Group7 Project - Coffee Rating

Ece Gulkirpik, Johana Coronel, Mengyi Dong

10/8/2020

```
## Install and load packages
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.4     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.0

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(tidytuesdayR)
library(ggplot2)
library(fmsb)
# Read data file
## coffee_rating
coffee_ratings <- readr::read_csv('https://raw.githubusercontent.com/rfordatascience/tidytuesday/master/
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   .default = col_character(),
##   total_cup_points = col_double(),
##   number_of_bags = col_double(),
##   aroma = col_double(),
##   flavor = col_double(),
##   aftertaste = col_double(),
##   acidity = col_double(),
##   body = col_double(),
##   balance = col_double(),
##   uniformity = col_double(),
##   clean_cup = col_double(),
##   sweetness = col_double(),
##   cupper_points = col_double(),
##   moisture = col_double(),
##   category_one_defects = col_double(),
##   quakers = col_double(),
```

```
##   category_two_defects = col_double(),
##   altitude_low_meters = col_double(),
##   altitude_high_meters = col_double(),
##   altitude_mean_meters = col_double()
## )
## i Use `spec()` for the full column specifications.
```

1. Introduction

Coffee traces its origin to a genus of plants known as Coffea. The two most commercially important species grown are varieties of *Coffeaarabica* (Arabicas) and *Coffeacanephora* (Robustas) (ISIC, 2014).

Coffee Arabica is descended from the original coffee trees discovered in Ethiopia. These trees produce a fine, mild, aromatic coffee and represent approximately 70% of the world's coffee production. The beans are flatter and more elongated than Robusta and lower in caffeine. Most of the world's Robusta is grown in Central and Western Africa, parts of Southeast Asia, including Indonesia and Vietnam, and in Brazil. Production of Robusta is increasing, though it accounts for only about 30% of the world market. Robusta is primarily used in blends and for instant coffees. The Robusta bean itself tends to be slightly rounder and smaller than an Arabica bean (NCA, 2020).

"Coffee_ratings" dataset puts together a great amount of data on the sensorial attributes of these two coffee species which were grown and processed in many different ways by various countries. This dataset not only provides valuable information about the sensory characteristics of coffee species but also gives considerable amount of background information belong to each species including farm and company names, producers, harvest years, and certifications, etc.

