Final_Project

Henry Lewis

18/12/2021

The dataset chosen for this assignment is called Craft Beers dataset. Description: This dataset contains a list of 2,410 US craft beers and 510 US breweries and its available at https://www.kaggle.com/nickhould/craft-cans

```
# Import Libraries
library("dplyr")
```

```
## Warning: package 'dplyr' was built under R version 4.1.2
```

```
##
## 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
```

Part 1: Analysis

Beer Dataset

```
# Read the beers csv to a variable beer with first row containing column names and removing white space
beer <- read.csv("beers.csv", header = TRUE, strip.white=TRUE)
# set the column name to beer_id
colnames(beer)[4] <- c("beer_id")
# set the column name to beer_name
colnames(beer)[5] <- c("beer_name")
# Remove the index column
beer_df <- beer[,-1]
```

Breweries Dataset

```
# Read the breweries csv to a variable breweries with first row containing column names and removing wh
breweries <- read.csv("breweries.csv", header = TRUE, strip.white=TRUE)
# set the column name to brewery_name
colnames(breweries)[2] <- c("brewery_name")
# set the column id to brewery_id
colnames(breweries)[1] <- c("brewery_id")
```

Merged dataset

```
# merge the beer and breweries and remove the first column
df <- merge(beer_df, breweries, by.x = "brewery_id")
df <- df[,-1]
head(df)
```

```
##     abv ibu beer_id     beer_name                                style ounces
## 1 0.045  50    2692  Get Together                         American IPA     16
## 2 0.049  26    2691 Maggie's Leap                   Milk / Sweet Stout     16
## 3 0.048  19    2690    Wall's End                    English Brown Ale     16
## 4 0.060  38    2689       Pumpion                          Pumpkin Ale     16
## 5 0.060  25    2688    Stronghold                       American Porter     16
## 6 0.056  47    2687   Parapet ESB Extra Special / Strong Bitter (ESB)     16
##        brewery_name        city state
## 1 NorthGate Brewing Minneapolis    MN
## 2 NorthGate Brewing Minneapolis    MN
## 3 NorthGate Brewing Minneapolis    MN
## 4 NorthGate Brewing Minneapolis    MN
## 5 NorthGate Brewing Minneapolis    MN
## 6 NorthGate Brewing Minneapolis    MN
```

Breweries Analysis

```
breweries_df <- breweries[,-1]
head(breweries_df)
```

```
##                brewery_name          city state
## 1         NorthGate Brewing   Minneapolis    MN
## 2 Against the Grain Brewery    Louisville    KY
## 3  Jack's Abby Craft Lagers    Framingham    MA
## 4 Mike Hess Brewing Company     San Diego    CA
## 5   Fort Point Beer Company San Francisco    CA
## 6     COAST Brewing Company    Charleston    SC
```

The breweries dataframe contains 558 observations and 3 columns that include the brewery name, city location, and state within the United States where the brewery is located.

```
# Structure of teh Breweries dataset
str(breweries_df)
```

```
## 'data.frame':    558 obs. of  3 variables:
##  $ brewery_name: chr  "NorthGate Brewing" "Against the Grain Brewery" "Jack's Abby Craft Lagers" "Mi
##  $ city        : chr  "Minneapolis" "Louisville" "Framingham" "San Diego" ...
##  $ state       : chr  "MN" "KY" "MA" "CA" ...
```

Analyse the number of Breweries

```
#Number of breweries per state
state_breweries <- table(breweries_df$state)
state_breweries
```

