COCO BEAN ANALYSIS

By

Nithesh K

22CESG23

I Msc data Analytics

Data source: Data.world

Problem:

If we want to open chocolate outlet, we have to analyze the raw materials quality and find which country produce more coco percent bean.

Dataset Description:

- Company – Company which produce Coco bean
- Specific Bean Origin – Bean Bar Origin
- REF – Reviews of that product
- Review Date – Date of that reviews
- Cocoa Percentage - % of coco bean
- Company Location – Location of that company
- Rating – Rating of that product
- Bean Type – Type of bean
- Broad Bean – Bean origin

Assumption:

- We need the purest coco percentage because of problem statement. So, we have to filter the Attributes more that than 80%
- Not only we want purest Coco we also need the quality of the Coco bean. So, we filter the ratings of the Coco bean greater than 3.75 and reviews which have the highest

```
library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

library(ggplot2)
library(lattice)
library(readr)
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 objects are masked from 'package:dplyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize

df = read_csv("flavors_of_cacao.csv")

## Rows: 1795 Columns: 9

## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## chr (5): Company
## (Maker-if known), Specific Bean Origin
## or Bar Name, Company...
## dbl (4): REF, Review
## Date, Cocoa
## Percent, Rating
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.

head(df,5)

## # A tibble: 5 × 9
##   Company\n(Maker…¹ Speci…²   REF Revie…³ Cocoa…⁴ Compa…⁵ Rating Bean\…⁶ Broad…⁷
##   <chr>             <chr>   <dbl>   <dbl>   <dbl> <chr>    <dbl> <chr>   <chr>
## 1 A. Morin          Agua G…  1876    2016      63 France    3.75 <NA>    Sao To…
## 2 A. Morin          Kpime    1676    2015      70 France    2.75 <NA>    Togo
## 3 A. Morin          Atsane   1676    2015      70 France    3    <NA>    Togo
```

```
## 4 A. Morin          Akata    1680    2015      70 France    3.5  <NA>    Togo
## 5 A. Morin          Quilla   1704    2015      70 France    3.5  <NA>    Peru
## # … with abbreviated variable names ¹`Company\n(Maker-if known)`,
## #   ²`Specific Bean Origin\nor Bar Name`, ³`Review\nDate`, ⁴`Cocoa\nPercent`,
## #   ⁵`Company\nLocation`, ⁶`Bean\nType`, ⁷`Broad Bean\nOrigin`

colSums(is.na(df))

##         Company\n(Maker-if known) Specific Bean Origin\nor Bar Name
##                                 0                                  0
##                               REF                      Review\nDate
##                                 0                                  0
##                   Cocoa\nPercent                 Company\nLocation
##                                 0                                  0
##                            Rating                       Bean\nType
##                                 0                                888
##               Broad Bean\nOrigin
##                                52

#coloumns
n <- length(colnames(df))
vec1 <- c(1:n)
data.frame(vec1, colnames(df))

##   vec1                       colnames.df.
## 1    1         Company\n(Maker-if known)
## 2    2 Specific Bean Origin\nor Bar Name
## 3    3                                REF
## 4    4                      Review\nDate
## 5    5                    Cocoa\nPercent
## 6    6                  Company\nLocation
## 7    7                             Rating
## 8    8                        Bean\nType
## 9    9                Broad Bean\nOrigin

#rename coloumns
colnames(df)=c("company","origin_bar","reviews","review_date","coco_per","location","rati
ng","bean_type","bean_origin")
head(df)

## # A tibble: 6 × 9
##   company  origin_bar  reviews review_d…¹ coco_…² locat…³ rating bean_…⁴ bean_…⁵
##   <chr>    <chr>         <dbl>      <dbl>   <dbl> <chr>    <dbl> <chr>   <chr>
## 1 A. Morin Agua Grande    1876       2016      63 France    3.75 <NA>    Sao To…
## 2 A. Morin Kpime          1676       2015      70 France    2.75 <NA>    Togo
## 3 A. Morin Atsane         1676       2015      70 France    3    <NA>    Togo
## 4 A. Morin Akata          1680       2015      70 France    3.5  <NA>    Togo
## 5 A. Morin Quilla         1704       2015      70 France    3.5  <NA>    Peru
## 6 A. Morin Carenero       1315       2014      70 France    2.75 Criollo Venezu…
## # … with abbreviated variable names ¹review_date, ²coco_per, ³location,
## #   ⁴bean_type, ⁵bean_origin

#filling empty strings
df["bean_type"][is.na(df["bean_type"])] <- "Unknown"
df["bean_origin"][is.na(df["bean_origin"])] <- "Unknown"
count(df,"bean_type")
```