"Coffee_ratings" dataset includes 43 variables in which 24 of them are character and the rest are numerical. The details about the variables can be obtained from the following R chunk:

```
## Getting to know the variables of "Coffee_ratings" dataset

head(coffee_ratings)
```

```
## # A tibble: 6 x 43
##   total_cup_points species owner country_of_orig~ farm_name lot_number mill
##              <dbl> <chr>   <chr> <chr>            <chr>     <chr>      <chr>
## 1             90.6 Arabica meta~ Ethiopia         "metad p~ <NA>       meta~
## 2             89.9 Arabica meta~ Ethiopia         "metad p~ <NA>       meta~
## 3             89.8 Arabica grou~ Guatemala        "san mar~ <NA>       <NA>
## 4             89   Arabica yidn~ Ethiopia         "yidneka~ <NA>       wole~
## 5             88.8 Arabica meta~ Ethiopia         "metad p~ <NA>       meta~
## 6             88.8 Arabica ji-a~ Brazil            <NA>     <NA>       <NA>
## # ... with 36 more variables: ico_number <chr>, company <chr>, altitude <chr>,
## #   region <chr>, producer <chr>, number_of_bags <dbl>, bag_weight <chr>,
## #   in_country_partner <chr>, harvest_year <chr>, grading_date <chr>,
## #   owner_1 <chr>, variety <chr>, processing_method <chr>, aroma <dbl>,
## #   flavor <dbl>, aftertaste <dbl>, acidity <dbl>, body <dbl>, balance <dbl>,
## #   uniformity <dbl>, clean_cup <dbl>, sweetness <dbl>, cupper_points <dbl>,
## #   moisture <dbl>, category_one_defects <dbl>, quakers <dbl>, color <chr>,
## #   category_two_defects <dbl>, expiration <chr>, certification_body <chr>,
## #   certification_address <chr>, certification_contact <chr>,
## #   unit_of_measurement <chr>, altitude_low_meters <dbl>,
## #   altitude_high_meters <dbl>, altitude_mean_meters <dbl>
```

```r
str(coffee_ratings)
```

```
## tibble [1,339 x 43] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
##  $ total_cup_points     : num [1:1339] 90.6 89.9 89.8 89 88.8 ...
##  $ species              : chr [1:1339] "Arabica" "Arabica" "Arabica" "Arabica" ...
##  $ owner                : chr [1:1339] "metad plc" "metad plc" "grounds for health admin" "yidnekach
##  $ country_of_origin    : chr [1:1339] "Ethiopia" "Ethiopia" "Guatemala" "Ethiopia" ...
##  $ farm_name            : chr [1:1339] "metad plc" "metad plc" "san marcos barrancas \"san cristobal
##  $ lot_number           : chr [1:1339] NA NA NA NA ...
##  $ mill                 : chr [1:1339] "metad plc" "metad plc" NA "wolensu" ...
##  $ ico_number           : chr [1:1339] "2014/2015" "2014/2015" NA NA ...
##  $ company              : chr [1:1339] "metad agricultural developmet plc" "metad agricultural develo
##  $ altitude             : chr [1:1339] "1950-2200" "1950-2200" "1600 - 1800 m" "1800-2200" ...
##  $ region               : chr [1:1339] "guji-hambela" "guji-hambela" NA "oromia" ...
##  $ producer             : chr [1:1339] "METAD PLC" "METAD PLC" NA "Yidnekachew Dabessa Coffee Planta
##  $ number_of_bags       : num [1:1339] 300 300 5 320 300 100 100 300 300 50 ...
##  $ bag_weight           : chr [1:1339] "60 kg" "60 kg" "1" "60 kg" ...
##  $ in_country_partner   : chr [1:1339] "METAD Agricultural Development plc" "METAD Agricultural Devel
##  $ harvest_year         : chr [1:1339] "2014" "2014" NA "2014" ...
##  $ grading_date         : chr [1:1339] "April 4th, 2015" "April 4th, 2015" "May 31st, 2010" "March 2
##  $ owner_1              : chr [1:1339] "metad plc" "metad plc" "Grounds for Health Admin" "Yidnekach
##  $ variety              : chr [1:1339] NA "Other" "Bourbon" NA ...
##  $ processing_method    : chr [1:1339] "Washed / Wet" "Washed / Wet" NA "Natural / Dry" ...
##  $ aroma                : num [1:1339] 8.67 8.75 8.42 8.17 8.25 8.58 8.42 8.25 8.67 8.08 ...
##  $ flavor               : num [1:1339] 8.83 8.67 8.5 8.58 8.5 8.42 8.5 8.33 8.67 8.58 ...
##  $ aftertaste           : num [1:1339] 8.67 8.5 8.42 8.42 8.25 8.42 8.33 8.5 8.58 8.5 ...
##  $ acidity              : num [1:1339] 8.75 8.58 8.42 8.42 8.5 8.5 8.5 8.42 8.42 8.5 ...
##  $ body                 : num [1:1339] 8.5 8.42 8.33 8.5 8.42 8.25 8.25 8.33 8.33 7.67 ...
##  $ balance              : num [1:1339] 8.42 8.42 8.42 8.25 8.33 8.33 8.25 8.5 8.42 8.42 ...
##  $ uniformity           : num [1:1339] 10 10 10 10 10 10 10 10 9.33 10 ...
##  $ clean_cup            : num [1:1339] 10 10 10 10 10 10 10 10 10 10 ...
##  $ sweetness            : num [1:1339] 10 10 10 10 10 10 10 9.33 9.33 10 ...
##  $ cupper_points        : num [1:1339] 8.75 8.58 9.25 8.67 8.58 8.33 8.5 9 8.67 8.5 ...
##  $ moisture             : num [1:1339] 0.12 0.12 0 0.11 0.12 0.11 0.11 0.03 0.03 0.1 ...
##  $ category_one_defects : num [1:1339] 0 0 0 0 0 0 0 0 0 0 ...
##  $ quakers              : num [1:1339] 0 0 0 0 0 0 0 0 0 0 ...
##  $ color                : chr [1:1339] "Green" "Green" NA "Green" ...
##  $ category_two_defects : num [1:1339] 0 1 0 2 2 1 0 0 0 4 ...
##  $ expiration           : chr [1:1339] "April 3rd, 2016" "April 3rd, 2016" "May 31st, 2011" "March 2
##  $ certification_body   : chr [1:1339] "METAD Agricultural Development plc" "METAD Agricultural Devel
##  $ certification_address: chr [1:1339] "309fcf77415a3661ae83e027f7e5f05dad786e44" "309fcf77415a3661a
##  $ certification_contact: chr [1:1339] "19fef5a731de2db57d16da10287413f5f99bc2dd" "19fef5a731de2db57
##  $ unit_of_measurement  : chr [1:1339] "m" "m" "m" "m" ...
##  $ altitude_low_meters  : num [1:1339] 1950 1950 1600 1800 1950 ...
##  $ altitude_high_meters : num [1:1339] 2200 2200 1800 2200 2200 NA NA 1700 1700 1850 ...
##  $ altitude_mean_meters : num [1:1339] 2075 2075 1700 2000 2075 ...
##  - attr(*, "spec")=
##   .. cols(
##   ..   total_cup_points = col_double(),
##   ..   species = col_character(),
##   ..   owner = col_character(),
##   ..   country_of_origin = col_character(),
##   ..   farm_name = col_character(),
```

```
##   ..   lot_number = col_character(),
##   ..   mill = col_character(),
##   ..   ico_number = col_character(),
##   ..   company = col_character(),
##   ..   altitude = col_character(),
##   ..   region = col_character(),
##   ..   producer = col_character(),
##   ..   number_of_bags = col_double(),
##   ..   bag_weight = col_character(),
##   ..   in_country_partner = col_character(),
##   ..   harvest_year = col_character(),
##   ..   grading_date = col_character(),
##   ..   owner_1 = col_character(),
##   ..   variety = col_character(),
##   ..   processing_method = col_character(),
##   ..   aroma = col_double(),
##   ..   flavor = col_double(),
##   ..   aftertaste = col_double(),
##   ..   acidity = col_double(),
##   ..   body = col_double(),
##   ..   balance = col_double(),
##   ..   uniformity = col_double(),
##   ..   clean_cup = col_double(),
##   ..   sweetness = col_double(),
##   ..   cupper_points = col_double(),
##   ..   moisture = col_double(),
##   ..   category_one_defects = col_double(),
##   ..   quakers = col_double(),
##   ..   color = col_character(),
##   ..   category_two_defects = col_double(),
##   ..   expiration = col_character(),
##   ..   certification_body = col_character(),
##   ..   certification_address = col_character(),
##   ..   certification_contact = col_character(),
##   ..   unit_of_measurement = col_character(),
##   ..   altitude_low_meters = col_double(),
##   ..   altitude_high_meters = col_double(),
##   ..   altitude_mean_meters = col_double()
##   .. )
```