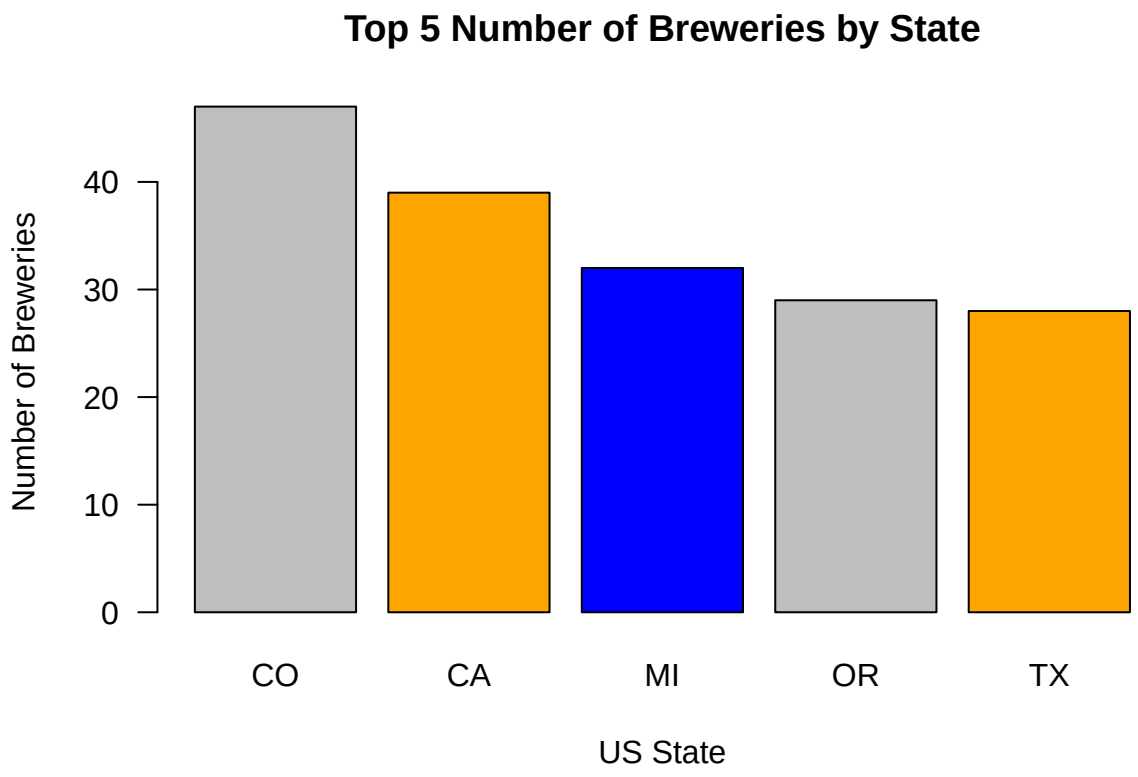
```
##
## AK AL AR AZ CA CO CT DC DE FL GA HI IA ID IL IN KS KY LA MA MD ME MI MN MO MS
##  7  3  2 11 39 47  8  1  2 15  7  4  5  5 18 22  3  4  5 23  7  9 32 12  9  2
## MT NC ND NE NH NJ NM NV NY OH OK OR PA RI SC SD TN TX UT VA VT WA WI WV WY
```

```
##  9 19  1  5  3  3  4  2 16 15  6 29 25  5  4  1  3 28  4 16 10 23 20  1  4
```

```
max5_state_breweries <-head(sort(state_breweries,decreasing = TRUE),5)
max5_state_breweries
```

```
##
## CO CA MI OR TX
## 47 39 32 29 28
```

```
colors = c("gray", "orange", "blue")
barplot(max5_state_breweries, main = "Top 5 Number of Breweries by State", xlab ="US State", ylab = "Nu
col= colors, las = 1)
```



As can be seen Colorado CO has the largest quantity of breweries with 47. Then comes California with 39, Michigan with 32, Oregon with 29 and Texas with 28.

```
colorado_brew <- breweries_df[which(breweries_df$state == "CO"),]
colorado_breweries <- colorado_brew[1]
nrow(colorado_breweries)
```

```
## [1] 47
```