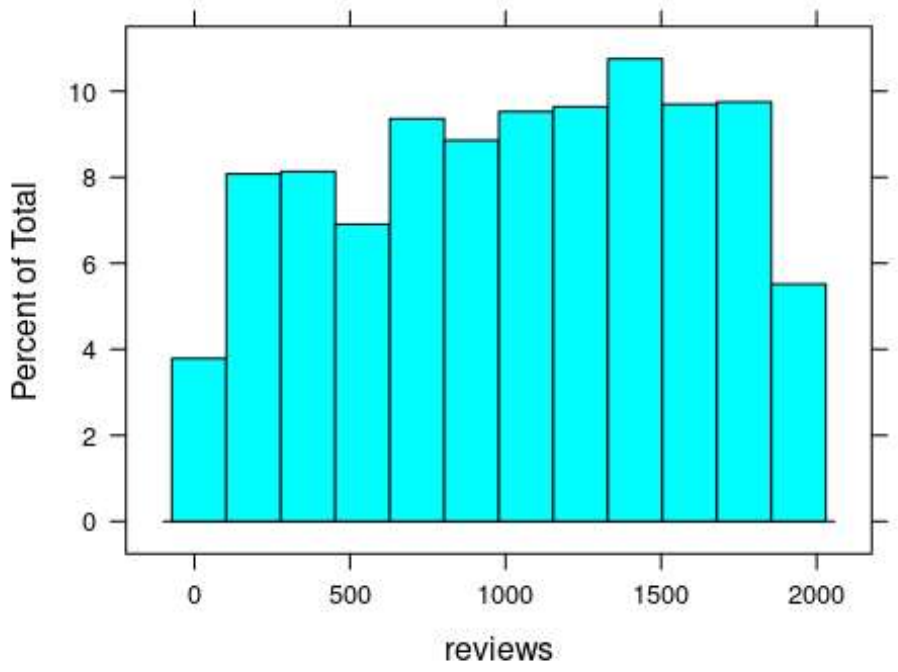
```
##                   bean_type freq
## 1                    Amazon    1
## 2                Amazon mix    2
## 3               Amazon, ICS    2
## 4                   Beniano    3
## 5                     Blend   41
## 6   Blend-Forastero,Criollo    1
## 7                     CCN51    1
## 8                   Criollo  153
## 9          Criollo (Amarru)    2
## 10     Criollo (Ocumare 61)    2
## 11     Criollo (Ocumare 67)    1
## 12     Criollo (Ocumare 77)    1
## 13        Criollo (Ocumare)    1
## 14      Criollo (Porcelana)   10
## 15           Criollo (Wild)    1
## 16               Criollo, +    1
## 17       Criollo, Forastero    2
## 18      Criollo, Trinitario   39
## 19                     EET    3
## 20                Forastero   87
## 21    Forastero (Amelonado)    1
## 22       Forastero (Arriba)   37
## 23   Forastero (Arriba) ASS    6
## 24  Forastero (Arriba) ASSS    1
## 25      Forastero (Catongo)    2
## 26     Forastero (Nacional)   52
## 27    Forastero (Parazinho)    8
## 28    Forastero, Trinitario    1
## 29   Forastero(Arriba, CCN)    1
## 30                   Matina    3
## 31                 Nacional    2
## 32        Nacional (Arriba)    3
## 33               Trinitario  419
## 34 Trinitario (85% Criollo)    2
## 35   Trinitario (Amelonado)    1
## 36     Trinitario (Scavina)    1
## 37      Trinitario, Criollo    9
## 38    Trinitario, Forastero    2
## 39     Trinitario, Nacional    1
## 40         Trinitario, TCGA    1
## 41                   Unknown  888
```

```
#summary
summary(df)
```

```
##    company           origin_bar           reviews      review_date
##  Length:1795        Length:1795        Min.   :   5   Min.   :2006
##  Class :character   Class :character   1st Qu.: 576   1st Qu.:2010
##  Mode  :character   Mode  :character   Median :1069   Median :2013
##                                        Mean   :1036   Mean   :2012
##                                        3rd Qu.:1502   3rd Qu.:2015
##                                        Max.   :1952   Max.   :2017
##     coco_per       location             rating       bean_type
##  Min.   : 42.0   Length:1795        Min.   :1.000   Length:1795
##  1st Qu.: 70.0   Class :character   1st Qu.:2.875   Class :character
```

```
##  Median : 70.0   Mode  :character   Median :3.250   Mode  :character
##  Mean   : 71.7                      Mean   :3.186
##  3rd Qu.: 75.0                      3rd Qu.:3.500
##  Max.   :100.0                      Max.   :5.000
##  bean_origin
##  Length:1795
##  Class :character
##  Mode  :character
##
##
##
```

```
histogram(~reviews,df)
```



