Group_13_Analysis

Shuyin Chen, Xuran Wang, Yingying Zhuo, Yunfei Chen, Yuxuan Li

Library

```
library(ggplot2)
library(readr)
library(knitr)
library(stringr)
library(jtools)
library(GGally)
library(gridExtra)
library(factoextra)
```

Wrangling of data

```
data <- read.csv("dataset13.csv")

str(data)
```

```
'data.frame':   1145 obs. of  8 variables:
 $ country_of_origin   : chr  "Myanmar" "Uganda" "Ethiopia" "Mexico" ...
 $ aroma               : num  7.25 8.33 8.42 7.17 7.75 7.92 7.92 7.83 7 7.33 ...
 $ flavor              : num  7.42 7.92 8 7.08 7.67 7.75 7.83 7.67 6.83 7.33 ...
 $ acidity             : num  7.5 7.92 8 7.25 7.5 7.75 7.67 7.58 7.17 7.5 ...
 $ category_two_defects: int  4 1 7 3 5 0 1 2 2 1 ...
 $ altitude_mean_meters: num  1219 1600 1700 1300 1880 ...
 $ harvested           : int  2015 2013 2014 2012 2012 2014 NA 2015 2013 2013 ...
 $ Qualityclass        : chr  "Poor" "Good" "Good" "Poor" ...
```

```
summary(data)
```

```
 country_of_origin      aroma           flavor         acidity
 Length:1145        Min.   :0.000   Min.   :0.000   Min.   :0.000
 Class :character   1st Qu.:7.420   1st Qu.:7.330   1st Qu.:7.330
 Mode  :character   Median :7.580   Median :7.580   Median :7.500
                    Mean   :7.571   Mean   :7.521   Mean   :7.536
                    3rd Qu.:7.750   3rd Qu.:7.750   3rd Qu.:7.750
                    Max.   :8.750   Max.   :8.670   Max.   :8.580

 category_two_defects altitude_mean_meters   harvested    Qualityclass
 Min.   : 0.000       Min.   :     1       Min.   :2010   Length:1145
 1st Qu.: 0.000       1st Qu.:  1100       1st Qu.:2012   Class :character
 Median : 2.000       Median :  1311       Median :2014   Mode  :character
 Mean   : 3.673       Mean   :  1851       Mean   :2014
 3rd Qu.: 5.000       3rd Qu.:  1600       3rd Qu.:2015
 Max.   :55.000       Max.   :190164       Max.   :2018
                      NA's   :201          NA's   :60
```

```r
#Remove missing values
data <- na.omit(data)
data <- data[data$altitude_mean_meters <= 8848, ]
str(data)
```

