Coffee Analysis

Sean,Ethan,Daniel,Nhi

2024-05-31

Introduction

The Coffee Quality Data (CQI May-2023) dataset, maintained by the Coffee Quality Institute (CQI), offers a comprehensive resource for understanding and analyzing various aspects of coffee production and quality. Founded in 1996, CQI is a non-profit organization headquartered in California, USA, dedicated to improving the quality and value of coffee worldwide. Through research, training, and certification programs, CQI collaborates with coffee growers, processors, roasters, and other stakeholders to enhance coffee quality standards, promote sustainability, and support the development of the specialty coffee industry.

This dataset includes 41 columns and 162 entries, with detailed information about the origin, processing, and sensory evaluation of coffee. Key variables include Country of Origin, Region, Farm Name, Altitude, Variety, and various coffee quality scores such as Flavor, Aroma, Sweetness, and Overall Score. This rich dataset provides an excellent opportunity for conducting a thorough data analysis to uncover insights into the factors that influence coffee quality.

As a group of students embarking on a data analysis project, this dataset presents an ideal opportunity to apply and develop various analytical skills. The analysis will involve working with data to create visualizations, perform statistical tests such as Z tests, T tests, and Chi-square tests, and apply linear regression and Principal Component Analysis (PCA). These techniques will help explore relationships between different variables, identify patterns, and draw meaningful conclusions about the factors affecting coffee quality.

The aim of this report is to provide a detailed exploration of the Coffee Quality Data (CQI May-2023) dataset, utilizing various data analysis techniques to gain insights into coffee quality and production practices. By doing so, this report will contribute to a deeper understanding of the coffee industry and support efforts to improve coffee quality and sustainability.

To input the data

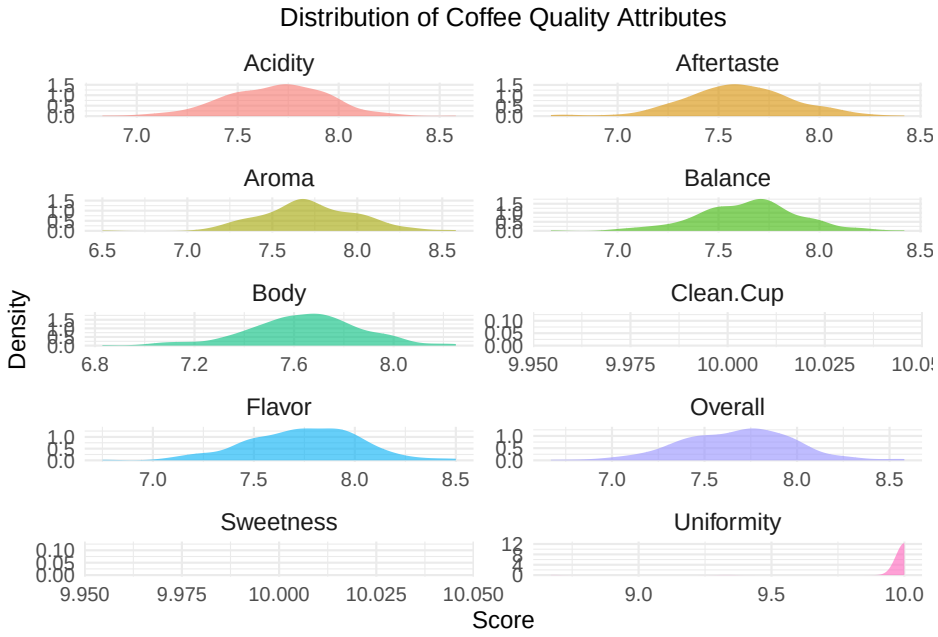
```
df<- read.csv(file = "H:/Downloads/coffee_quality.csv",
                      stringsAsFactors = FALSE)
```

General Analysis

Since there are many Sensory Score, it would be interesting to see their distribution so that we can find the correct testing method for later analysis:

```
# List of columns to plot
quality_columns <- c("Aroma", "Flavor", "Aftertaste", "Acidity", "Body", "Balance",
                     "Uniformity", "Clean.Cup", "Sweetness", "Overall")

# Reshape the df to long format for easy plotting
library(tidyr)
coffee_long <- df %>%
  select(all_of(quality_columns)) %>%
```

```
  pivot_longer(cols = everything(), names_to = "Quality_Attribute", values_to = "Score")

ggplot(coffee_long, aes(x = Score, fill = Quality_Attribute)) +
  geom_density(alpha = 0.6, color = NA) +
  facet_wrap(~ Quality_Attribute, scales = "free", ncol = 2) +
  labs(title = "Distribution of Coffee Quality Attributes",
       x = "Score",
       y = "Density") +
  theme_minimal() +
  theme(legend.position = "none",
        plot.title = element_text(hjust = 0.5),
        axis.text = element_text(size = 10),
        axis.title = element_text(size = 12),
        strip.text = element_text(size = 12))
```



We can see that here, for almost all the Sensory Score, they have the shape of normal distribution.

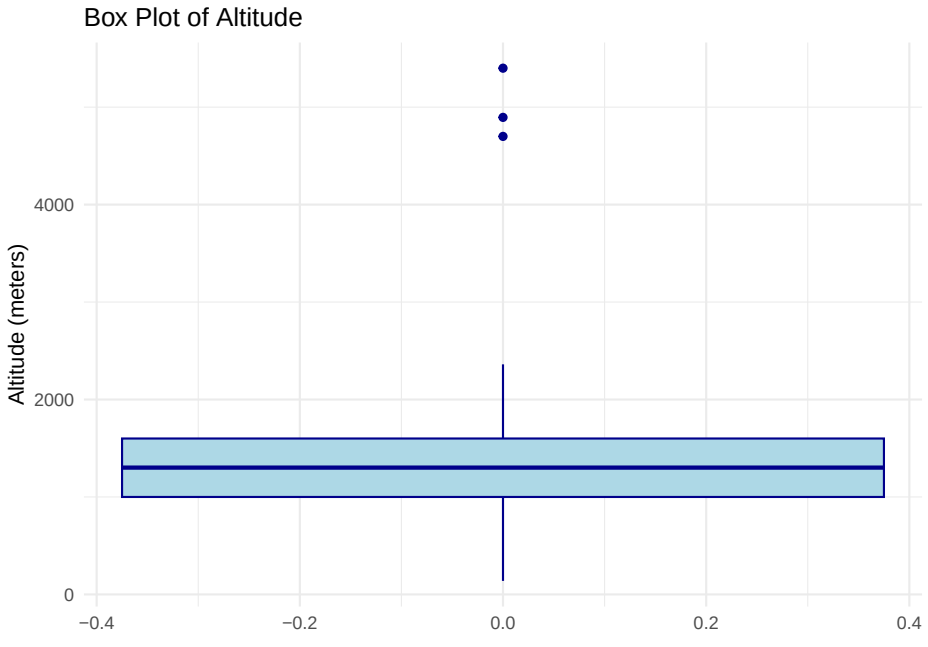
We want to see if there is a relationship between Altitude and Aroma. But since in the dataset, the Altitude has some values in a range instead of number, we decided to take the average if the value is in a range.

```
# Function to replace ranges with their averages
replace_range_with_average <- function(altitude) {
  if (grepl("-", altitude)) {
    # Split the range into two numbers
    range_values <- strsplit(altitude, "-")[[1]]
    # Convert the string values to numeric
    num_values <- as.numeric(range_values)
    # Calculate the average
    avg_value <- mean(num_values)
    # Return the average as a character string
    return(as.character(avg_value))
  } else {
    # Return the original value if it's not a range
```

```
    return(altitude)
  }
}

# Apply the function to the Altitude column
df$Altitude <- sapply(df$Altitude, replace_range_with_average)

# Convert the Altitude column to numeric for further analysis
df$Altitude <- as.numeric(df$Altitude)
head(df)
View(df)
```

Box plot for Altitude:

```
ggplot(df, aes(y = Altitude)) +
  geom_boxplot(fill = "lightblue", color = "darkblue") +
  labs(title = "Box Plot of Altitude",
       y = "Altitude (meters)") +
  theme_minimal()
```



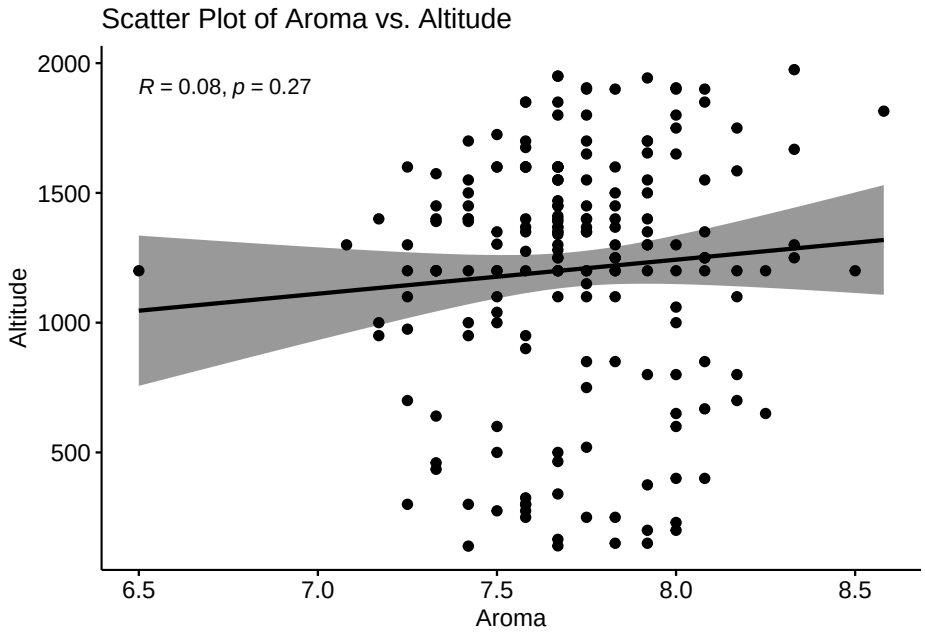
Here since there are 3 outliers that are much larger than the rest, I removed the 3 outliers to get better estimation on the relationship between the two variables.

```
df$Altitude <- as.numeric(df$Altitude)
df$Aroma <- as.numeric(df$Aroma)


correlation <- cor(df$Altitude, df$Aroma, method = "pearson", use = "complete.obs")
correlation
unique(df$Altitude)

df_filter <- subset(df, df$Altitude <2000)
```

```r
library("ggpubr")
```

```
## Warning: package 'ggpubr' was built under R version 4.3.3
```

```r
ggscatter(df_filter, x = "Aroma", y = "Altitude",
          add = "reg.line", conf.int = TRUE,
          cor.coef = TRUE, cor.method = "pearson",
          xlab = "Aroma", ylab = "Altitude") +
      ggtitle("Scatter Plot of Aroma vs. Altitude")
```