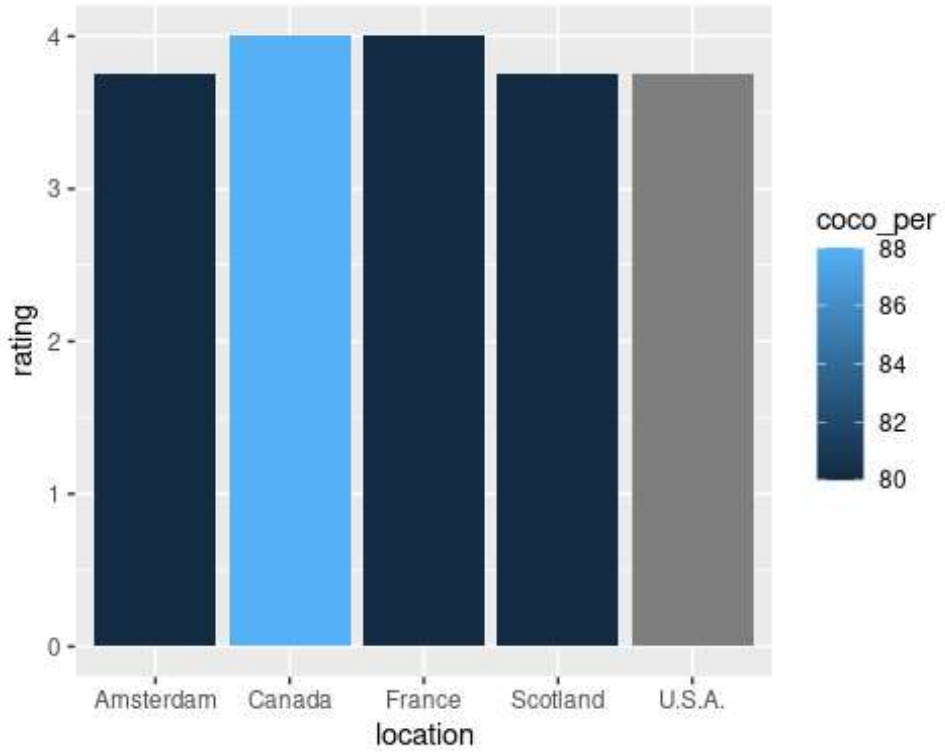

```
#choco percent
df2= subset(df, coco_per >=80)

#rating of a choco
df3= subset(df2,rating>=3.75)
count(df3,"location")
```

```
##    location freq
## 1 Amsterdam    1
## 2    Canada    1
## 3    France    1
## 4  Scotland    1
## 5    U.S.A.    4
```

