +
       theme(axis.text.x = element_text(angle = 90)) + scale_fill_gradient(high='black',
low='red')
# regression diagnostics
par(mfrow=c(2,2))

reg_model = lm(quality~., data=wine)
plot(reg_model)
summary(reg_model)$r.squared
# split data into training and test sets

n = nrow(wine_data)
i = sample(1:n, round(0.75*n))

train_data = wine_data[i,]
test_data = wine_data[-i,]
# stepwise regression

stepwise_model = regsubsets(quality~., data=wine, nvmax=11)
model_summary = summary(stepwise_model)

data.frame(adjr2 = which.max(model_summary$adjr2),
           mallows = which.min(model_summary$cp),
           bic = which.min(model_summary$bic))

# for simplicity sake we take BIC
model_select = regsubsets(quality~., data=wine, nvmax=6)
summary(model_select)$outmat

final_reg_model = lm(quality~alcohol+volatile.acidity+sulphates+total.sulfur.dioxide+chl
orides+pH, data=wine)
summary(final_reg_model)$r.squared
par(mfrow=c(2,2))
plot(final_reg_model)
# lda

lda_model = lda(good.wine~.-quality, data=train_data)
lda_model$means


(lda_conf = table(true=test_data$good.wine, predicted=predict(lda_model, test_data)$clas
s))
lda_error = (lda_conf[1,2]+lda_conf[2,1])/sum(lda_conf)
# logisitic regression
par(mfrow=c(2,2))

logi_model = glm(good.wine~.-quality, data=train_data, family='binomial')
coef(logi_model %>% stepAIC(trace=FALSE))
```

```
stepwise_logi_model = glm(good.wine~density+chlorides+residual.sugar+total.sulfur.dioxid
e+
                          fixed.acidity+volatile.acidity+sulphates+alcohol, data=train_d
ata, family='binomial')

logi_pred = predict(logi_model, test_data, type='response')
stepwise_pred = predict(stepwise_logi_model, test_data, type='response')

logi_conf = table(true=test_data$good.wine, predicted = ifelse(logi_pred > 0.5, 1, 0))
stepwise_conf = table(true=test_data$good.wine, predicted = ifelse(stepwise_pred > 0.5,
1, 0))

logi_error = (logi_conf[1,2]+logi_conf[2,1])/sum(logi_conf)
stepwise_error = (stepwise_conf[1,2]+stepwise_conf[2,1])/sum(stepwise_conf)

logi_conf
stepwise_conf
# knn
k=5

knn_model = knn(train=train_data[,1:11], test=test_data[,1:11], cl=train_data$good.wine,
k=k)
knn_conf = table(true = test_data$good.wine, predicted = knn_model)
knn_error = (knn_conf[1,2] + knn_conf[2,1])/sum(knn_conf)
knn_conf
sessionInfo()
```