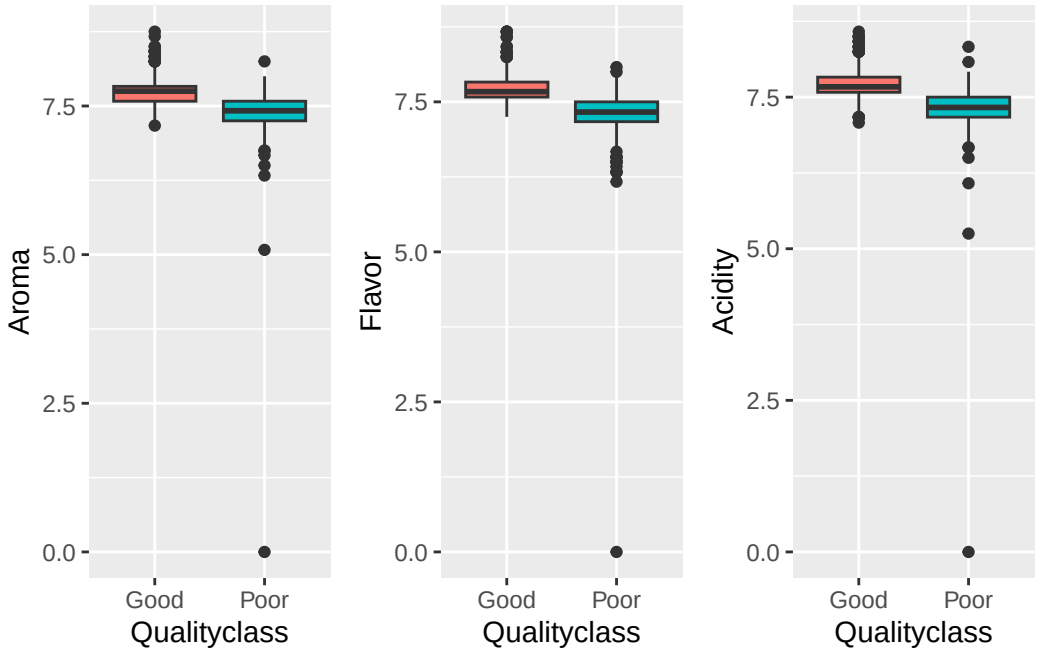
```
'data.frame':   931 obs. of  8 variables:
 $ country_of_origin   : chr  "Myanmar" "Uganda" "Ethiopia" "Mexico" ...
 $ aroma               : num  7.25 8.33 8.42 7.17 7.75 7.92 7.83 7 7.33 7.67 ...
 $ flavor              : num  7.42 7.92 8 7.08 7.67 7.75 7.67 6.83 7.33 7.58 ...
 $ acidity             : num  7.5 7.92 8 7.25 7.5 7.75 7.58 7.17 7.5 7.67 ...
 $ category_two_defects: int  4 1 7 3 5 0 2 2 1 0 ...
 $ altitude_mean_meters: num  1219 1600 1700 1300 1880 ...
 $ harvested           : int  2015 2013 2014 2012 2012 2014 2015 2013 2013 2012 ...
 $ Qualityclass        : chr  "Poor" "Good" "Good" "Poor" ...
 - attr(*, "na.action")= 'omit' Named int [1:210] 7 18 19 26 28 32 38 39 41 49 ...
  ..- attr(*, "names")= chr [1:210] "7" "18" "19" "26" ...
```

```r
#Convert Qualityclass category variables to 0, 1: Good=1, Poor=0
data$Qualityclass_binary <- ifelse(data$Qualityclass == "Good", 1, 0)
```

Data visualisation

Plotting box plots of aroma, flavour and acidity.

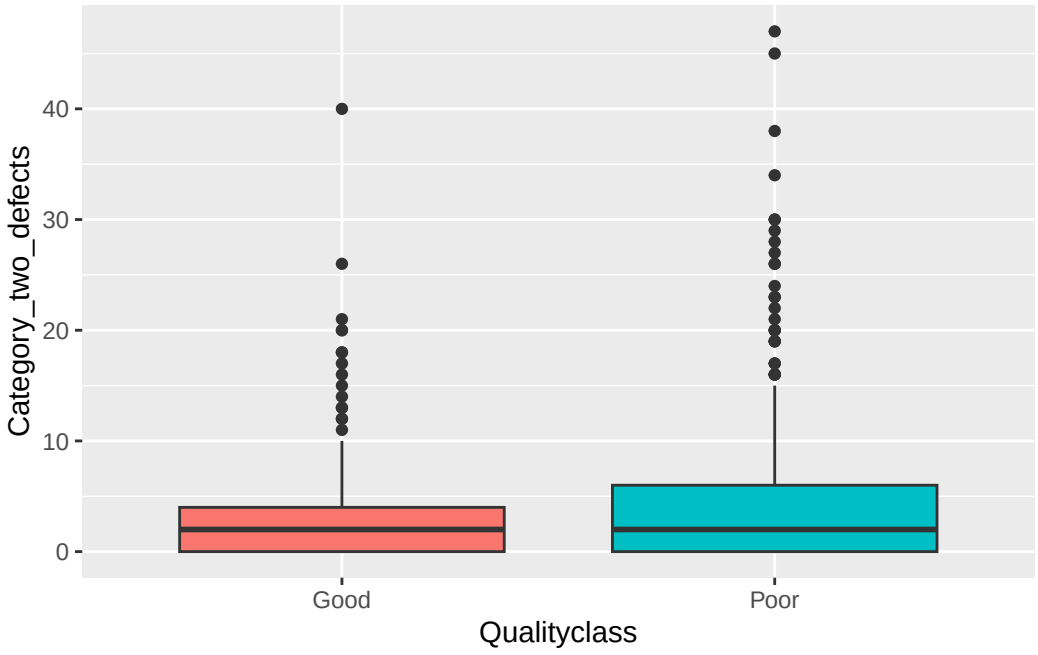
```
#Boxplot
g1 <- ggplot(data = data, aes(x = Qualityclass, y = aroma, fill = Qualityclass)) +
  geom_boxplot() +
  labs(x = "Qualityclass", y = "Aroma")+
  theme(legend.position = "none")
g2 <- ggplot(data = data, aes(x = Qualityclass, y = flavor, fill = Qualityclass)) +
  geom_boxplot() +
  labs(x = "Qualityclass", y = "Flavor")+
  theme(legend.position = "none")
g3 <- ggplot(data = data, aes(x = Qualityclass, y = acidity, fill = Qualityclass)) +
  geom_boxplot() +
  labs(x = "Qualityclass", y = "Acidity")+
  theme(legend.position = "none")
grid.arrange(g1,g2,g3, ncol=3)
```