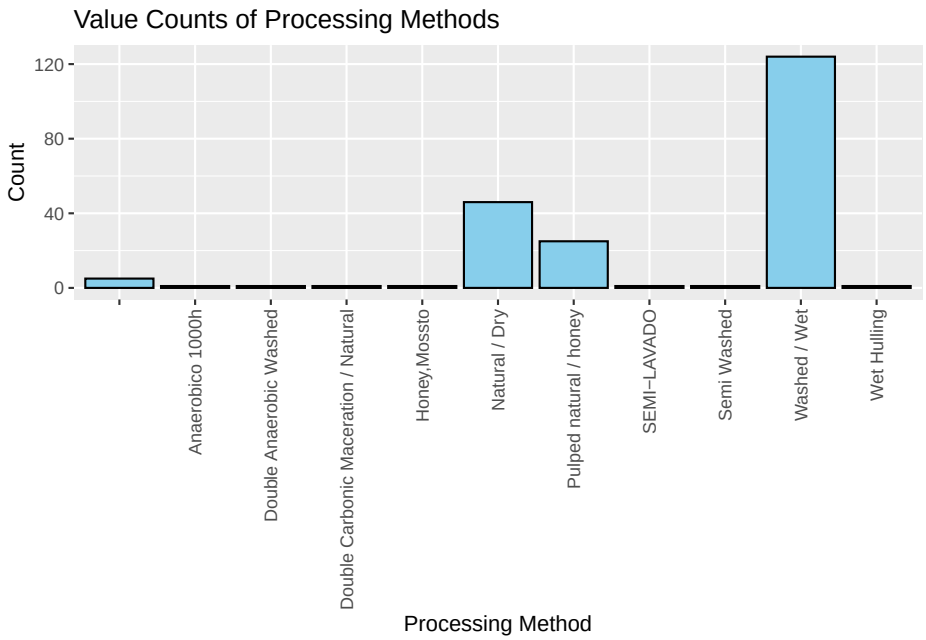
Since the correlation coefficient is 0.27, it indicates a weak positive relationship between altitude and aroma. This suggests that higher altitudes are associated with slightly higher aroma scores, but the relationship is not strong.

Moving on, now take a look on processing method:

```r
unique(df$Processing.Method)
library(dplyr)


processing_method_counts <- table(df$Processing.Method)
print(processing_method_counts)

# Convert value counts to a df frame
processing_method_counts_df <- as.data.frame(processing_method_counts)
names(processing_method_counts_df) <- c("Processing.Method", "Count")
```

```r
# Create a bar plot
ggplot(processing_method_counts_df, aes(x = Processing.Method, y = Count)) +
  geom_bar(stat = "identity", fill = "skyblue", color = "black") +
  labs(title = "Value Counts of Processing Methods",
```

```
       x = "Processing Method",
       y = "Count") +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1))
```



Here we can see the trend of Washed/Wet processing method among all of the processes.

Take the average of all Scores, grouped by Processing Method:

```
average_by_processing_method <- df %>%
  group_by(Processing.Method) %>%
  summarize(
    Avg_Aroma = mean(Aroma, na.rm = TRUE),
    Avg_Flavor = mean(Flavor, na.rm = TRUE),
    Avg_Aftertaste = mean(Aftertaste, na.rm = TRUE),
    Avg_Acidity = mean(Acidity, na.rm = TRUE),
    Avg_Body = mean(Body, na.rm = TRUE),
    Avg_Balance = mean(Balance, na.rm = TRUE),
    Avg_Uniformity = mean(Uniformity, na.rm = TRUE),
    Avg_Clean_Cup = mean(Clean.Cup, na.rm = TRUE),
    Avg_Sweetness = mean(Sweetness, na.rm = TRUE),
    Avg_Overall = mean(Overall, na.rm = TRUE)
  )

# Print the result
average_by_processing_method
```

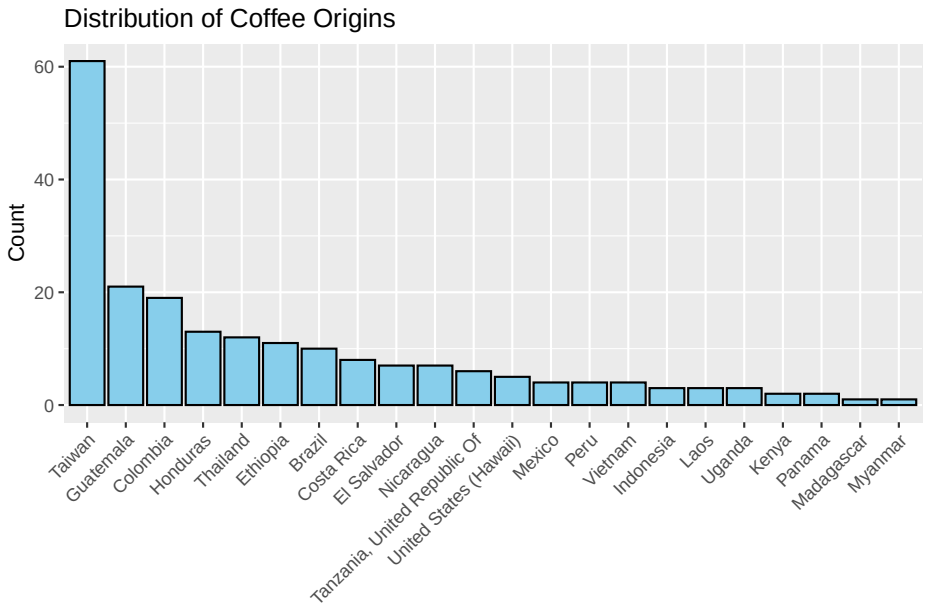
Here we just want to see the mean of each Sensory Score based on Processing Methods.

Count of Country of Origin and make a graph on it:

```
country_counts <- df %>%
  count(Country.of.Origin, sort = TRUE)
```

```r
# Print the result
print(country_counts)
```

```r
# Bar plot of Country of Origin counts
ggplot(country_counts, aes(x = reorder(Country.of.Origin, -n), y = n)) +
  geom_bar(stat = "identity", fill = "skyblue", color = "black") +
  labs(title = "Distribution of Coffee Origins",
       x = "Country of Origin",
       y = "Count") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



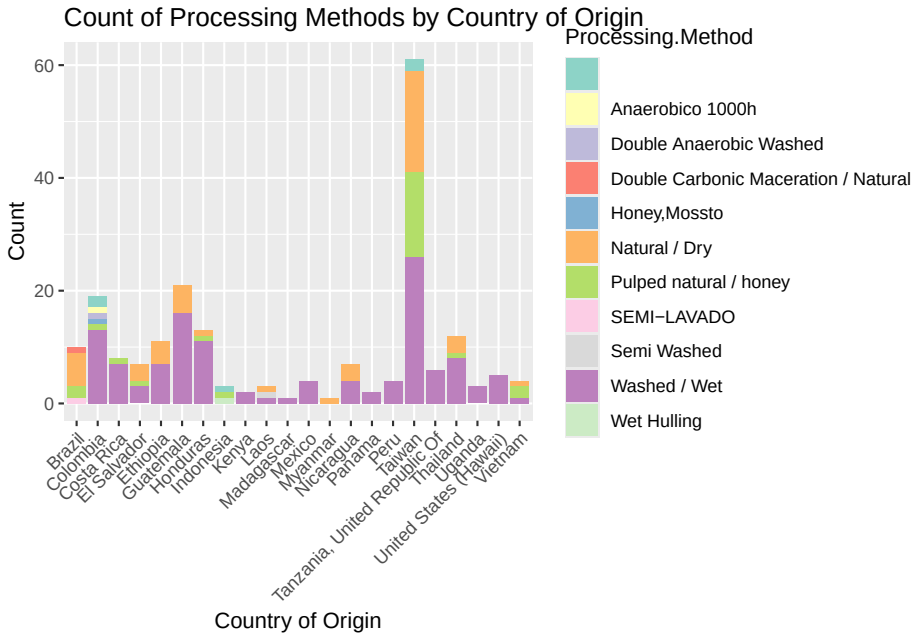
We can see Taiwan took the majority of data in this dataset.

Count of occurrences of each processing method for each country of origin:

```r
count_by_country_method <- df %>%
  group_by(Country.of.Origin, Processing.Method) %>%
  count()

# Print the result
print(count_by_country_method)
```

Stacked bar charts where each bar represents the count of processing methods for each country of origin

```r
library(ggplot2)

ggplot(count_by_country_method, aes(x = Country.of.Origin, y = n, fill = Processing.Method)) +
  geom_bar(stat = "identity") +
  labs(title = "Count of Processing Methods by Country of Origin",
       x = "Country of Origin",
       y = "Count") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  scale_fill_brewer(palette = "Set3")
```



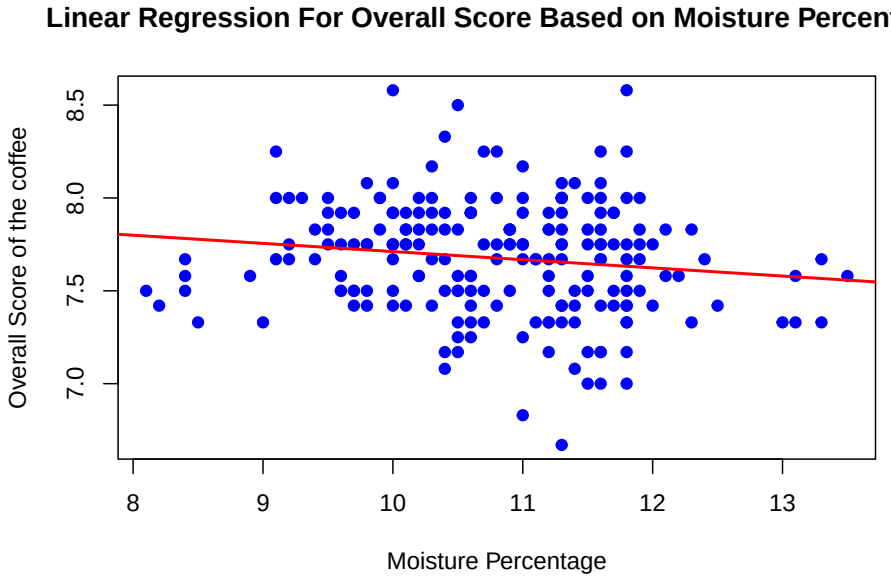
As you can see on the plot, for most of the bars, the grey color which stands for Washed / Wet dominates the most which means for almost all the countries, they prefer this processing method.

Moisture Analysis

Linear Regression of Overall and Moisture

I use Linear Regression to see if Moisture Percentage can predict the Overall Score of the coffee. I saw a anomaly in the data because its imposible to make a coffee with 0 moisture so i remove that outlier.