The first step of data analysis is to identify basic statistical properties of a given dataset, such as ranges, means, medians, quantiles, max and min values of variables, etc. For this purpose, as it is shown in the following R chunk, *"summary()"* function is used.

```
# Summary statistics of the "Coffee Rating" dataset

summary(coffee_ratings)
```

```
##  total_cup_points   species             owner           country_of_origin
##  Min.   : 0.00    Length:1339        Length:1339        Length:1339
##  1st Qu.:81.08    Class :character   Class :character   Class :character
##  Median :82.50    Mode  :character   Mode  :character   Mode  :character
##  Mean   :82.09
```

```
##  3rd Qu.:83.67
##  Max.   :90.58
##
##   farm_name          lot_number            mill            ico_number
##  Length:1339        Length:1339        Length:1339        Length:1339
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##    company            altitude           region            producer
##  Length:1339        Length:1339        Length:1339        Length:1339
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##  number_of_bags    bag_weight        in_country_partner harvest_year
##  Min.   :   0.0   Length:1339        Length:1339        Length:1339
##  1st Qu.:  14.0   Class :character   Class :character   Class :character
##  Median : 175.0   Mode  :character   Mode  :character   Mode  :character
##  Mean   : 154.2
##  3rd Qu.: 275.0
##  Max.   :1062.0
##
##  grading_date         owner_1            variety          processing_method
##  Length:1339        Length:1339        Length:1339        Length:1339
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##      aroma            flavor       aftertaste       acidity           body
##  Min.   :0.000   Min.   :0.00   Min.   :0.000   Min.   :0.000   Min.   :0.000
##  1st Qu.:7.420   1st Qu.:7.33   1st Qu.:7.250   1st Qu.:7.330   1st Qu.:7.330
##  Median :7.580   Median :7.58   Median :7.420   Median :7.580   Median :7.500
##  Mean   :7.567   Mean   :7.52   Mean   :7.401   Mean   :7.536   Mean   :7.517
##  3rd Qu.:7.750   3rd Qu.:7.75   3rd Qu.:7.580   3rd Qu.:7.750   3rd Qu.:7.670
##  Max.   :8.750   Max.   :8.83   Max.   :8.670   Max.   :8.750   Max.   :8.580
##
##     balance        uniformity       clean_cup        sweetness
##  Min.   :0.000   Min.   : 0.000   Min.   : 0.000   Min.   : 0.000
##  1st Qu.:7.330   1st Qu.:10.000   1st Qu.:10.000   1st Qu.:10.000
##  Median :7.500   Median :10.000   Median :10.000   Median :10.000
##  Mean   :7.518   Mean   : 9.835   Mean   : 9.835   Mean   : 9.857
##  3rd Qu.:7.750   3rd Qu.:10.000   3rd Qu.:10.000   3rd Qu.:10.000
##  Max.   :8.750   Max.   :10.000   Max.   :10.000   Max.   :10.000
##
##  cupper_points       moisture       category_one_defects    quakers
##  Min.   : 0.000   Min.   :0.00000   Min.   : 0.0000      Min.   : 0.0000
##  1st Qu.: 7.250   1st Qu.:0.09000   1st Qu.: 0.0000      1st Qu.: 0.0000
```

```
##  Median : 7.500   Median :0.11000   Median : 0.0000      Median : 0.0000
##  Mean   : 7.503   Mean   :0.08838   Mean   : 0.4795      Mean   : 0.1734
##  3rd Qu.: 7.750   3rd Qu.:0.12000   3rd Qu.: 0.0000      3rd Qu.: 0.0000
##  Max.   :10.000   Max.   :0.28000   Max.   :63.0000      Max.   :11.0000
##                                                          NA's   :1
##     color           category_two_defects  expiration        certification_body
##  Length:1339        Min.   : 0.000       Length:1339        Length:1339
##  Class :character   1st Qu.: 0.000       Class :character   Class :character
##  Mode  :character   Median : 2.000       Mode  :character   Mode  :character
##                     Mean   : 3.556
##                     3rd Qu.: 4.000
##                     Max.   :55.000
##
##  certification_address certification_contact unit_of_measurement
##  Length:1339           Length:1339           Length:1339
##  Class :character      Class :character      Class :character
##  Mode  :character      Mode  :character      Mode  :character
##
##
##
##
##  altitude_low_meters altitude_high_meters altitude_mean_meters
##  Min.   :     1      Min.   :     1       Min.   :     1
##  1st Qu.:  1100      1st Qu.:  1100       1st Qu.:  1100
##  Median :  1311      Median :  1350       Median :  1311
##  Mean   :  1751      Mean   :  1799       Mean   :  1775
##  3rd Qu.:  1600      3rd Qu.:  1650       3rd Qu.:  1600
##  Max.   :190164      Max.   :190164       Max.   :190164
##  NA's   :230         NA's   :230          NA's   :230
```