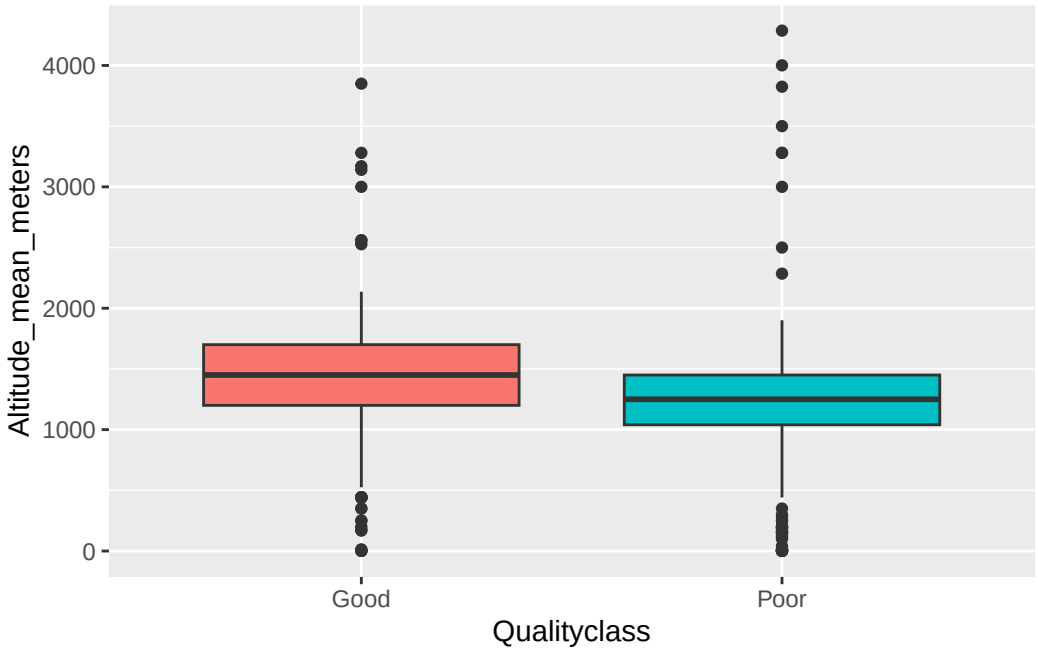
Plotting boxplots of Category_two_defects and Altitude_mean_meters.

```
#Boxplot
ggplot(data = data, aes(x = Qualityclass, y = category_two_defects, fill = Qualityclass))
  geom_boxplot() +
  labs(x = "Qualityclass", y = "Category_two_defects")+
```

```
  theme(legend.position = "none")
```