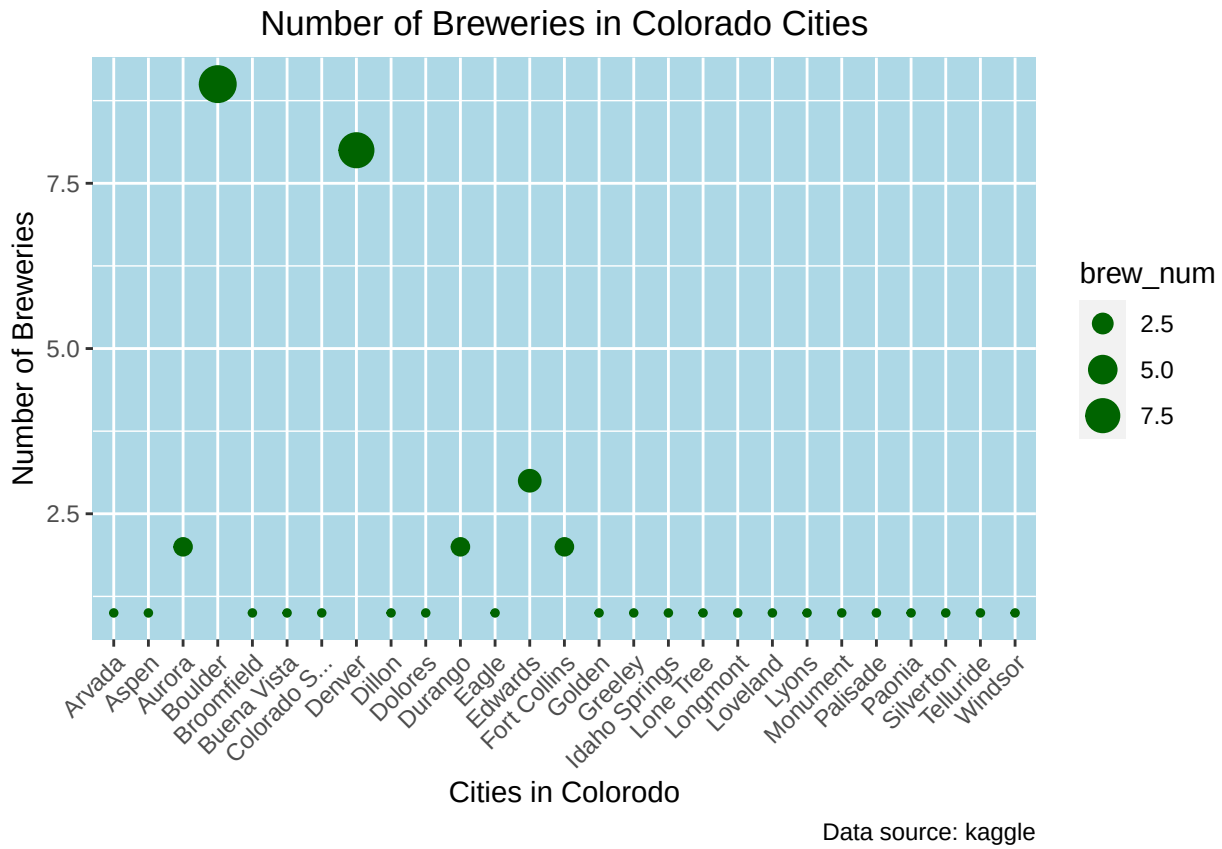
```
colorado_brew_cities <- colorado_brew[1:2]

brewery_cities <- colorado_brew_cities %>%
group_by(city) %>% summarize(n())

brewery_cities <- as.data.frame(brewery_cities)
colnames(brewery_cities)[1] <- c("city")
```

```
colnames(brewery_cities)[2] <- c("brew_num")
```

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.1.2
```

```
ggplot(brewery_cities, aes(x=city, y=brew_num, size = brew_num)) +
    geom_point(color = "darkgreen")+
  # angles the labels
  theme(axis.text.x = element_text(angle = 45, hjust=1))+
  # truncates labels
  scale_x_discrete(label = function(x) stringr::str_trunc(x, 13)) + ggtitle("Number of Breweries in Colo
  # Adds a theme and adds a centered title
  theme(plot.title = element_text(hjust = 0.5))+
  # Adds detailed caption information alongside a theme
  labs(caption = "Data source: kaggle",
              x = "Cities in Colorodo", y = "Number of Breweries") + theme(
                                          panel.background = element_rect(fill = "lightblue",
                                                                          colour = "lightblue",
                                                                          size = 0.5, linetype = "solid
                                          panel.grid.major = element_line(size = 0.5, linetype = 'soli
                                                                          colour = "white"),
                                          panel.grid.minor = element_line(size = 0.25, linetype = 'so
                                                                          colour = "white")
                                          )
```



Analyse the Beers