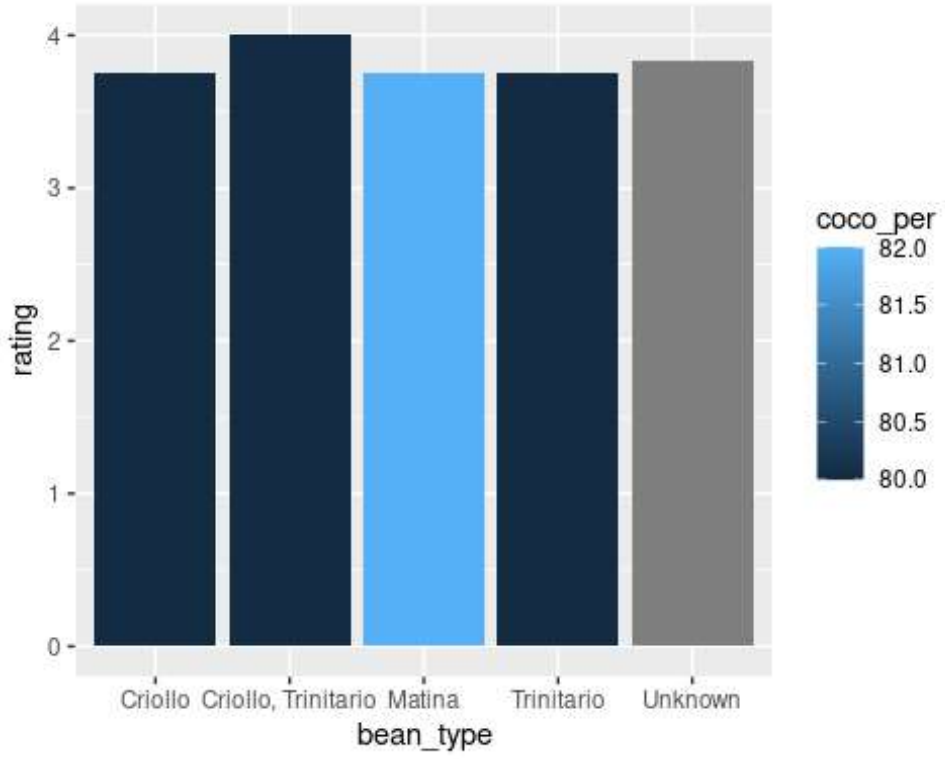
```
ggplot(data = df3, aes(x = location, y=rating)) +
  geom_bar(stat="summary", aes(fill=coco_per))
```

```
## No summary function supplied, defaulting to `mean_se()`
```




```
ggplot(data = df3, aes(x = bean_type, y=rating)) +
  geom_bar(stat="summary", aes(fill=coco_per))
```

```
## No summary function supplied, defaulting to `mean_se()`
```




```
head(df3,5)
```

```
## # A tibble: 5 × 9
##   company         origi…¹ reviews revie…² coco_…³ locat…⁴ rating bean_…⁵ bean_…⁶
```

```
##   <chr>           <chr>     <dbl>   <dbl>   <dbl> <chr>    <dbl> <chr>   <chr>
## 1 Chocolate Make… Peru, …    1530    2015      80 Amster…   3.75 Criollo Peru
## 2 Chocolate Tree… Carene…    1582    2015      80 Scotla…   3.75 Trinit… Venezu…
## 3 Ethereal        Domini…    1275    2014      80 U.S.A.    3.75 Unknown Domini…
## 4 Potomac         Upala,…     607    2010      82 U.S.A.    3.75 Matina  Costa …
## 5 Pralus          Fortis…      93    2006      80 France    4    Crioll… Ecuador
## # … with abbreviated variable names ¹origin_bar, ²review_date, ³coco_per,
## #   ⁴location, ⁵bean_type, ⁶bean_origin
```

Insight:

- From fig-1
  - The max reviews for the chocolate bean type are around 1400-1500
- From fig-2
  - The Country Canada has the highest rating also produce purest form of coco Powder among all the countries.
  - The Country France and Amsterdam takes place a and 3 respectively with producing Same amount of coco percentage with average rating of 4 and 3.75 respectively.
- From fig-3
  - The bean type Matina is the purest loco form with average ratings of 3.75
  - Though Trinitario bean type have the highest bean raking it has lowest amount of Coco percentage.

Inference:

So, we have to buy the Coco bean of Matina from the Canada who sold the highest percentage of Coco bean with highest rating among all the countries and Coco bean type.