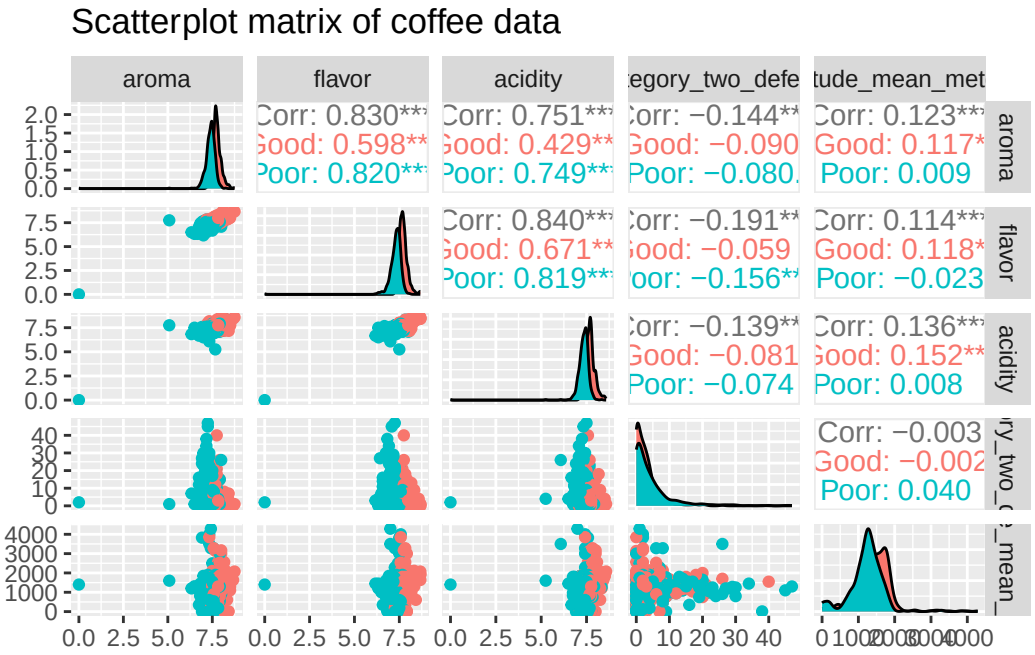

```
ggplot(data = data, aes(x = Qualityclass, y = altitude_mean_meters, fill = Qualityclass))
  geom_boxplot() +
  labs(x = "Qualityclass", y = "Altitude_mean_meters")+
  theme(legend.position = "none")
```



Check correlations, distribution and print correlation coefficient

```
#scartterplot
par(mfrow = c(1, 1))

# Check correlations (as scatterplots), distribution and print correlation coefficient
ggpairs(data[,2:6],
        title = "Scatterplot matrix of coffee data",
        mapping = aes(color = data$Qualityclass))
```