```
#To remove the outlier
df <- subset(df, df$Moisture.Percentage >2 )

# Visualize with Scatter plot
plot(df$Moisture.Percentage, df$Overall, main = "Linear Regression For Overall Score Based on Moisture
     xlab = "Moisture Percentage", ylab = "Overall Score of the coffee",
     pch = 19, col = "blue")

# Linear regression using Linear Model 'lm'
model <- lm(Overall ~ Moisture.Percentage, data = df)

# Plotting the line of best fit
abline(model, col = "red", lwd = 2)
```



Based on the result we conclude that Moisture does affect the Overall score of coffee, Eventhough the Relationship is weak.

Chi Square for Which country influence moisture level in the coffee

The reason why i use Chi Square is to see if there is a relation between Moisture Level and Country

```
# Im Creating Moisture_Level variable so when the Moisture Percentage is higher than the Mean of MP it
df <- df %>%
  mutate(Moisture_Level = ifelse(Moisture.Percentage < mean(Moisture.Percentage, na.rm = TRUE), "Low",

# I remove rows with missing values in 'Country of Origin' or 'Moisture_Level'
cleaned_data <- df %>%
  filter(!is.na(`Country.of.Origin`) & !is.na(Moisture_Level))

# I remove the row which have missing value in it to ensure data lengths are consistent
cleaned_data <- cleaned_data %>%
  drop_na(`Country.of.Origin`, Moisture_Level)

# Create a contingency table
contingency_table <- table(cleaned_data$`Country.of.Origin`, cleaned_data$Moisture_Level)
contingency_table
```

```
##
##                              High Low
##   Brazil                        9   1
##   Colombia                     17   2
##   Costa Rica                    3   4
##   El Salvador                   5   2
##   Ethiopia                      8   3
##   Guatemala                    14   7
##   Honduras                      3  10
##   Indonesia                     3   0
##   Kenya                         1   1
```

```
##   Laos                            2   1
##   Madagascar                      1   0
##   Mexico                          3   1
##   Myanmar                         0   1
##   Nicaragua                       3   4
##   Panama                          0   2
##   Peru                            3   1
##   Taiwan                         15  46
##   Tanzania, United Republic Of    3   3
##   Thailand                        8   4
##   Uganda                          2   1
##   United States (Hawaii)          2   3
##   Vietnam                         3   1
```

```
# Perform Chi-square test
chi_square_test <- chisq.test(contingency_table)
chi_square_test
```

```
##
##  Pearson's Chi-squared test
##
## data:  contingency_table
## X-squared = 55.789, df = 21, p-value = 5.42e-05
```

```
# Chi-square test results
chi_square_test$statistic
```

```
## X-squared
##  55.78895
```

```
chi_square_test$p.value
```

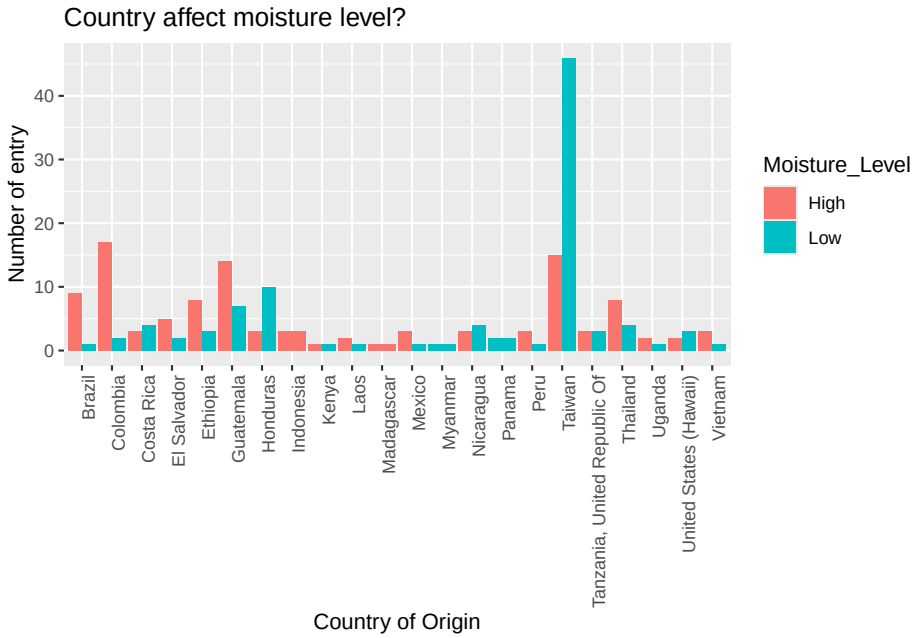
```
## [1] 5.420134e-05
```

```
chi_square_test$expected
```

```
##
##                                 High        Low
##   Brazil                   5.2427184  4.7572816
##   Colombia                 9.9611650  9.0388350
##   Costa Rica               3.6699029  3.3300971
##   El Salvador              3.6699029  3.3300971
##   Ethiopia                 5.7669903  5.2330097
##   Guatemala               11.0097087  9.9902913
##   Honduras                 6.8155340  6.1844660
##   Indonesia                1.5728155  1.4271845
##   Kenya                    1.0485437  0.9514563
##   Laos                     1.5728155  1.4271845
##   Madagascar               0.5242718  0.4757282
##   Mexico                   2.0970874  1.9029126
##   Myanmar                  0.5242718  0.4757282
```

```
##   Nicaragua                      3.6699029  3.3300971
##   Panama                         1.0485437  0.9514563
##   Peru                           2.0970874  1.9029126
##   Taiwan                        31.9805825 29.0194175
##   Tanzania, United Republic Of   3.1456311  2.8543689
##   Thailand                       6.2912621  5.7087379
##   Uganda                         1.5728155  1.4271845
##   United States (Hawaii)         2.6213592  2.3786408
##   Vietnam                        2.0970874  1.9029126
```

```r
# To Visualize the relation between Country of Origin and the Moisture Level
ggplot(cleaned_data, aes(x = `Country.of.Origin`, fill = Moisture_Level)) +
  geom_bar(position = "dodge") +
  labs(title = "Country affect moisture level?",
       x = "Country of Origin",
       y = "Number of entry",
       fill = "Moisture_Level") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



Based on the Chi Square Test

With a p-value of 4.692e-05, it is significantly smaller than the commonly chosen significance level of 0.05.

Since the p-value is less than 0.05, we reject the null hypothesis. Therefore, we conclude that there is a significant association between Country of Origin and Moisture_Level. In other words, they are not independent.

Based on this we can see that the country of origin does have an influence on the moisture level of the coffee samples in the dataset.

Chi Square for How method of coffee processing influence the Moisture Level of the coffee#

The reason why i use Chi Square is to see if there is a relation between Moisture Level and Processing Method. I use the same method as CS of Country and Moisture

```r
cleaned_dataa <- df %>%
  filter(!is.na(`Processing.Method`) & !is.na(Moisture_Level))

cleaned_data <- cleaned_data %>%
  drop_na(`Processing.Method`, Moisture_Level)

contingency_table <- table(cleaned_data$`Processing.Method`, cleaned_data$Moisture_Level)
contingency_table
```

```
##
##                                       High Low
##                                          4   1
##   Anaerobico 1000h                       1   0
##   Double Anaerobic Washed                1   0
##   Double Carbonic Maceration / Natural   1   0
##   Honey,Mossto                           1   0
##   Natural / Dry                         23  23
##   Pulped natural / honey                11  14
##   SEMI-LAVADO                            1   0
##   Semi Washed                            0   1
##   Washed / Wet                          64  59
##   Wet Hulling                            1   0
```

```r
chi_square_test <- chisq.test(contingency_table)
chi_square_test
```

```
##
##  Pearson's Chi-squared test
##
## data:  contingency_table
## X-squared = 8.8988, df = 10, p-value = 0.5417
```

```r
chi_square_test$statistic
```

```
## X-squared
##  8.898785
```

```r
chi_square_test$p.value
```

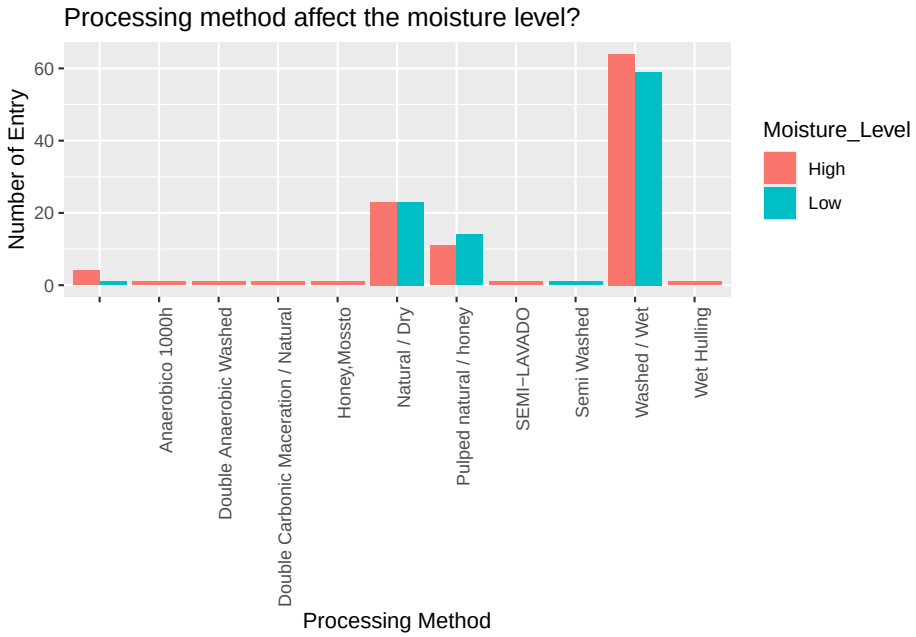
```
## [1] 0.5417355
```

```r
chi_square_test$expected
```

```
##
##                                          High        Low
##                                     2.6213592  2.3786408
##   Anaerobico 1000h                  0.5242718  0.4757282
##   Double Anaerobic Washed           0.5242718  0.4757282
##   Double Carbonic Maceration / Natural 0.5242718  0.4757282
##   Honey,Mossto                      0.5242718  0.4757282
```

```
##   Natural / Dry                      24.1165049 21.8834951
##   Pulped natural / honey             13.1067961 11.8932039
##   SEMI-LAVADO                         0.5242718  0.4757282
##   Semi Washed                         0.5242718  0.4757282
##   Washed / Wet                       64.4854369 58.5145631
##   Wet Hulling                         0.5242718  0.4757282
```

```
ggplot(cleaned_data, aes(x = `Processing.Method`, fill = Moisture_Level)) +
  geom_bar(position = "dodge") +
  labs(title = "Processing method affect the moisture level?",
       x = "Processing Method",
       y = "Number of Entry",
       fill = "Moisture_Level") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



Based on the Chi Square Test

With a p-value of 0.5402, it is greater than the commonly chosen significance level of 0.05.

Since the p-value is greater than 0.05, we fail to reject the null hypothesis. Therefore, we conclude that there is not enough evidence to suggest a significant association between `Processing Method` and `Moisture Level`. In other words, they are considered independent.

We can see that the processing method does not have a significant influence on the moisture level of the coffee samples in the dataset.

Conclusion

Moisture have a significant association between Country of Origin and Moisture Level, indicating that the country influences the moisture content of coffee samples. But there is no significant association between Processing Method and Moisture Level, suggesting that the method of processing does not influence the moisture content of coffee samples. We also can predict that Moisture Percentage does affect the Overall Score of coffee, but the relationship is weak.

Does location changes the overall coffee rating?

In this Analysis, we will be determining if the Average 'Overall' rating of coffee is different between country/location, or if its standardized within each country or region.