2. Analysis

2.1 Total Rating Points for Each Country

We hope to compare the overall rating of different country and regions. To better visualize the data, we proposed to use a map chart with a choropleth world map. Choropleth map is a type of thematic map that usually used to represent an aggregate summary of a geographic characteristic (Holtz, 2018). Here, we use a serial of color shades to represent the average/ mean overall score of coffees from different countries. Also, a summary table of the highest score, lowest score, and average score of each country is provided.

```
## 2.1.1 Create a summary table with number of coffee rated, highest, lowest, and
#average rating score for each country.
ratingbycountry <- coffee_ratings %>%
  group_by(country_of_origin) %>%
  summarise(
    count = n(),
    highest_score = max(total_cup_points),
    lowest_score = min(total_cup_points),
    mean_score = mean(total_cup_points)
  )
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```r
## From the ratingbycountry file, we see that some of the country names are not standard
#names. Thus, the first thing to do is to match the country names with standard names
#in the world map.
ratingbycountry[6,1] <- "Ivory Coast"
ratingbycountry[29,1] <- "Tanzania"
ratingbycountry[34,1] <- "Puerto Rico"
ratingbycountry[32,1] <- "USA"
ratingbycountry[33,1] <- "USA(Hawaii)"

## 2.1.2 Create a boxplot to display the coffee rating scores for each country.
countryrating.plot <- ggplot(coffee_ratings) +
  aes(x = country_of_origin, y = total_cup_points) +
  geom_boxplot(fill = "#d8576b") +
  labs(x = "Country of origin", y = "Total cup points", title = "Coffee Rating by Country") +
  theme_linedraw() +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))+
  theme(plot.title = element_text(hjust = 0.5)) +
  ylim(55L, 95L)

## 2.1.3 Create a choropleth map chart for world coffee rating.
## First, we need to load the packages.
require(maps)
```

```
## Loading required package: maps
```

```
##
## Attaching package: 'maps'
```

```
## The following object is masked from 'package:purrr':
##
##     map
```

```r
require(viridis)
```

```
## Loading required package: viridis
```

```
## Loading required package: viridisLite
```

```r
world_map <- map_data("world")
theme_set(
  theme_void()
)
## Second, we are going to select countries that are listed in coffee rating.
#USA and USA(Hawaii) are excluded from the coutry_of_orignin1 group, and they are listed seperately.
country_of_origin1 <- c("Brazil", "Burundi","China","Colombia","Costa Rica","Ivory Coast","Ecuador","El
,"India", "Indonesia","Japan","Kenya","Laos", "Malawi","Mauritius","Mexico"               ,"Myanmar","N
world_map1 <- map_data ("world", region = country_of_origin1)
world_Hawaii <- map_data("world", region = "USA.Hawaii")
world_Hawaii[,5] <- "USA(Hawaii)"
world_USA <- map_data ("world", region = "USA")
world_USA1<- subset(world_USA, subregion != "Hawaii")
```

```r
## Now, merge the map regions.
new_map <- rbind(world_map1,world_Hawaii,world_USA1)
## Create labels for mapping regions.
world_map1_label <- map_data ("world", region = country_of_origin1) %>%
  group_by(region) %>%
  summarise(long = mean(long), lat = mean(lat))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```r
world_Hawaii_label <- map_data("world", region = "USA.Hawaii") %>%
  group_by(region) %>%
  summarise(long = mean(long), lat = mean(lat))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

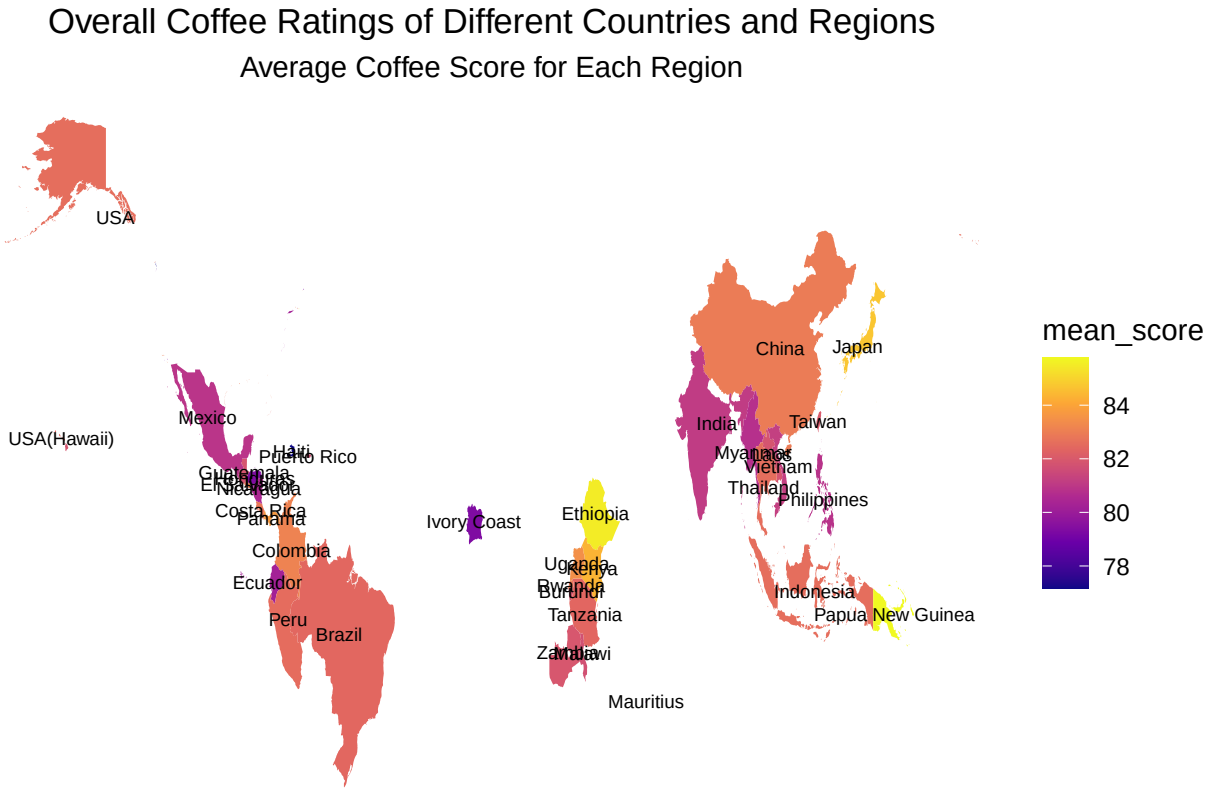
```r
world_Hawaii_label[1,1] <- "USA(Hawaii)"
world_USA1_label<- subset(world_USA, subregion != "Hawaii")%>%
  group_by(region) %>%
  summarise(long = mean(long), lat = mean(lat))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```r
new_map_label <- rbind(world_map1_label,world_Hawaii_label,world_USA1_label)