```
# first 5 lines of the beers dataset
head(beer_df)
```

```
##     abv ibu beer_id           beer_name                          style
## 1 0.050  NA    1436            Pub Beer            American Pale Lager
## 2 0.066  NA    2265         Devil's Cup        American Pale Ale (APA)
## 3 0.071  NA    2264 Rise of the Phoenix                  American IPA
## 4 0.090  NA    2263            Sinister American Double / Imperial IPA
## 5 0.075  NA    2262       Sex and Candy                  American IPA
## 6 0.077  NA    2261        Black Exodus                 Oatmeal Stout
##   brewery_id ounces
## 1        408     12
## 2        177     12
## 3        177     12
## 4        177     12
## 5        177     12
## 6        177     12
```

The total number of beers contained within the dataset is 2410.

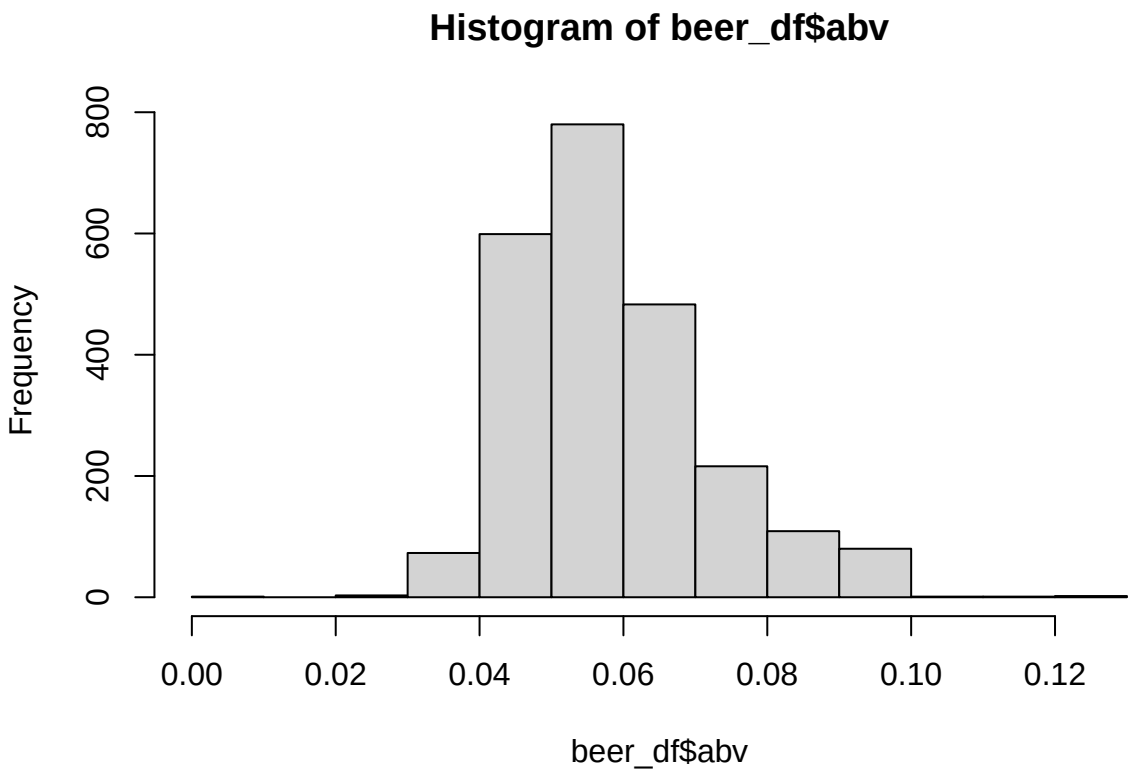
```
# function for calculating the number of N/A values
beer_missing <- sapply(beer_df, function(x)sum(is.na(x)))
```

The total number of missing values contained within the ABV (Alcohol By Volume) column is 62 representing 2.5726141% of the dataset, whilst the number of missing values contained within IBU (International Bitterness Units) is 1005 representing 41.7012448 % of the dataset. The remaining columns have no missing values.

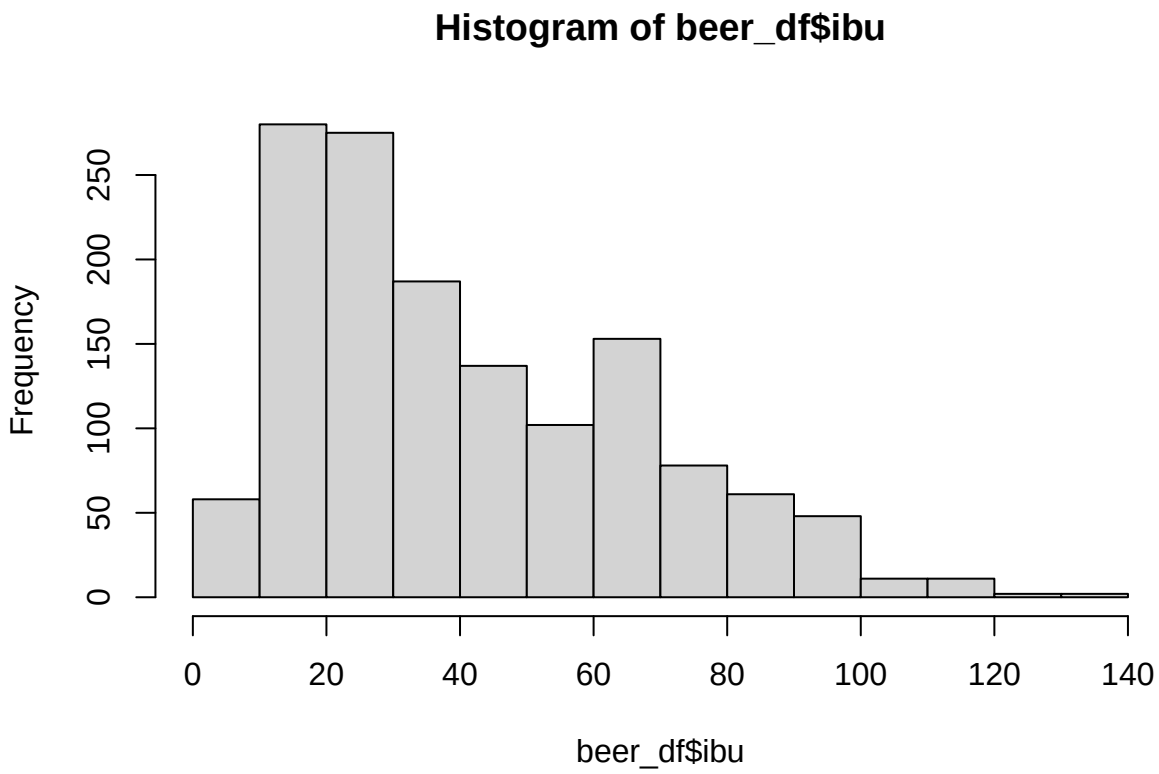
```
beer_averages <- sapply(beer_df[1:2], function(x) mean(x, na.rm = TRUE))
beer_averages
```