```
#cleaning the altitude data calculates the average in any range


# Function to replace ranges with their averages
replace_range_with_average <- function(altitude) {
  if (grepl("-", altitude)) {
    # Split the range into two numbers
    range_values <- strsplit(altitude, "-")[[1]]
    # Convert the string values to numeric
    num_values <- as.numeric(range_values)
    # Calculate the average
    avg_value <- mean(num_values)
    # Return the average as a character string
    return(as.character(avg_value))
  } else {
    # Return the original value if it's not a range
    return(altitude)
  }
}

# Apply the function to the Altitude column
df$Altitude <- sapply(df$Altitude, replace_range_with_average)

# Convert the Altitude column to numeric for further analysis
df$Altitude <- as.numeric(df$Altitude)


# Function to extract the first year
extract_first_year <- function(year_range) {
  first_year <- strsplit(year_range, "/")[[1]][1]
  return(as.numeric(first_year))
}

# Apply the function to the Harvest.Year column
df$Harvest.Year <- sapply(df$Harvest.Year, extract_first_year)
```

We would like to see if there mean overall rating varies between 2 countries.

We want to choose the countries with the most entries.

```
library(dplyr)

# Count the number of entries per country and sort them
countCountries <- df %>%
  count(Country.of.Origin) %>%
  arrange(desc(n))

# Print the sorted count of countries
print(countCountries)
```

Here it seems that Taiwan has the most data entries by far. So on order to get a better result, we shall choose Colombia and Guatemala for our analysis.

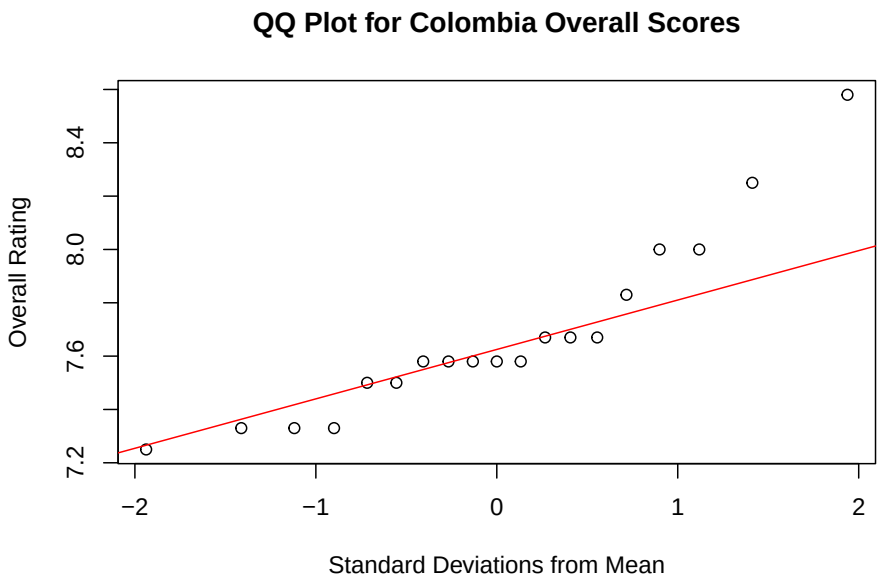
Now we'll create a table for each country.

```
# Subsetting data for Colombia
df_colombia <- df[df$Country.of.Origin == "Colombia", ]

# Subsetting data for Guatemala
df_guatemala <- df[df$Country.of.Origin == "Guatemala", ]
```

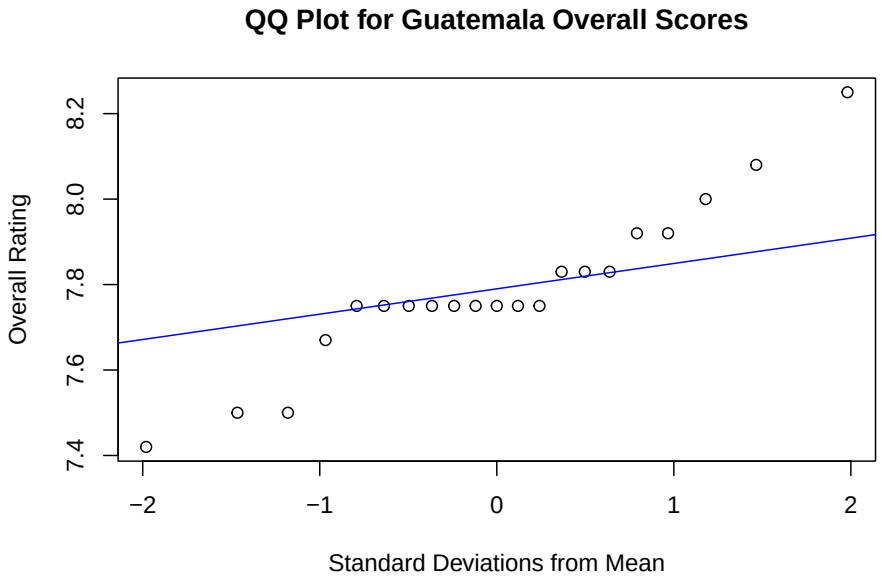
To see if we can use a t test we must see if the "overall" variable has a normal distribution. So we will make a Q-Q Plot for each country

First we will look at colombia

```
qqnorm(df_colombia$Overall, main = "QQ Plot for Colombia Overall Scores",
       xlab = "Standard Deviations from Mean", ylab = "Overall Rating")
qqline(df_colombia$Overall, col = "red")
```



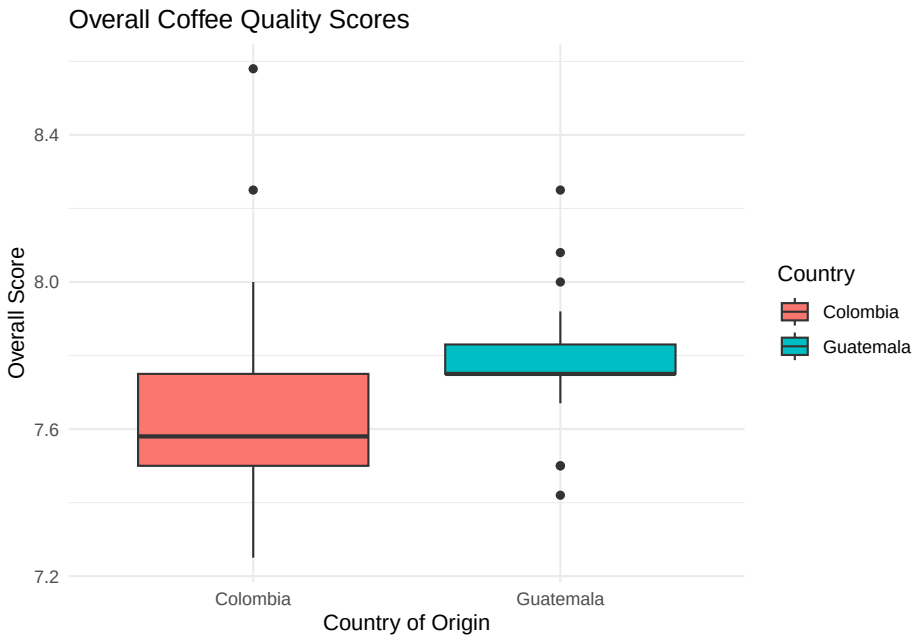

```
# QQ plot for Guatemala with titles and axis labels
qqnorm(df_guatemala$Overall, main = "QQ Plot for Guatemala Overall Scores",
       xlab = "Standard Deviations from Mean", ylab = "Overall Rating")
qqline(df_guatemala$Overall, col = "blue")
```



It seems that the data points follow the direction of the normal line, although not accurately. This could be due to the lack of data points, lets compare to the total overall ratings...

For now lets continue the analysis between Guatemala and Colombia.Below is a box plot showing the variances of the 2 countries 'overall' ratings

```
# Combine data into a single data frame for plotting
df_combined <- rbind(
  data.frame(Country = "Colombia", Overall = df_colombia$Overall),
  data.frame(Country = "Guatemala", Overall = df_guatemala$Overall)
)


ggplot(df_combined, aes(x = Country, y = Overall, fill = Country)) +
  geom_boxplot() +
  labs(title = "Overall Coffee Quality Scores",
       x = "Country of Origin",
       y = "Overall Score") +
  theme_minimal()
```



Here we see that the variance of Guatemala is much smaller than Colombia, and the mean is greater in Colombia.

Now lets do the t-test...

```
# Perform a t-test
t_test_result1 <- t.test(df_guatemala$Overall, df_colombia$Overall,var.equal = F)

# Print t-test results
print(t_test_result1)
```

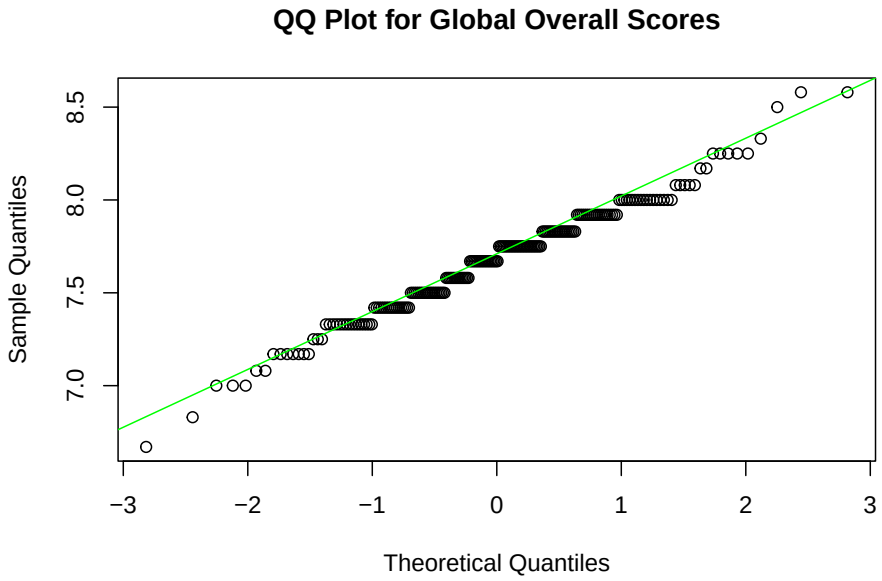
```
##
##  Welch Two Sample t-test
##
## data:  df_guatemala$Overall and df_colombia$Overall
## t = 1.2767, df = 27.718, p-value = 0.2123
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.06748249  0.29049001
## sample estimates:
## mean of x mean of y
##  7.785714  7.674211
```

H0(Null): The means of the two groups are equal. H1(Alternative):The means of the two groups are not equal.

Here the t value is greater than the p value. So we must reject the null hypothesis. This analysis concludes that the means of the 'overall' variable are not equal by countries.

Now lets see if we can make a better analysis with a larger sample. Here we want to see if the total 'overall' ratings follow a normal distribution.