## Merge coffee rating data with the map data.
world_coffee_rating <- left_join(ratingbycountry,new_map, by=c("country_of_origin" = "region"))

## Create a world map.
ggplot(world_coffee_rating, aes(long, lat)) +
  geom_polygon(aes(group = group, fill = mean_score))+
  geom_text(aes(label = region), data = new_map_label, size = 2.5, hjust = 0.5) +
  scale_fill_viridis_c(option = "C") +
  labs(
    title = "Overall Coffee Ratings of Different Countries and Regions",
    subtitle = "Average Coffee Score for Each Region",
    caption = "Data: coffee_ratings | Creation: Mengyi Dong | CPSC441 Group-7 Project"
  ) +
  theme(plot.title = element_text(hjust = 0.5)) +
  theme(plot.subtitle = element_text(hjust = 0.5))
```



2.2 Sensory Attributes of Coffee Species

```
## Code for spider graph
#2.2.1.1 Create the data sets for sensory atributes. Since Ethiopia has the highest score on
#cup points and Ecuador has the lowest. We will show in a Spider plot how the sensory
#attributes look for both countries.

#Summary of sensory atributes from Ethiopia.
data_sensory_1= select(coffee_ratings,country_of_origin, aroma, flavor, aftertaste, acidity, body, balan

ethiophia_data_sensory= filter(data_sensory_1, country_of_origin== "Ethiopia")
ethiophia_data_sensory
```

```
## # A tibble: 44 x 10
##    country_of_orig~ aroma flavor aftertaste acidity  body balance uniformity
##    <chr>            <dbl>  <dbl>      <dbl>   <dbl> <dbl>   <dbl>      <dbl>
##  1 Ethiopia          8.67   8.83       8.67    8.75  8.5     8.42      10
##  2 Ethiopia          8.75   8.67       8.5     8.58  8.42    8.42      10
##  3 Ethiopia          8.17   8.58       8.42    8.42  8.5     8.25      10
##  4 Ethiopia          8.25   8.5        8.25    8.5   8.42    8.33      10
##  5 Ethiopia          8.25   8.33       8.5     8.42  8.33    8.5       10
##  6 Ethiopia          8.67   8.67       8.58    8.42  8.33    8.42       9.33
##  7 Ethiopia          8.08   8.58       8.5     8.5   7.67    8.42      10
##  8 Ethiopia          8.17   8.67       8.25    8.5   7.75    8.17      10
```

```
##  9 Ethiopia          8.25   8.33       8.5     8.25  8.58    8.75       9.33
## 10 Ethiopia          8.17   8.33       8.25    8.33  8.42    8.33       9.33
## # ... with 34 more rows, and 2 more variables: sweetness <dbl>, moisture <dbl>
```

```
graph_sensory_et= select(ethiophia_data_sensory, aroma, flavor, aftertaste, acidity, body, balance, uni
graph_sensory_et
```

```
## # A tibble: 44 x 9
##    aroma flavor aftertaste acidity  body balance uniformity sweetness moisture
##    <dbl>  <dbl>      <dbl>   <dbl> <dbl>   <dbl>      <dbl>     <dbl>    <dbl>
##  1  8.67   8.83       8.67    8.75  8.5     8.42      10        10        0.12
##  2  8.75   8.67       8.5     8.58  8.42    8.42      10        10        0.12
##  3  8.17   8.58       8.42    8.42  8.5     8.25      10        10        0.11
##  4  8.25   8.5        8.25    8.5   8.42    8.33      10        10        0.12
##  5  8.25   8.33       8.5     8.42  8.33    8.5       10         9.33     0.03
##  6  8.67   8.67       8.58    8.42  8.33    8.42       9.33      9.33     0.03
##  7  8.08   8.58       8.5     8.5   7.67    8.42      10        10        0.1
##  8  8.17   8.67       8.25    8.5   7.75    8.17      10        10        0.1
##  9  8.25   8.33       8.5     8.25  8.58    8.75       9.33      9.33     0.05
## 10  8.17   8.33       8.25    8.33  8.42    8.33       9.33      9.33     0.05
## # ... with 34 more rows
```

```
class(graph_sensory_et)
```

```
## [1] "tbl_df"     "tbl"        "data.frame"
```

```
#Summary of sensory atributes from Ecuador
ecuador_data_sensory= filter(data_sensory_1, country_of_origin== "Ecuador")
ecuador_data_sensory
```