```
##         abv         ibu
##  0.05977342 42.71316726
```

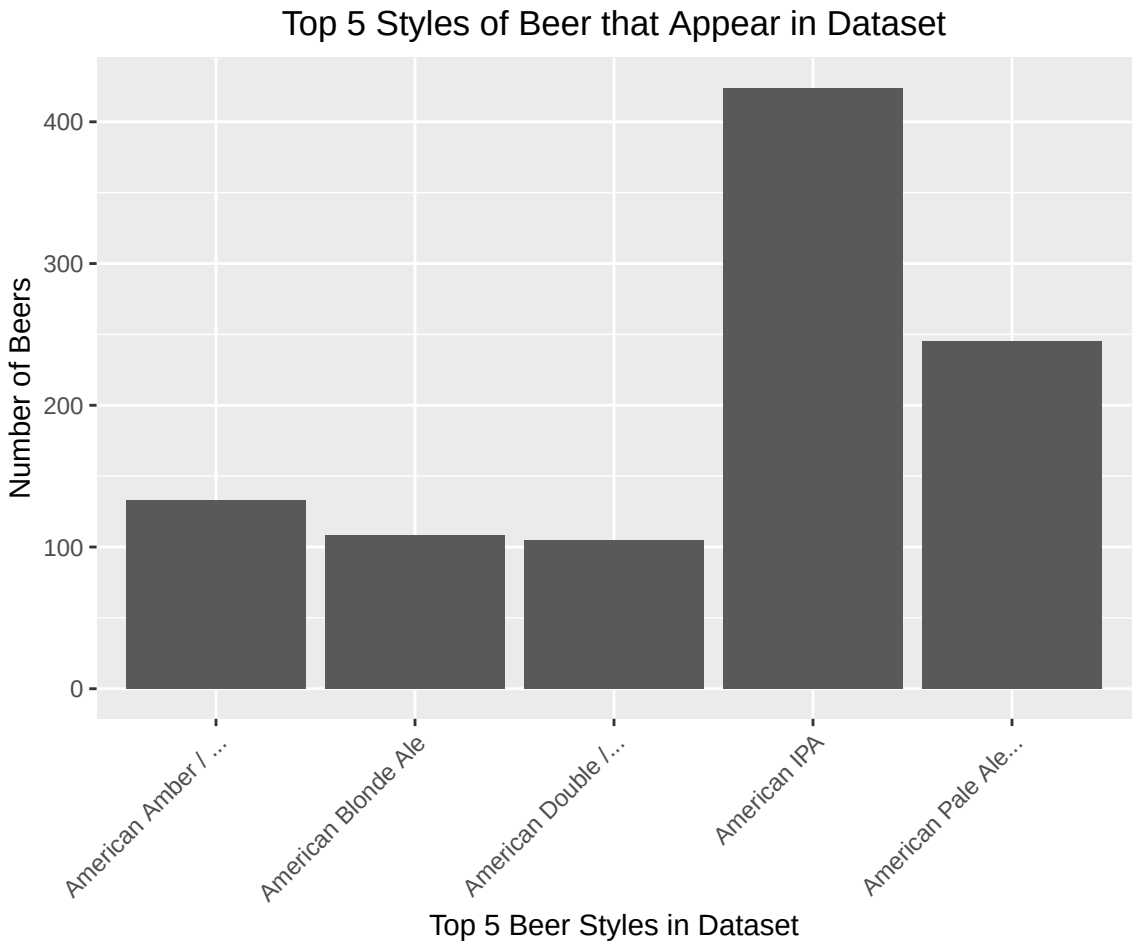
```
hist(beer_df$abv)
```