```
qqnorm(df$Overall,main="QQ Plot for Global Overall Scores")
qqline(df$Overall,col="green")
```



Here we can say pretty confidently, that the distribution of the overall ratings globally follow a normal distribution.

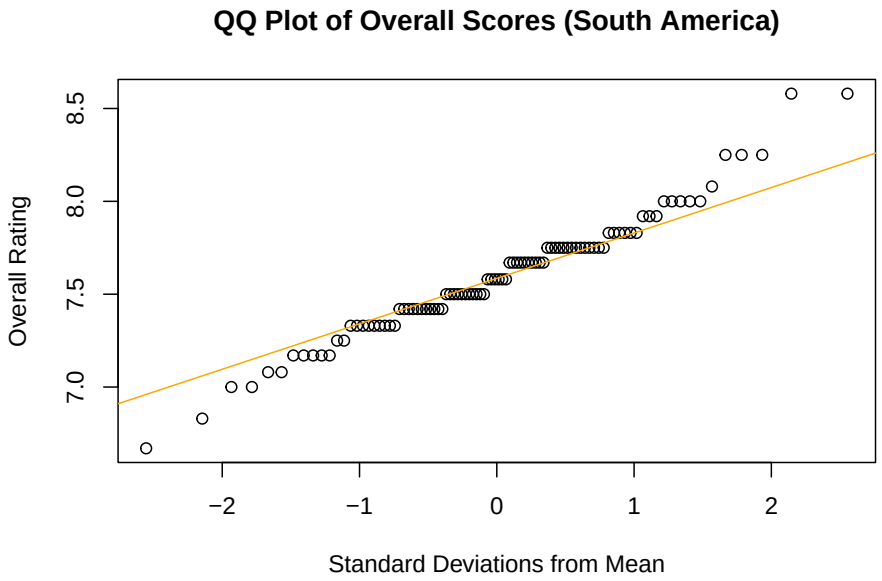
Now that we know that the Overall is normally distributed we can continue our comparative analysis with a larger sample size. Lets compare Asia to South America which will increase the data points from 20 to 95 in each Table…

```r
#find all countries
unique_countries <- unique(df$Country.of.Origin)

#list countries in south america
south_america <- c("Colombia", "Costa Rica", "Guatemala", "Brazil", "Peru", "Panama", "Nicaragua", "Hon

#list countries in asia
asia <- c("Taiwan", "Laos", "Tanzania, United Republic Of", "Ethiopia", "Thailand", "United States (Hawa

# Filter data for South American countries
df_south_america <- df[df$Country.of.Origin %in% south_america, ]

# Filter data for Asian countries
df_asia <- df[df$Country.of.Origin %in% asia, ]

# Count the number of entries in each data table
count_south_america <- nrow(df_south_america)
count_asia <- nrow(df_asia)

#deleting rows from the larger dataset
# Calculate the number of rows to remove from df_asia
rows_to_remove <- 17

# Remove rows from df_asia
df_asia <- df_asia[-(1:rows_to_remove), ]
```

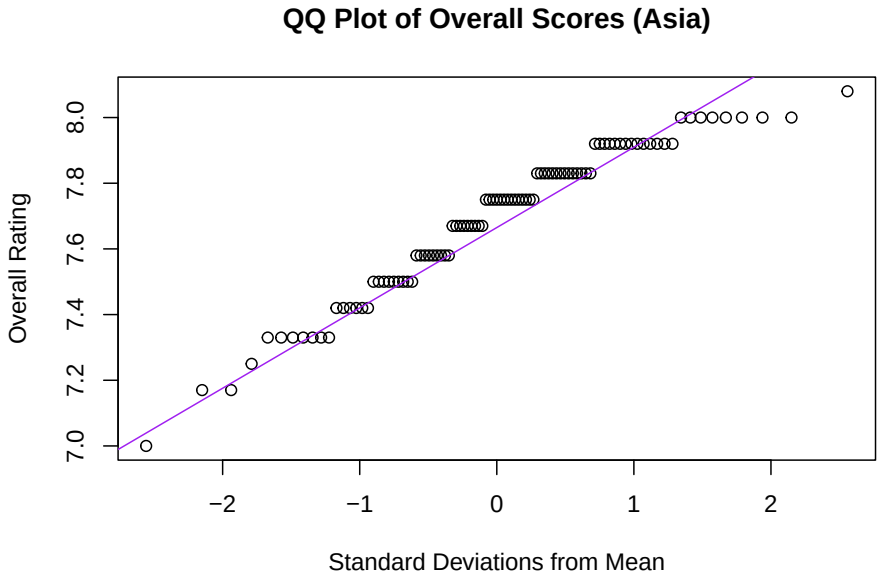
By combining all of the entries from countries within South America and Asia and subtracting a few entries from the larger dataset, we now have 2 data sets with 95 entries each. Compared to the 20 from before.

Once again, lets now test the distribution of the overall rating in South America and Asia...

```
# South America QQ plot
qqnorm(df_south_america$Overall, main = "QQ Plot of Overall Scores (South America)",
       xlab = "Standard Deviations from Mean", ylab = "Overall Rating")
qqline(df_south_america$Overall, col = "orange")
```




```
# Asia QQ plot
qqnorm(df_asia$Overall, main = "QQ Plot of Overall Scores (Asia)",
       xlab = "Standard Deviations from Mean", ylab = "Overall Rating")
qqline(df_asia$Overall, col = "purple")
```

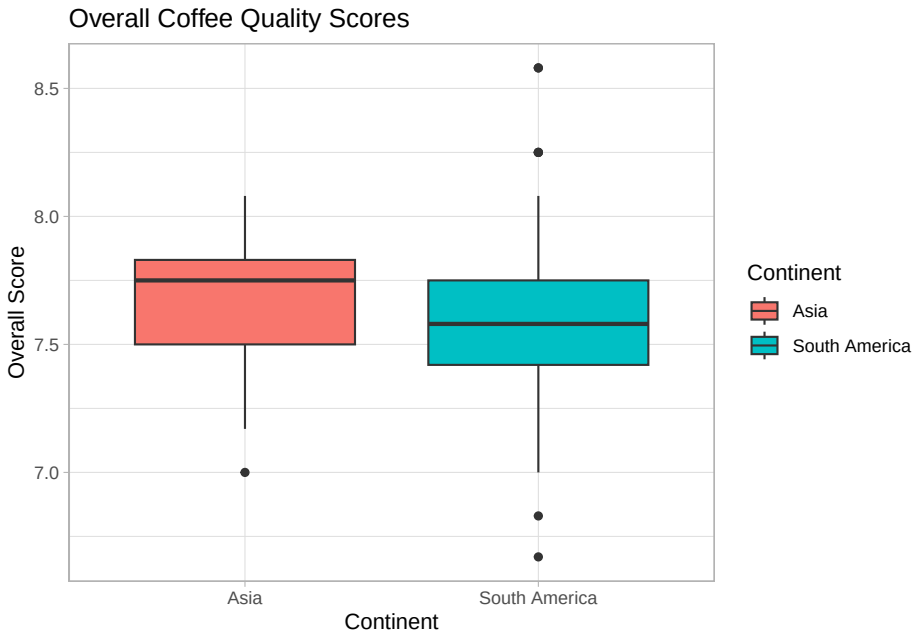


We can say with confi-

dence that the "overall" column in both data sets follows a normal distribution. So we can continue with our analysis.

Next we must test the variances of each table to see if they match.

```
# Combine data into a single data frame for plotting
df_combined <- rbind(
  data.frame(Continent = "South America", Overall = df_south_america$Overall),
  data.frame(Continent = "Asia", Overall = df_asia$Overall)
)

# Create the boxplot
ggplot(df_combined, aes(x = Continent, y = Overall, fill = Continent)) +
  geom_boxplot() +
  labs(title = "Overall Coffee Quality Scores",
       x = "Continent",
       y = "Overall Score") +
  theme_light()
```



Here we can see that the variance isn't equal between the continents. This graph tells us that the mean overall rating is higher in Asia. We also see that South America has the coffee with the highest and lowest overall rating.

Now that we know the variances aren't equal we can preform a comparative t test.

```
t_test_result <- t.test(df_south_america$Overall, df_asia$Overall,var.equal = F)

# Print t-test results
print(t_test_result)
```

```
##
##  Welch Two Sample t-test
##
## data:  df_south_america$Overall and df_asia$Overall
## t = -2.3185, df = 165.62, p-value = 0.02164
```

```
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.17832634 -0.01429404
## sample estimates:
## mean of x mean of y
##  7.590532  7.686842
```

H0(Null): The means of the two groups are equal.

H1(Alternative):The means of the two groups are not equal.

Given the results of the T-test, we reject H0. This implies that on average the Average overall rating of coffee from Asia is higher than that from of South America.

Summary:

In conclusion, we were able to determine that the average overall rating of coffee between Asia and South America, and seperate countries do not share the value. Additionally, our analysis suggests that on average, coffee from Asia is Overall better than coffee from South America.

Introduction: For this part of the analysis we'll analyse how the Altitude at which the coffee is grown can affect or is correlated to the sensory scores of its variety.

```
#In this cell we install some packages needed for the following plots
#we had done it the normal way but there was a problem with the knitting and we had to debug using chat
# Set CRAN mirror
options(repos = c(CRAN = "https://cran.r-project.org"))

# List of packages to be installed
packages <- c("ggplot2", "ggridges", "corrr", "FactoMineR", "ggcorrplot", "factoextra")

# Function to check if packages are installed and install them if not
install_if_missing <- function(p) {
  if (!requireNamespace(p, quietly = TRUE)) {
    install.packages(p)
  }
}

# Install missing packages
lapply(packages, install_if_missing)

# Load the packages
lapply(packages, library, character.only = TRUE)
```

The first part of the process is to clean the data, for this, we select the columns of the variables we will be using, since we notice that some values of the altitude are set as ranges and we need them as numerical values we do a function call avg_altitude that if the value of altitude is separated with a "-" it will separate the string by the dash, using the strsplit function, that returns an array , then it will turn this values in the array into numbers and average them out returning in that way the average altitude.