```
## # A tibble: 3 x 10
##   country_of_orig~ aroma flavor aftertaste acidity  body balance uniformity
##   <chr>            <dbl>  <dbl>      <dbl>   <dbl> <dbl>   <dbl>      <dbl>
## 1 Ecuador           7.5    7.67       7.58    7.75  7.83    7.83         10
## 2 Ecuador           7.75   7.58       7.33    7.58  5.08    7.83         10
## 3 Ecuador           7.5    7.67       7.75    7.75  5.17    5.25         10
## # ... with 2 more variables: sweetness <dbl>, moisture <dbl>
```

```
graph_sensory_ec= select(ecuador_data_sensory, aroma, flavor, aftertaste, acidity, body, balance, unifo
graph_sensory_ec
```

```
## # A tibble: 3 x 9
##   aroma flavor aftertaste acidity  body balance uniformity sweetness moisture
##   <dbl>  <dbl>      <dbl>   <dbl> <dbl>   <dbl>      <dbl>     <dbl>    <dbl>
## 1  7.5    7.67       7.58    7.75  7.83    7.83         10     10        0.09
## 2  7.75   7.58       7.33    7.58  5.08    7.83         10      7.75     0
## 3  7.5    7.67       7.75    7.75  5.17    5.25         10      8.42     0
```

```
class(graph_sensory_ec)
```

```
## [1] "tbl_df"     "tbl"        "data.frame"
```

```r
#Average of summary attributes from Ethiopia and Ecuador
Ethiopia = colMeans(graph_sensory_et[sapply(graph_sensory_et, is.numeric)])
Ecuador = colMeans(graph_sensory_ec[sapply(graph_sensory_et, is.numeric)])

df_et_ecu = as.data.frame(t(cbind(Ethiopia, Ecuador)))
df_et_ecu
```

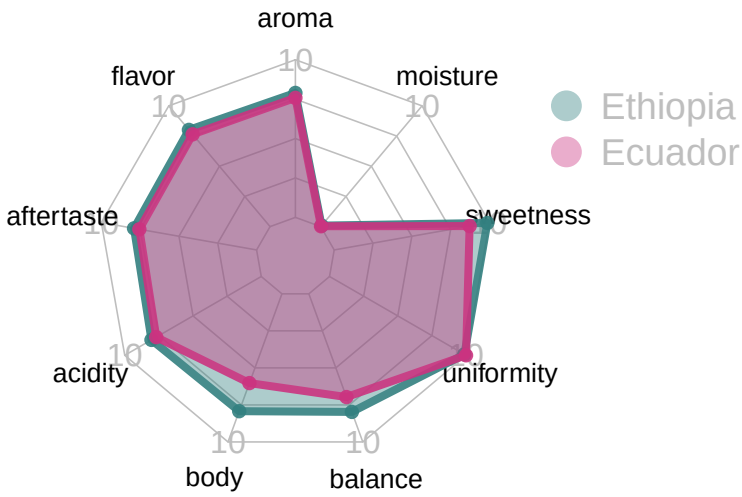
```
##             aroma   flavor aftertaste  acidity     body  balance uniformity
## Ethiopia 7.896364 8.009091   7.893864 8.043636 7.924091 7.972273   9.878409
## Ecuador  7.583333 7.640000   7.553333 7.693333 6.026667 6.970000  10.000000
##          sweetness   moisture
## Ethiopia  9.863409 0.08295455
## Ecuador   8.723333 0.03000000
```

```r
#To use the fmbs package, We have to add 2 lines to the dataframe: max and min
#of each variable to show on plot
data_sensory_graph_ethiopia_ecuador= rbind(rep(10,5), rep(0,5), df_et_ecu)

#Color vector
colors_border=c( rgb(0.2,0.5,0.5,0.9), rgb(0.8,0.2,0.5,0.9) , rgb(0.7,0.5,0.1,0.9) )
colors_in=c( rgb(0.2,0.5,0.5,0.4), rgb(0.8,0.2,0.5,0.4) , rgb(0.7,0.5,0.1,0.4) )

#Radarchart
radarchart(data_sensory_graph_ethiopia_ecuador, axistype=2 ,
           pcol=colors_border , pfcol=colors_in , plwd=4 , plty=1,
           cglcol="grey", cglty=1, axislabcol="grey", caxislabels=seq(0,20,5), cglwd=0.8,
           vlcex=0.8, title = "Sensory Atributes of Coffee from Ethiopia and Ecuador",
    )

legend(x=1.2, y=1, legend = rownames(df_et_ecu[1:2,]), bty = "n", pch=20 , col=colors_in , text.col = "
```



2.3 The Effect of Type of Species and Processing Method on the Selected Sensory Attributes

After we have analyzed the sensory attributes, we would like to investigate the effect of processing methods,type of species and their interaction on them. For this purpose, we've run 2-way ANOVA for selected sensory attributes (acidity, aroma, aftertaste, balance, flavor and sweetness). In order to perform this analysis, first, we've created linear models for each attribute. The structure of these models were as follows: Z ~ X * Y where "Z" is the outcome / dependent variable, like *aroma*, and "X" and "Y" are independent variables, like *processing method* and *species*.