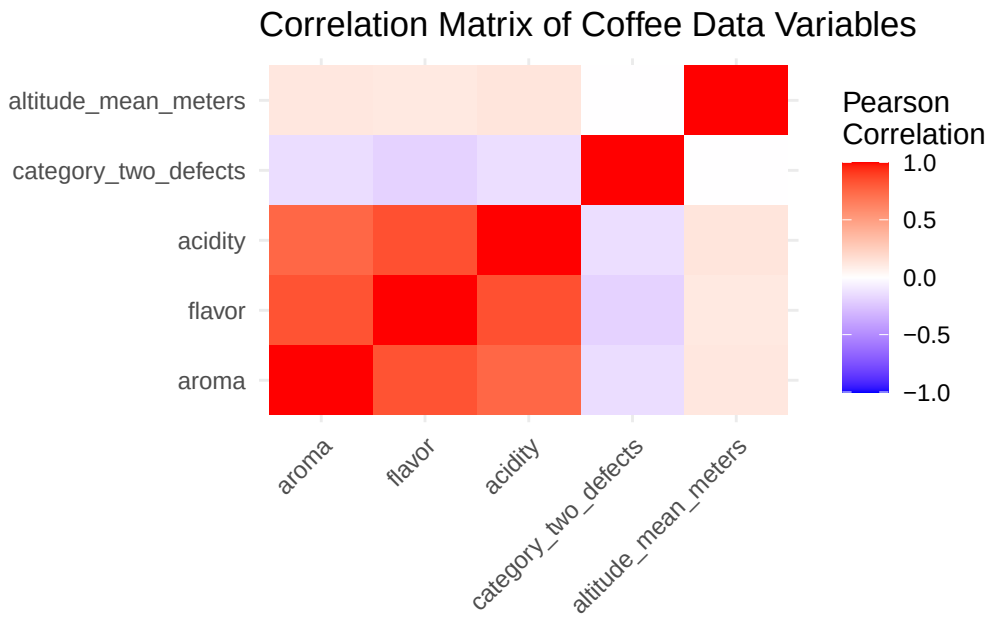
Create a correlation matrix as heatmap

```
numeric_vars <- data[,2:6]
cor_matrix <- cor(numeric_vars)
print(cor_matrix)
```

```
                          aroma     flavor    acidity category_two_defects
aroma                 1.0000000  0.8299135  0.7509670         -0.143668998
flavor                0.8299135  1.0000000  0.8399379         -0.191079851
acidity               0.7509670  0.8399379  1.0000000         -0.138939928
category_two_defects -0.1436690 -0.1910799 -0.1389399          1.000000000
altitude_mean_meters  0.1234430  0.1144632  0.1363216         -0.002622067
                     altitude_mean_meters
aroma                         0.123442998
flavor                        0.114463243
acidity                       0.136321627
category_two_defects         -0.002622067
altitude_mean_meters          1.000000000
```

```
cor_melt <- reshape2::melt(cor_matrix)
#
```

```
ggplot(data = cor_melt, aes(x=Var1, y=Var2, fill=value)) +
  geom_tile() +
  scale_fill_gradient2(low = "blue", high = "red", mid = "white",
                       midpoint = 0, limit = c(-1,1), space = "Lab",
                       name="Pearson\nCorrelation") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust=1)) +
  labs(x='', y='', title='Correlation Matrix of Coffee Data Variables')
```



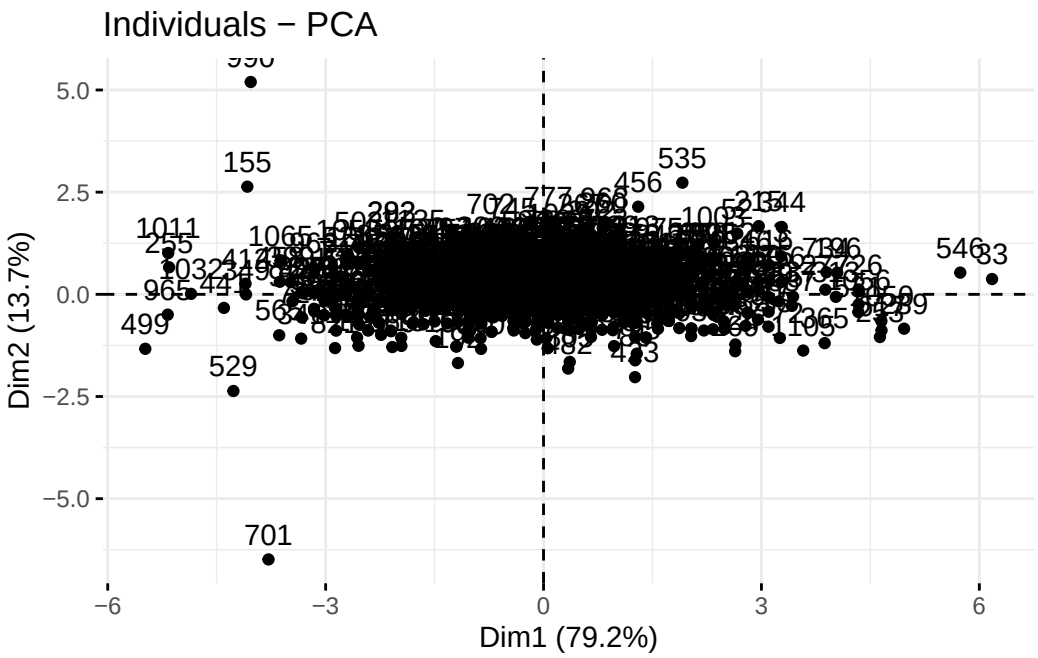
Delete the outliers for the variables aroma, flavor and acidity.

```
# delete Aroma == 0
data <- data[data$aroma != 0, ]
data <- data[data$flavor != 0, ]
data <- data[data$acidity != 0, ]
```