```
df<-read.csv("H:/Downloads/coffee_quality.csv")
df<- df[,c("Altitude","Variety","Aroma","Flavor","Aftertaste","Acidity","Body","Balance","Overall")]
avg_altitude <- function(altitude) {
  if (grepl("-", altitude)) {
    range_values <- strsplit(altitude, "-")[[1]]
    num_values <- as.numeric(range_values)
```

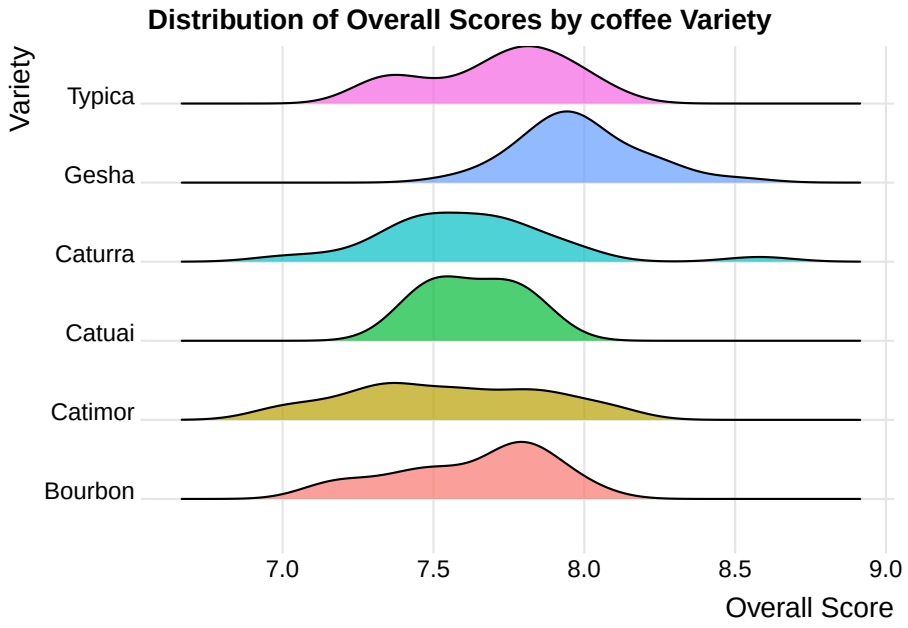
```
    avg_value <- mean(num_values)
    return(as.character(avg_value))
  } else {
    return(altitude)
  }
}

# Apply the function to the Altitude column
df$Altitude <- sapply(df$Altitude, avg_altitude)

# Convert the Altitude column to numeric for further analysis
df$Altitude <- as.numeric(df$Altitude)

# View the modified data frame
summary(df)
head(df)
```

Since some varieties of coffee don't present don't have a significant sample, we subset the dataset to drop rows which contain varieties with less than entries we also eliminate rows containing missing values.

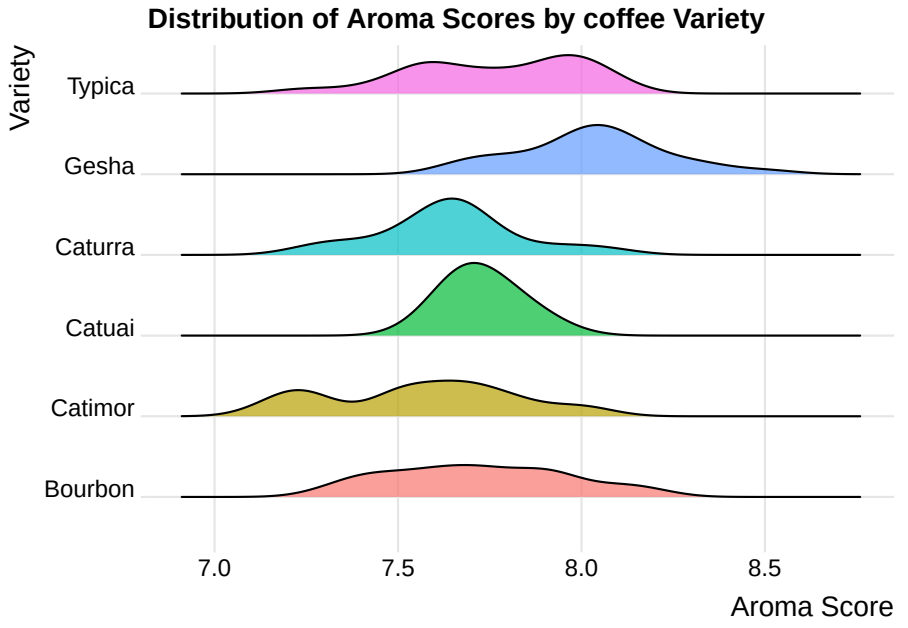
```
vc<-table(df$Variety)
vc<-names(vc[vc>10])
df<-subset(df, (df$Variety %in% vc)& df$Variety!="unknown")
unique(df$Variety)
df<-na.omit(df)
View(df)
```

Now using a ridgeline Plot we can visuallize the distribution of the variables we are going to study with respect to the variety of coffee

```
ggplot(df, aes(x = Overall, y = Variety, fill = Variety)) +
  geom_density_ridges(scale = 0.9, alpha = 0.7) +
  labs(title = "Distribution of Overall Scores by coffee Variety",
       x = "Overall Score",
       y = "Variety") +
  theme_ridges() +
  theme(legend.position = "none")
```

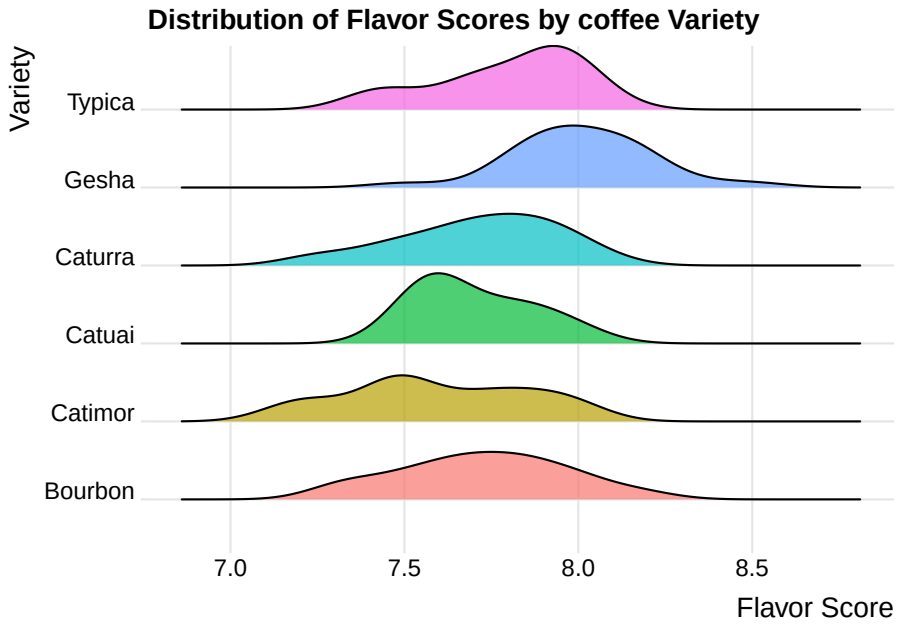



```
ggplot(df, aes(x = Aroma, y = Variety, fill = Variety)) +
  geom_density_ridges(scale = 0.9, alpha = 0.7) +
  labs(title = "Distribution of Aroma Scores by coffee Variety",
       x = "Aroma Score",
       y = "Variety") +
  theme_ridges() +
  theme(legend.position = "none")
```

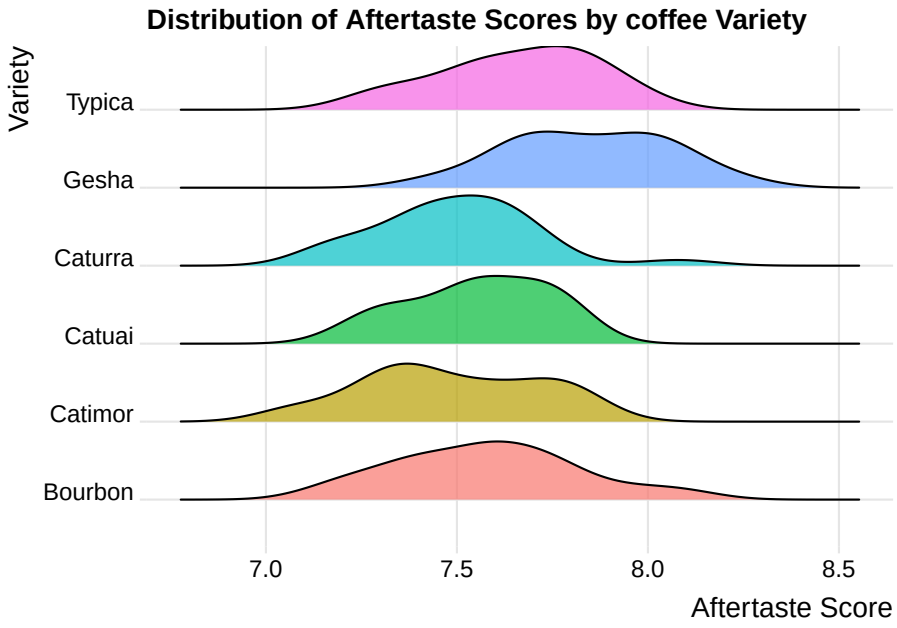



```
ggplot(df, aes(x = Flavor, y = Variety, fill = Variety)) +
  geom_density_ridges(scale = 0.9, alpha = 0.7) +
```

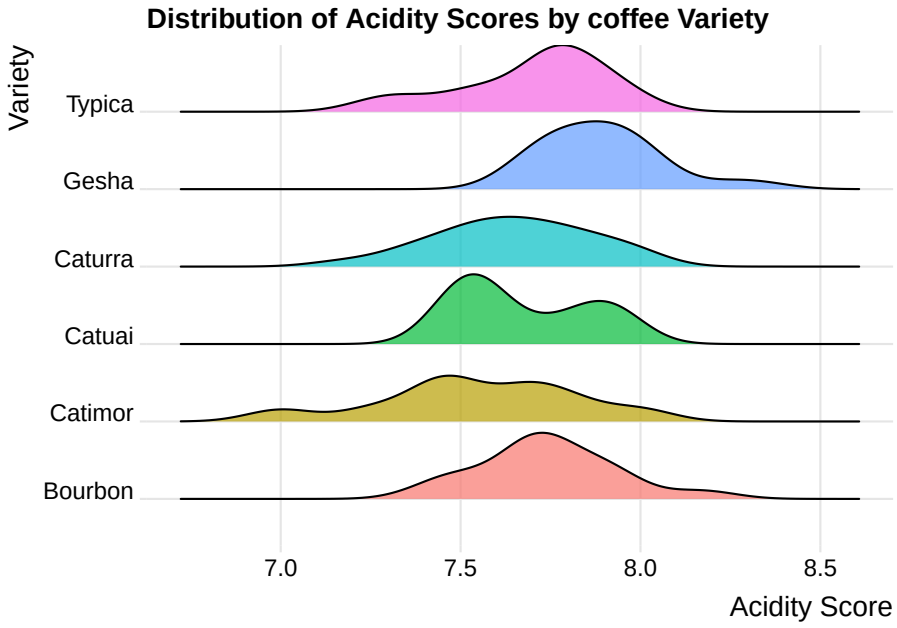
```
  labs(title = "Distribution of Flavor Scores by coffee Variety",
       x = "Flavor Score",
       y = "Variety") +
  theme_ridges() +
  theme(legend.position = "none")
```