```
## Code for ANOvA statistical analysis
#First, we removed the missing values from the dataset:
coffee_ratings2 <- na.omit(coffee_ratings)
write.csv(coffee_ratings2, "coffee_ratings2.csv")

#acidity
model_1 = lm(acidity ~ processing_method * species, data=coffee_ratings2)
model_acidity = aov(model_1)
summary(model_acidity)
```

```
##                            Df Sum Sq Mean Sq F value  Pr(>F)
## processing_method           3  0.845 0.28178   4.439 0.00531 **
## species                     1  0.011 0.01124   0.177 0.67463
## processing_method:species   1  0.048 0.04849   0.764 0.38379
```

```
## Residuals                 126  7.998 0.06348
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#aroma
model_2 <- lm(aroma ~ processing_method * species, data=coffee_ratings2)
model_aroma = aov(model_2)
summary(model_aroma)
```

```
##                            Df Sum Sq Mean Sq F value Pr(>F)
## processing_method           3  0.154 0.05122   1.831  0.145
## species                     1  0.025 0.02504   0.895  0.346
## processing_method:species   1  0.001 0.00084   0.030  0.863
## Residuals                 126  3.524 0.02797
```

```
#aftertaste
model_3 <- lm(aftertaste ~ processing_method * species, data=coffee_ratings2)
model_aftertaste = aov(model_3)
summary(model_aftertaste)
```

```
##                            Df Sum Sq Mean Sq F value  Pr(>F)
## processing_method           3  0.710 0.23653   4.349 0.00595 **
## species                     1  0.120 0.11973   2.202 0.14037
## processing_method:species   1  0.004 0.00362   0.067 0.79674
## Residuals                 126  6.852 0.05438
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#balance
model_4 <- lm(balance~ processing_method * species, data=coffee_ratings2)
model_balance = aov(model_4)
summary(model_balance)
```

```
##                            Df Sum Sq Mean Sq F value Pr(>F)
## processing_method           3  0.571 0.19036   3.143 0.0276 *
## species                     1  0.169 0.16893   2.790 0.0974 .
## processing_method:species   1  0.072 0.07190   1.187 0.2780
## Residuals                 126  7.631 0.06056
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
#flavor
model_5 <- lm(flavor ~ processing_method * species, data=coffee_ratings2)
model_flavor = aov(model_5)
summary(model_flavor)
```

```
##                            Df Sum Sq Mean Sq F value Pr(>F)
## processing_method           3  0.357 0.11892   2.601  0.055 .
## species                     1  0.102 0.10214   2.234  0.138
## processing_method:species   1  0.003 0.00252   0.055  0.815
## Residuals                 126  5.761 0.04572
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
#sweetness
model_6 <- lm(sweetness ~ processing_method * species, data=coffee_ratings2)
model_sweetness = aov(model_6)
summary(model_sweetness)
```

```
##                            Df Sum Sq Mean Sq F value   Pr(>F)
## processing_method           3  2.558   0.853   6.346 0.000484 ***
## species                     1  8.915   8.915  66.339 3.18e-13 ***
## processing_method:species   1  0.026   0.026   0.195 0.659301
## Residuals                 126 16.933   0.134
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

After that, we highlighted which variable(s) have a significant effect on these sensory attributes by gathering the information we obtained from previous ANOVA analysis in the table below.

```r
#Factors that have signifcant effects on sensory attributes

A <- c('Acidity','Aroma','Aftertaste','Balance','Balance','Flavor','Sweetness','Sweetness')
B <- c('process_method','none','process_method', 'process_method', 'species', 'process_method', 'proces
C <- c( 0.01, "none", 0.001, 0.001,0.05,0.05,0,0)
D <- cbind(A,B,C)
colnames(D) <- c('Sensory_attributes', 'Factor','Significance Level')
knitr::kable(D, colnames = c('Sensory Attributes', 'Factor','Significance Level'), align = "lcc", forma
```

Sensory_attributes	Factor	Significance Level
Acidity	process_method	0.01
Aroma	none	none
Aftertaste	process_method	0.001
Balance	process_method	0.001
Balance	species	0.05
Flavor	process_method	0.05
Sweetness	process_method	0
Sweetness	species	0

One of the outcomes shown at the table above, processing methods have a significant effect of flavor at p=0.05 significance level. However, we still don't know which processing method has the most effect on the flavor. Therefore, we would like to run an ls-means test for processing method variable.

```r
#In order to do this analysis, first, we installed the following packages:

library("lsmeans")
```

```
## Loading required package: emmeans
```

```
## The 'lsmeans' package is now basically a front end for 'emmeans'.
## Users are encouraged to switch the rest of the way.
## See help('transition') for more information, including how to
## convert old 'lsmeans' objects and scripts to work with 'emmeans'.
```

```r
library("multcompView")
library("plyr")
```

```
## ------------------------------------------------------------------------------
```

```
## You have loaded plyr after dplyr - this is likely to cause problems.
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:
## library(plyr); library(dplyr)
```

```
## ------------------------------------------------------------------------------
```

```
##
## Attaching package: 'plyr'
```

```
## The following object is masked from 'package:maps':
##
##     ozone
```

```
## The following objects are masked from 'package:dplyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
```

```
## The following object is masked from 'package:purrr':
##
##     compact
```