Due to the significant multicollinearity between these three variables, a principal component analysis was performed to obtain the loading matrix and the scree plot.

```
#pca for aroma flavor acidity
varable3 <- scale(data[, 2:4])
```

```r
pca_result <- prcomp(varable3, center = TRUE, scale. = TRUE)
fviz_pca_ind(pca_result)
```



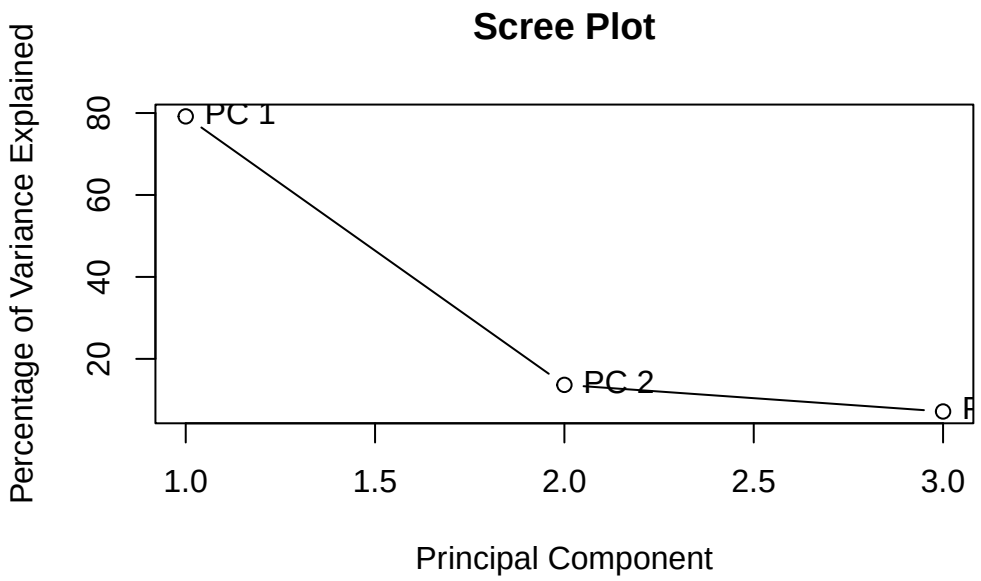

```r
# Explain the variance
print(summary(pca_result))
```

```
Importance of components:
                          PC1    PC2    PC3
Standard deviation     1.5412 0.6401 0.4635
Proportion of Variance 0.7918 0.1366 0.0716
Cumulative Proportion  0.7918 0.9284 1.0000
```

```r
# Principal component loading
print(pca_result$rotation)
```

```
              PC1         PC2        PC3
aroma    0.5613942  0.73121779 -0.3875011
flavor   0.6027161 -0.04041114  0.7969318
acidity  0.5670713 -0.68094603 -0.4634033
```

```r
variance <- pca_result$sdev^2
variance_percentage <- variance / sum(variance) * 100
plot(variance_percentage, type = "b", xlab = "Principal Component", ylab = "Percentage of
text(variance_percentage, labels = paste("PC", 1:length(variance_percentage)), pos = 4)
```