```
hist(beer_df$ibu)
```




```
# count the number of each beer type
beer_type <- beer_df %>%
  count(style)

df_bt <-beer_type[order(beer_type$n,decreasing = TRUE),]
# get the top 5 beers
top_5_beer_types <- head(df_bt,5)

# Plot the 5 top beer types in the dataset

ggplot(data = top_5_beer_types, aes(x=style, y=n)) +
    geom_bar(stat="identity")+
  theme(axis.text.x = element_text(angle = 45, hjust=1))+
  # truncates labels
  scale_x_discrete(label = function(x) stringr::str_trunc(x, 20)) + ggtitle("Top 5 Styles of Beer that
  # Adds a theme and adds a centered title
  theme(plot.title = element_text(hjust = 0.5))+
  # Adds detailed caption information alongside a theme
  labs(x = "Top 5 Beer Styles in Dataset", y = "Number of Beers")
```



- The top beer present in the dataset is American IPA followed by America Pale Ale. The American IPA is almost twice as popular as the America pale ale. The other beers are represented in fewer quantities within the dataset than the top two beers.

```
# Beer with the highest abv
abv <- beer_df[order(beer_df$abv,decreasing = TRUE),]
ibu <- beer_df[order(beer_df$ibu,decreasing = TRUE),]

max_abv <- head(abv,1)
min_abv <- tail(abv,1)