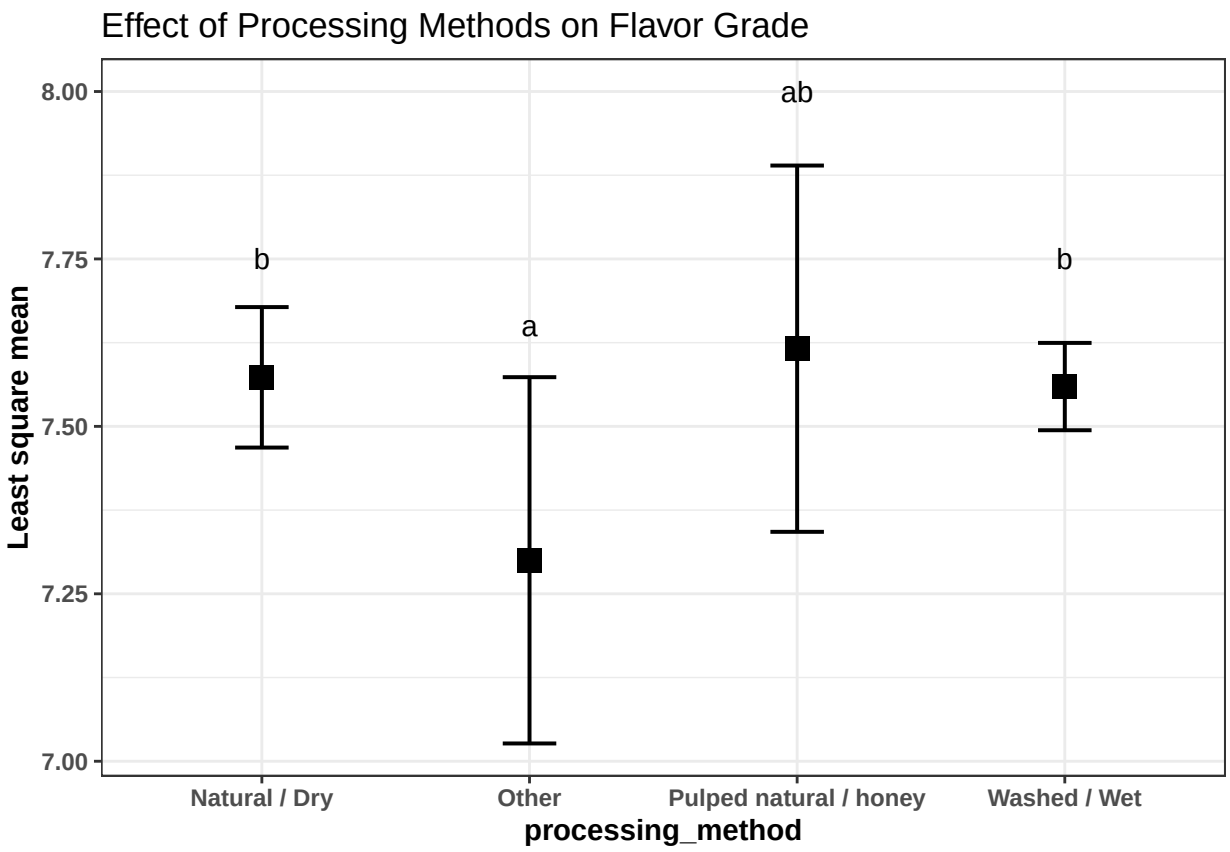
```r
#Again, we created our model containing flavor and processing_method variables:

model_fp <- lm(flavor ~ processing_method, data=coffee_ratings2)
fp = lsmeans(model_fp, pairwise~processing_method, adjust="tukey")
fp2 <- as.data.frame(fp)
fp3 = fp2[1:4,-2]

#Finally, we would like to plot lsmeans vs processing method graph:

library(ggplot2)
pd = position_dodge(0.4)
p <- ggplot(fp3,
       aes(x=processing_method,
           y=lsmean))+
  geom_point(shape = 15,
             size = 4,
             position = pd)+
  geom_errorbar(aes(ymin = lower.CL,
                    ymax = upper.CL),
                    width = 0.2,
                    size = 0.7,
                    position = pd)+
  geom_text(label = c("b", "a", "ab","b"), aes(y = c(7.75, 7.65, 8.00, 7.75), x = processing_method), si
  theme_bw()+
```

```
  theme(axis.title = element_text(face = "bold"),
        axis.text = element_text(face = "bold"),
        plot.caption = element_text(hjust = 0)) +
  ylab("Least square mean")+
  ggtitle("Effect of Processing Methods on Flavor Grade")
p
```




```
ggsave('p.png',width=6, height=4,dpi=300)
```

3. Conclusion

In this project, we explored the "coffee_rating" dataset in "TidyTuesday". First, we demonstrated the overall quality of coffee for different countries using a choropleth map; the result showed that Ethiopia and Papua New Guinea had the highest average rating scores, while Ivory Coast and Ecuador had the lowest rating scores. Then, we decided to plot in a spider chart the sensory attributes from Ethiopia and Ecuador that has the highest and the lowest rating scores respectively in order to see which attribute affects the cup score. We found that balance, body and sweetness are the attributes that affect the cup score in Ecuadorian coffee. Next, we performed a two-way ANOVA analysis to investigate if the sensory attributes of coffee are significantly affected by processing methods, type of coffee species or their interactions. Finally, we applied ls-means test to understand whether there is a significant differences between processing methods in terms of their impact on flavor of the coffee species.

4. References

1. Institute for Scientific Information on Coffee (ISIC). (2014, November 24). Where Coffee Grows. Retrieved from https://www.coffeeandhealth.org/all-about-coffee/where-coffee-grows/

2. National Coffee Association (NCA). (2020). The History of Coffee. Retrieved from https://www.ncausa.org/about-coffee/history-of-coffee

3. Holtz, Yan.(2018). Choropleth Map. Retrived from https://www.r-graph-gallery.com/choropleth-map.html

4. kassambara. (2020). How to Create a Map using GGPlot2. Retrived from https://www.datanovia.com/en/blog/how-to-create-a-map-using-ggplot2/

5. Mangiafico, S. S. (2016). Least Square Means for Multiple Comparisons. Retrieved October 16, 2020, from https://rcompanion.org/handbook/G_06.html