According to the loading matrix, it can be seen that the first principal component explains 79.18% of these three variables, so the first principal component is selected as the variable of the model. It can be interpreted as the average of the three variables aroma, flavour, and acidity based on the score coefficient of the first principal component.

```r
## name pca1 as characteristics
pca1 <- as.data.frame(pca_result$x[, 1])
colnames(pca1) <- "characteristics"
head(pca1)
```

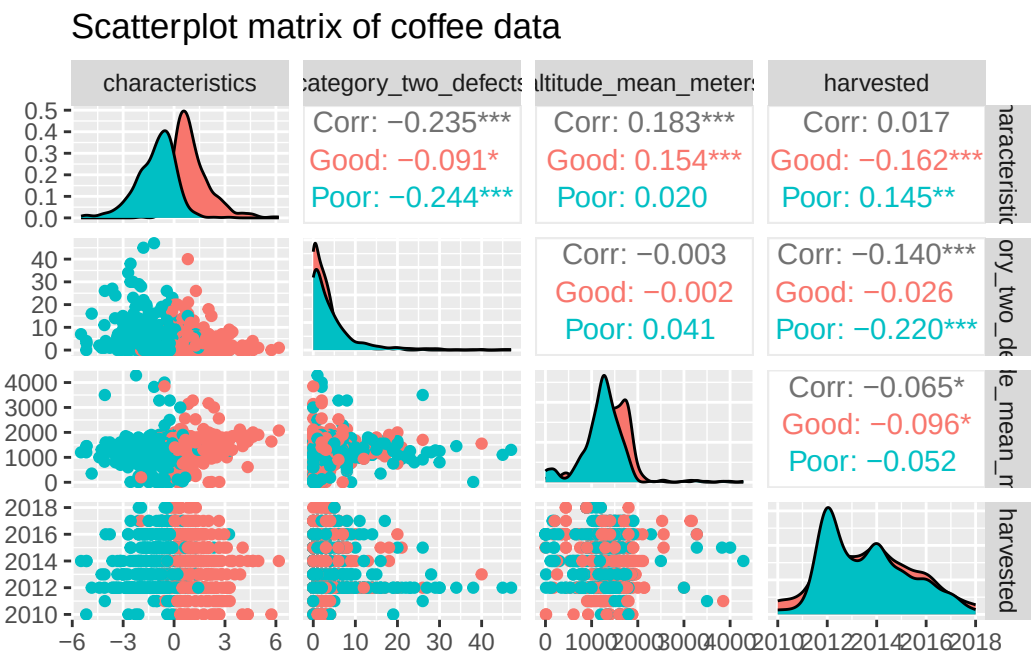
```
  characteristics
1      -0.8657112
2       2.8109981
3       3.2703451
4      -2.0985665
5       0.5177875
```

```
6       1.4328658
```

```
## Re-verify the correlation between the variables
data1 <- cbind(characteristics = pca1, data[, 5:9])
str(data1)
```

```
'data.frame':   930 obs. of  6 variables:
 $ characteristics     : num  -0.866 2.811 3.27 -2.099 0.518 ...
 $ category_two_defects: int  4 1 7 3 5 0 2 2 1 0 ...
 $ altitude_mean_meters: num  1219 1600 1700 1300 1880 ...
 $ harvested           : int  2015 2013 2014 2012 2012 2014 2015 2013 2013 2012 ...
 $ Qualityclass        : chr  "Poor" "Good" "Good" "Poor" ...
 $ Qualityclass_binary : num  0 1 1 0 1 1 1 0 0 1 ...
```

```
ggpairs(data1[,1:4],
        title = "Scatterplot matrix of coffee data",
        mapping = aes(color = data1$Qualityclass))
```



Based on the scatterplot matrix, it can be seen that there is no multicollinearity in the newly merged dataset and binary logistic regression can be performed.

$$characteristics_{\mathrm{i}} = 0.56 \cdot aroma_{\mathrm{i}} + 0.60 \cdot flavor_{\mathrm{i}} - 0.57 \cdot acidity_{\mathrm{i}}$$