```
ggplot(df, aes(x = Aftertaste, y = Variety, fill = Variety)) +
  geom_density_ridges(scale = 0.9, alpha = 0.7) +
  labs(title = "Distribution of Aftertaste Scores by coffee Variety",
       x = "Aftertaste Score",
       y = "Variety") +
  theme_ridges() +
  theme(legend.position = "none")
```

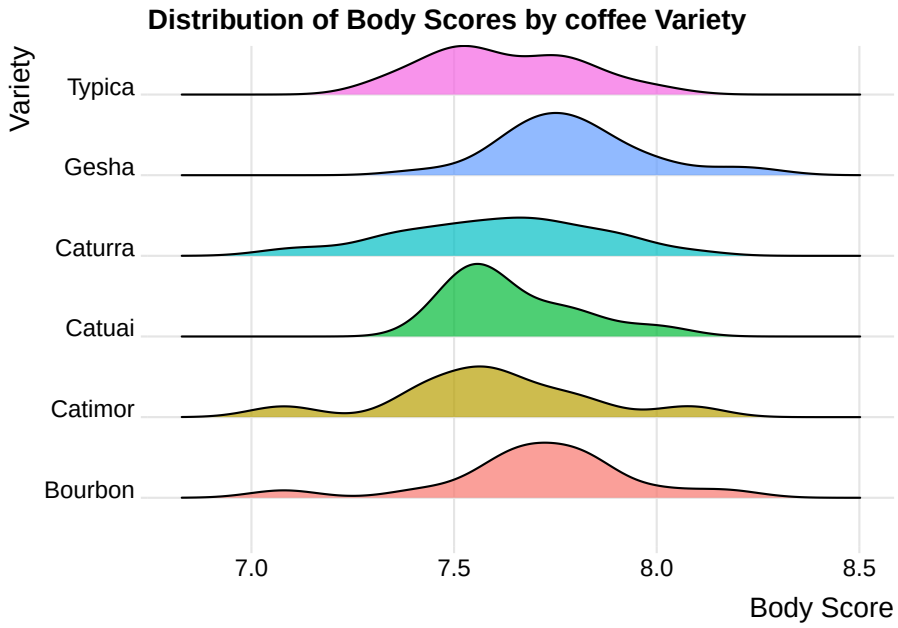



```
ggplot(df, aes(x = Acidity, y = Variety, fill = Variety)) +
  geom_density_ridges(scale = 0.9, alpha = 0.7) +
  labs(title = "Distribution of Acidity Scores by coffee Variety",
       x = "Acidity Score",
       y = "Variety") +
  theme_ridges() +
  theme(legend.position = "none")
```

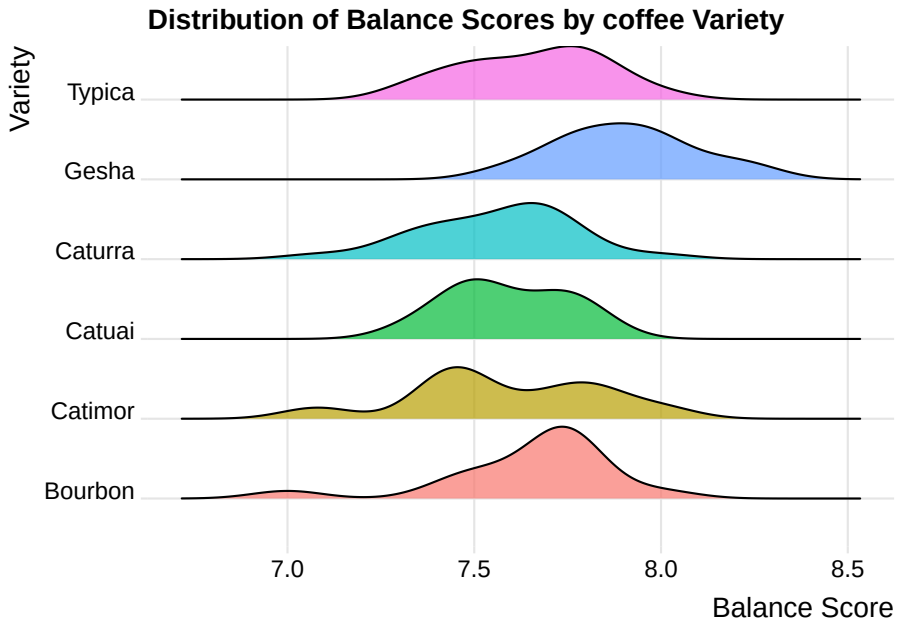



```
ggplot(df, aes(x = Body, y = Variety, fill = Variety)) +
  geom_density_ridges(scale = 0.9, alpha = 0.7) +
```

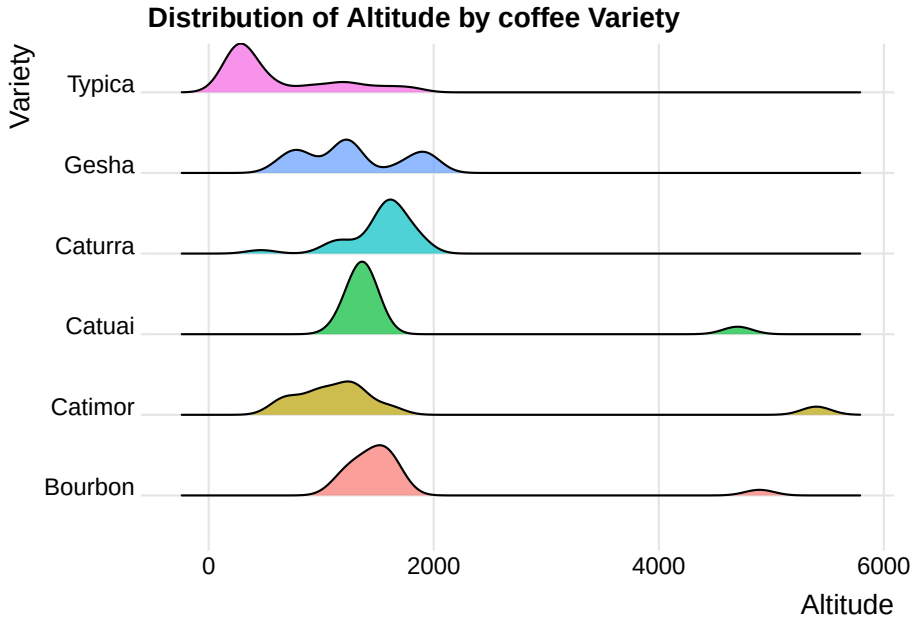
```
  labs(title = "Distribution of Body Scores by coffee Variety",
       x = "Body Score",
       y = "Variety") +
  theme_ridges() +
  theme(legend.position = "none")
```




```
ggplot(df, aes(x = Balance, y = Variety, fill = Variety)) +
  geom_density_ridges(scale = 0.9, alpha = 0.7) +
  labs(title = "Distribution of Balance Scores by coffee Variety",
       x = "Balance Score",
       y = "Variety") +
  theme_ridges() +
  theme(legend.position = "none")
```




```
ggplot(df, aes(x = Altitude, y = Variety, fill = Variety)) +
  geom_density_ridges(scale = 0.9, alpha = 0.7) +
  labs(title = "Distribution of Altitude by coffee Variety",
       x = "Altitude",
       y = "Variety") +
  theme_ridges() +
  theme(legend.position = "none")
```

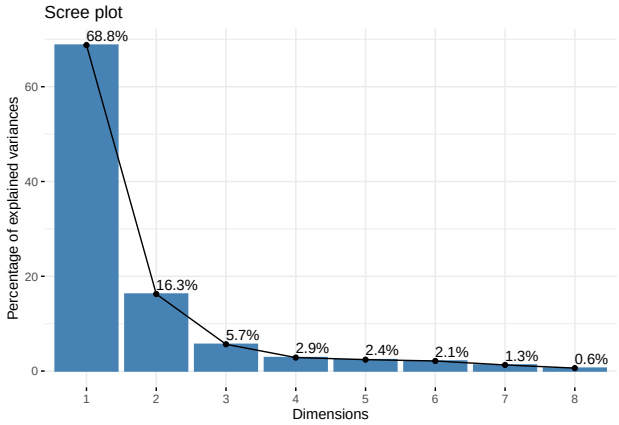


Now that the data is clean, we'll start the study, first we subset the data by its coffee Variety, then we remove the Variety column to ensure that the dataset will only contain numerical data, we use the scale function to

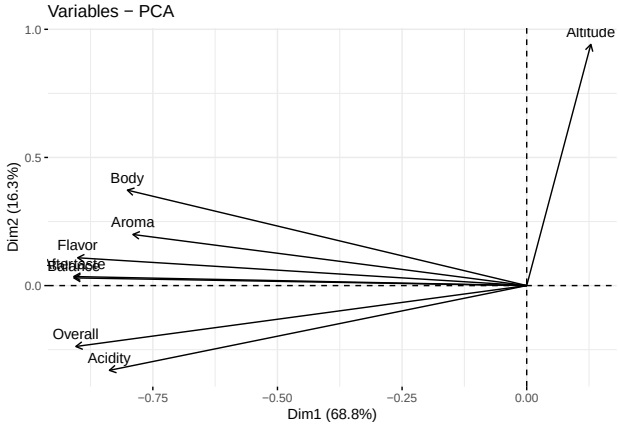
normalize the data in order to avoid introducing bias via the scales of the variables, and we finish by using PCA in order to visualize the correlation of the variables.

```
#subset by Variety and eliminate non-numerical variable
Typica<-df[df$Variety=="Typica",]
Typica<-Typica[,-c(2)]
#normalizing Data
Typica<-scale(Typica)
head(Typica)
#principal components
Typica_data.pca <- princomp(Typica)
summary(Typica_data.pca)
# Value of variable in each component
Typica_data.pca$loadings[, 1:2]
```

```
#scree plot
fviz_eig(Typica_data.pca, addlabels = TRUE)
```



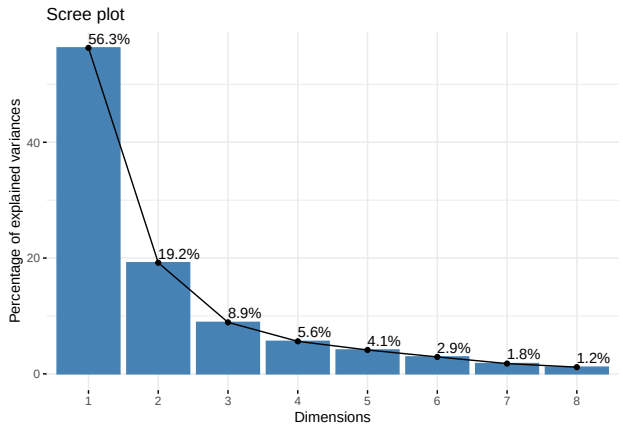

```
#PCA graph
fviz_pca_var(Typica_data.pca, col.var = "black")
```



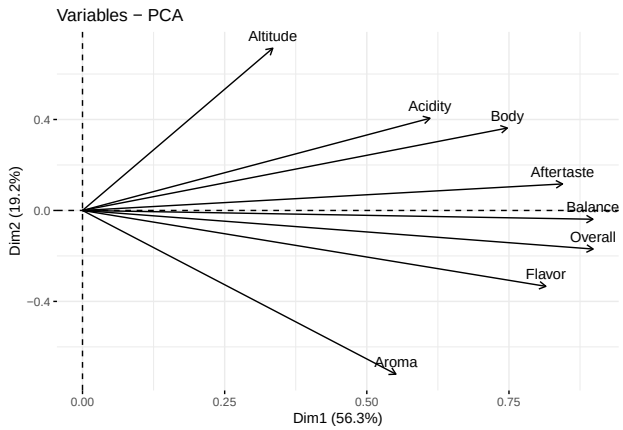
From the PCA graph for the Typica coffee, we can see that the altitude has a very week negative correlation with the sensory scores of the coffee, since is very far from the other variables and going in an oposite sense.