max_ibu <- head(ibu,1)
min_ibu <- tail(ibu,1)
max_ibu
```

```
##       abv ibu beer_id                 beer_name                          style
## 148 0.082 138     980 Bitter Bitch Imperial IPA American Double / Imperial IPA
##     brewery_id ounces
## 148        374     12
```

- The beer with the highest ABV is Lee Hill Series Vol. 5 - Belgian Style Quadrupel Ale with an ABV of 12.8 %, whilst the beer with the lowest ABI is Oâ€™Malleyâ€™s Irish Style Cream Ale with an ABV of 0.1 %.
- The beer with the highest IBU is Bitter Bitch Imperial IPA with an ABI of 138, whilst the beer with

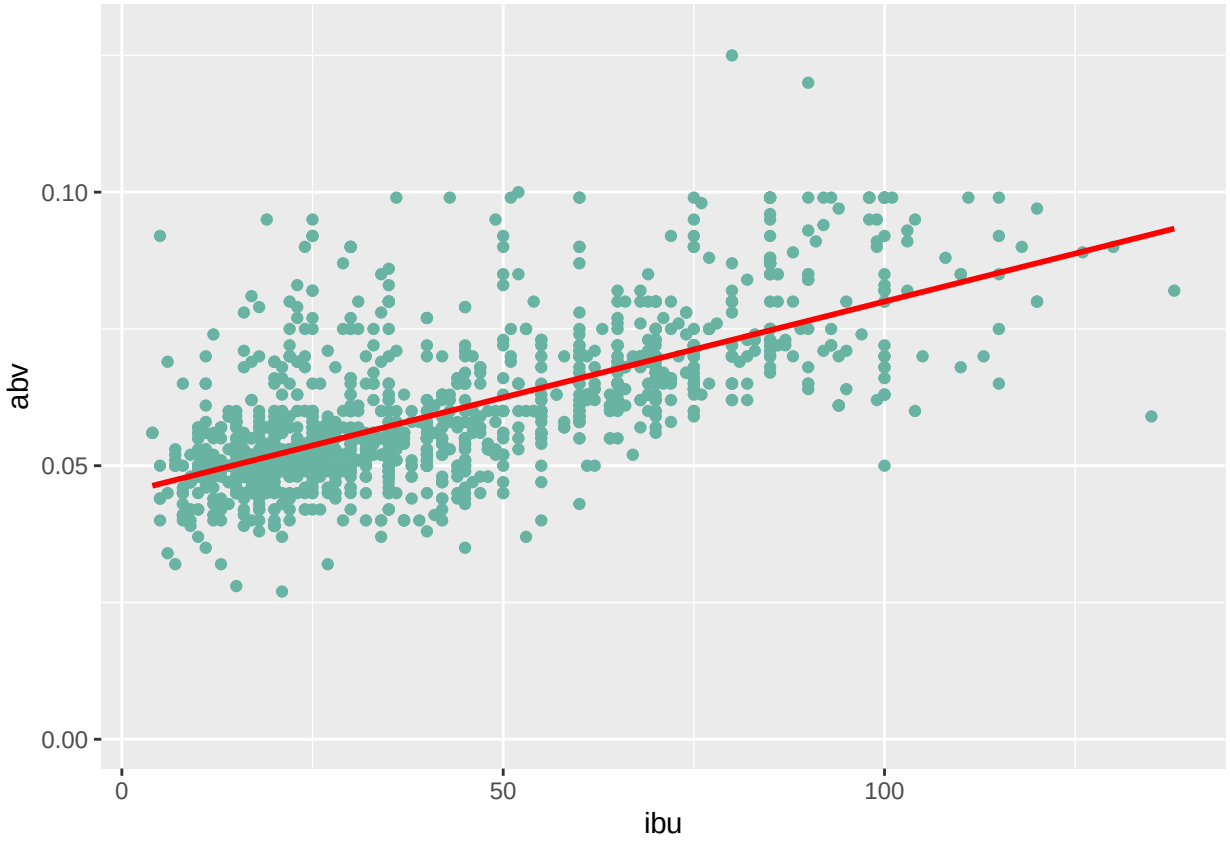
the lowest ABI is Rail Yard Ale (2009) with an ABI of 4.

```
# graph of ABV vs IBU
x <- beer_df$abv
y <- beer_df$ibu
ggplot(data = beer_df, aes(x=ibu, y=abv)) +
    geom_point( color="#69b3a2")+
  geom_smooth(method=lm , color="red", se=FALSE)
```

```
## `geom_smooth()` using formula 'y ~ x'
```

```
## Warning: Removed 1005 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 1005 rows containing missing values (geom_point).
```



Part 2: R Package

As part of this section, I have chosen to explore the equisse package. This is an R package that facilitates data manipulation for exploratory data analysis. It is one of the core packages of the tidyverse. The main goal of this package is to make it easier to manipulate data and allows the operators to easily extract, rearrange and manipulate to provide insights into the datasets in a user friendly way. The package also allows the use of pipes that allow the output of one command to become the input of another command. As part of this analysis, the previous datasets will utilized as well as the combined dataset of beers and breweries.