Creating model

```
#formula
formula1 <- as.formula(paste("Qualityclass_binary ~", paste(names(data1)[1:4], collapse =

# Fit model for the Qualityclass_binary
logistic_model1 <- glm(formula1, data = data1, family = binomial)

summary(logistic_model1)
```

```
Call:
glm(formula = formula1, family = binomial, data = data1)

Coefficients:
                       Estimate Std. Error z value Pr(>|z|)
(Intercept)          -1.582e+02  1.152e+02  -1.374   0.1695
characteristics       2.888e+00  2.052e-01  14.074   <2e-16 ***
category_two_defects  6.206e-04  2.597e-02   0.024   0.9809
altitude_mean_meters  5.435e-04  2.214e-04   2.455   0.0141 *
harvested             7.819e-02  5.717e-02   1.368   0.1714
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1289.15  on 929  degrees of freedom
Residual deviance:  555.02  on 925  degrees of freedom
AIC: 565.02

Number of Fisher Scoring iterations: 7
```

```
# Filter variables using stepwise regression
stepwise_model <- step(logistic_model1)
```

```
Start:  AIC=565.02
```

```
Qualityclass_binary ~ characteristics + category_two_defects +
    altitude_mean_meters + harvested

                       Df Deviance     AIC
- category_two_defects  1   555.02  563.02
- harvested             1   556.89  564.89
<none>                      555.02  565.02
- altitude_mean_meters  1   560.96  568.96
- characteristics       1  1233.07 1241.07

Step:  AIC=563.02
Qualityclass_binary ~ characteristics + altitude_mean_meters +
    harvested

                       Df Deviance     AIC
- harvested             1   556.90  562.90
<none>                      555.02  563.02
- altitude_mean_meters  1   560.97  566.97
- characteristics       1  1254.96 1260.96

Step:  AIC=562.9
Qualityclass_binary ~ characteristics + altitude_mean_meters

                       Df Deviance     AIC
<none>                      556.90  562.90
- altitude_mean_meters  1   562.08  566.08
- characteristics       1  1257.42 1261.42
```

```r
summary(stepwise_model)
```

```
Call:
glm(formula = Qualityclass_binary ~ characteristics + altitude_mean_meters,
    family = binomial, data = data1)

Coefficients:
                      Estimate Std. Error z value Pr(>|z|)
(Intercept)         -0.7144533  0.3089404  -2.313   0.0207 *
characteristics      2.8774852  0.2036705  14.128   <2e-16 ***
altitude_mean_meters 0.0005013  0.0002187   2.292   0.0219 *
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1289.1  on 929  degrees of freedom
Residual deviance:  556.9  on 927  degrees of freedom
AIC: 562.9

Number of Fisher Scoring iterations: 7
```

```r
model_deviance <- deviance(stepwise_model)
null_deviance <- deviance(glm(Qualityclass_binary ~ 1, family = binomial, data = data1))
# Calculate R_square
R_square <- 1 - (model_deviance / null_deviance)
R_square
```

```
[1] 0.5680107
```

```r
# Output model results
model_summary <- summary(stepwise_model)

coefficients_table <- model_summary$coefficients

write.csv(coefficients_table, "logistic_model_summary.csv", row.names = TRUE)
```

According to our analysis, we can get the model

$$Qualityclass\_binary_i = \beta_0 + \beta_1 \cdot characteristics_i + \beta_2 \cdot altitude\_mean\_meters_i + \varepsilon_i$$

where

1. the interceptβ_0 is the expected value of $Qualityclass\_binary_i$,when all independent variables are zero.
2. $\beta_1 \cdot characteristics_i$ is This term represents the effect of the $i_th$observation's characteristics on the quality class.
3. $\beta_2 \cdot altitude\_mean\_meters_i$ is the effect of the i_th observation's average altitude (measured in meters) on the quality class.
4. ε_i is unobserved factors that affect the quality class of the i_th observation.