```
Gesha<-df[df$Variety=="Gesha",]
Gesha<-Gesha[,-c(2)]
Gesha<-scale(Gesha)
head(Gesha,n=5)
Gesha_data.pca <- princomp(Gesha)
summary(Gesha_data.pca)
Gesha_data.pca$loadings[, 1:2]
```

```
fviz_eig(Gesha_data.pca, addlabels = TRUE)
```



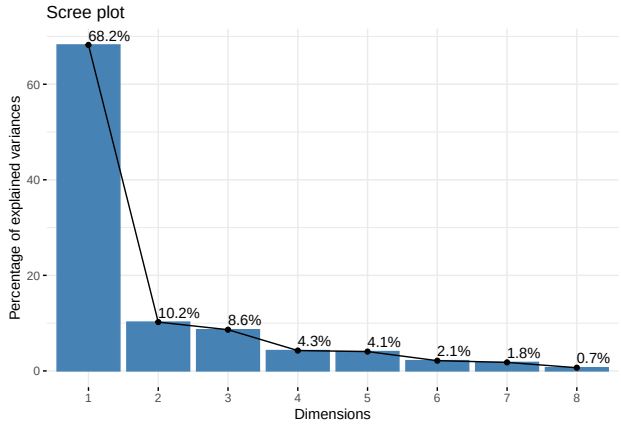

```
fviz_pca_var(Gesha_data.pca, col.var = "black")
```



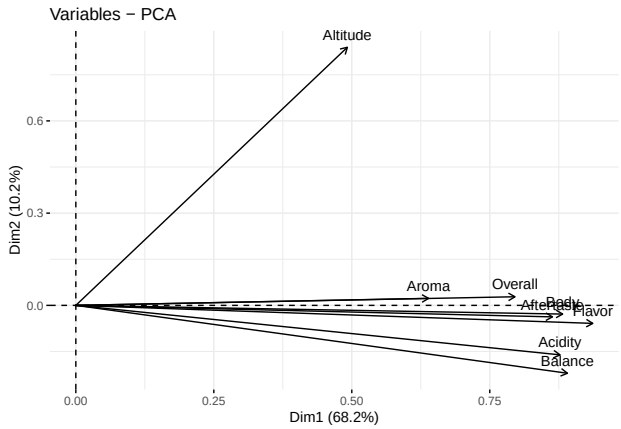
From the PCA graph of the Gesha coffee we see that the altitude has weak positive correlation with the sensory scores of the coffee, since its closer to the other variables and share the same sense.

```
Caturra<-df[df$Variety=="Caturra",]
Caturra<-Caturra[,-c(2)]
Caturra<-scale(Caturra)
head(Caturra)
Caturra_data.pca <- princomp(Caturra)
summary(Caturra_data.pca)
Caturra_data.pca$loadings[, 1:2]
```

```
fviz_eig(Caturra_data.pca, addlabels = TRUE)
```



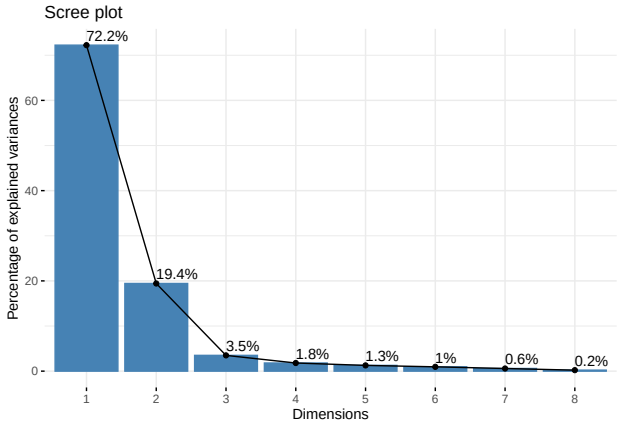

```
fviz_pca_var(Caturra_data.pca, col.var = "black")
```



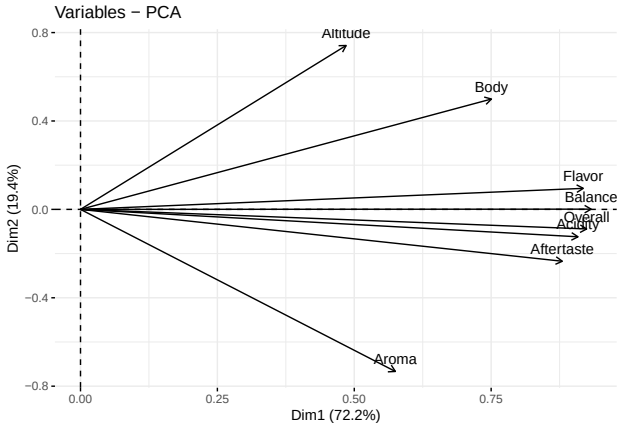
From the PCA graph of the Caturra coffee we can observe that the altitude has a very week positive correlation with the other variables since shares the same sens but is far from the sensory score cluster.

```
Catuai<-df[df$Variety=="Catuai",]
Catuai<-Catuai[,-c(2)]
Catuai<-scale(Catuai)
head(Catuai)
Catuai_data.pca <- princomp(Catuai)
summary(Catuai_data.pca)
Catuai_data.pca$loadings[, 1:2]
```

```
fviz_eig(Catuai_data.pca, addlabels = TRUE)
```



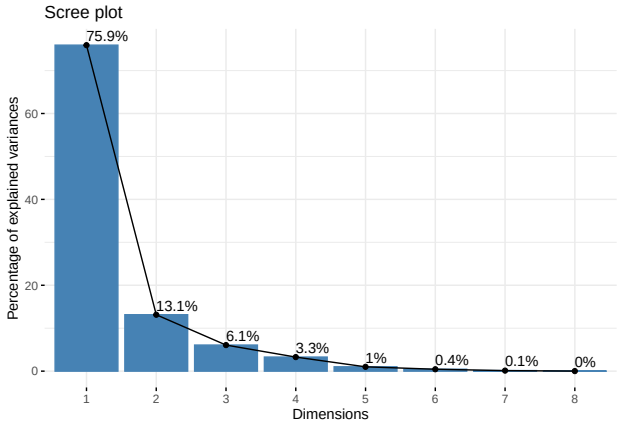

```
fviz_pca_var(Catuai_data.pca, col.var = "black")
```



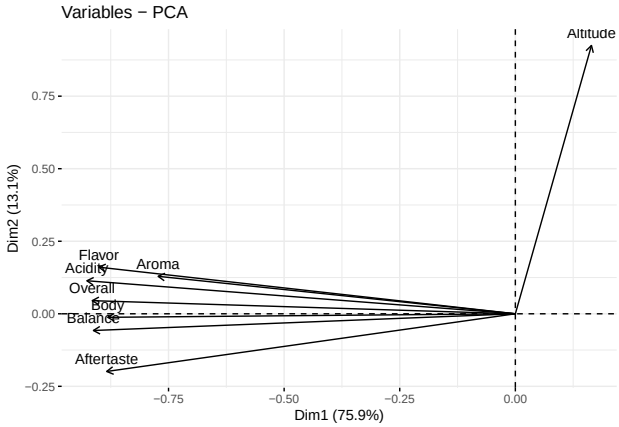
Similarly to the Gesha form the PCA graph the Catuai Variety has a stronger positive correlation with the other sensory scores although we can still see that its considerably farther away from the other variables.

```
Catimor<-df[df$Variety=="Catimor",]
Catimor<-Catimor[,-c(2)]
Catimor<-scale(Catimor)
head(Catimor)
Catimor_data.pca <- princomp(Catimor)
summary(Catimor_data.pca)
Catimor_data.pca$loadings[, 1:2]
```

```
fviz_eig(Catimor_data.pca, addlabels = TRUE)
```



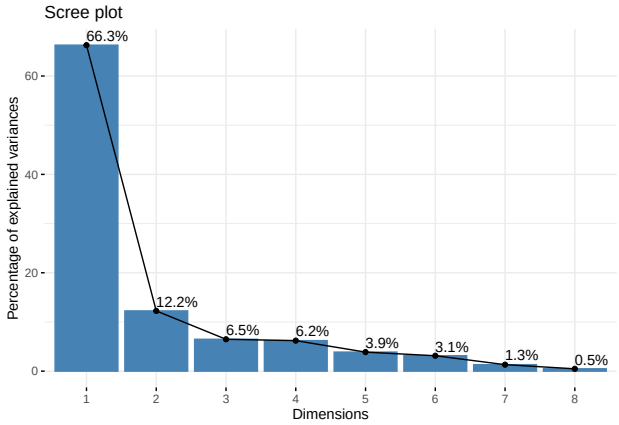

```
fviz_pca_var(Catimor_data.pca, col.var = "black")
```



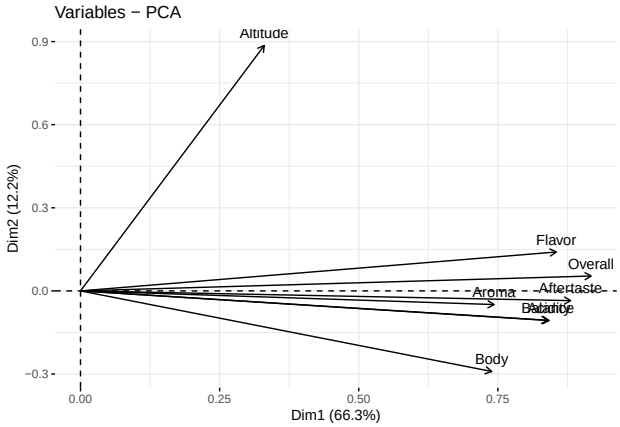
From the PCA graph for the Catimor coffee, we can see also a very weak negative correlation between the altitude and the other sensory scores.

```
Bourbon<-df[df$Variety=="Bourbon",]
Bourbon<-Bourbon[,-c(2)]
Bourbon<-scale(Bourbon)
head(Bourbon)
Bourbon_data.pca <- princomp(Bourbon)
summary(Bourbon_data.pca)
Bourbon_data.pca$loadings[, 1:2]
```

```
fviz_eig(Bourbon_data.pca, addlabels = TRUE)
```




```
fviz_pca_var(Bourbon_data.pca, col.var = "black")
```



From the PCA graph of the bourbon variety we can observe that there is a very weak positive correlation with the sensory scores.

Conclusion: From the previous graph we can observe that the sensory scores of the coffees are more or less snsible of correlated to altitude depending on the variety, this said, the correlation between the altitude and the sensory scored of the coffee tend to be pretty weak. It is also worth mentioning than on average PC1 and PC2 account for 83.01 of the variation which indicate that this PCA graphs are very good to visualize the correlation of the variables.