```
library(esquisse)
```

```
## Warning: package 'esquisse' was built under R version 4.1.2
```

```r
# getting the highest 5 average abv
highest_avg_abv <- beer_df %>%
                  group_by(style) %>%
                  summarize(Mean = mean(abv, na.rm=TRUE))
highest_average_abv <- highest_avg_abv[order(highest_avg_abv$Mean,decreasing = TRUE),]
high_5_abv <- head(highest_average_abv,5)
```

```r
#install.packages("DataExplorer")
#create_report(high_5_abv)
```

```r
# getting the highest 5 average ibu
highest_avg_ibu <- beer_df %>%
                  group_by(style) %>%
                  summarize(Mean = mean(ibu, na.rm=TRUE))
highest_average_ibu <- highest_avg_ibu[order(highest_avg_ibu$Mean,decreasing = TRUE),]
head(highest_average_ibu,5)
```

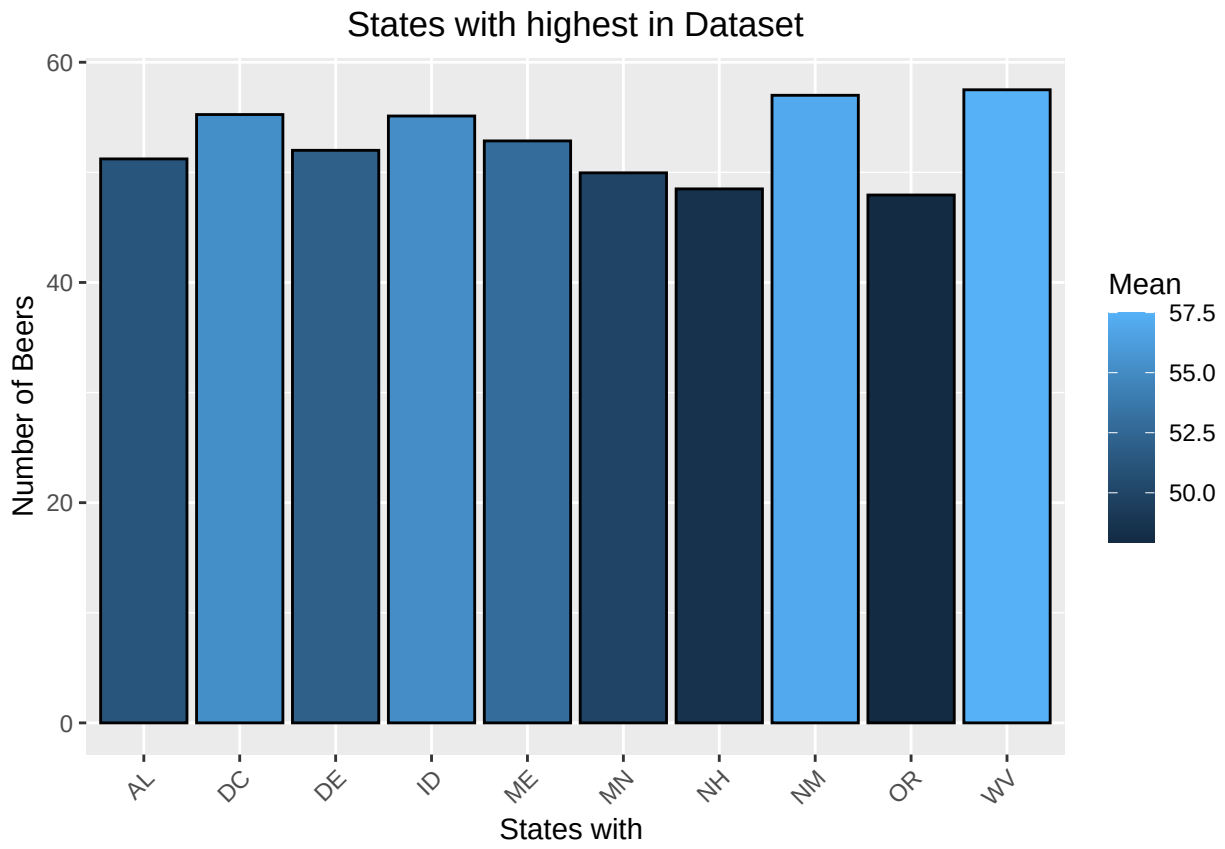
```
## # A tibble: 5 x 2
##   style                              Mean
##   <chr>                             <dbl>
## 1 American Barleywine                96
## 2 American Double / Imperial IPA     93.3
## 3 Russian Imperial Stout             86.5
## 4 American Double / Imperial Pilsner 85
## 5 Belgian Strong Dark Ale            72
```

```r
# getting the lowest 5 average ibu
min_ibu <- beer_df %>%
                  group_by(style) %>%
                  summarize(Mean = mean(ibu, na.rm=TRUE))

mimimum_ibu <- min_ibu[order(min_ibu$Mean,decreasing = FALSE),]
head(mimimum_ibu,5)
```

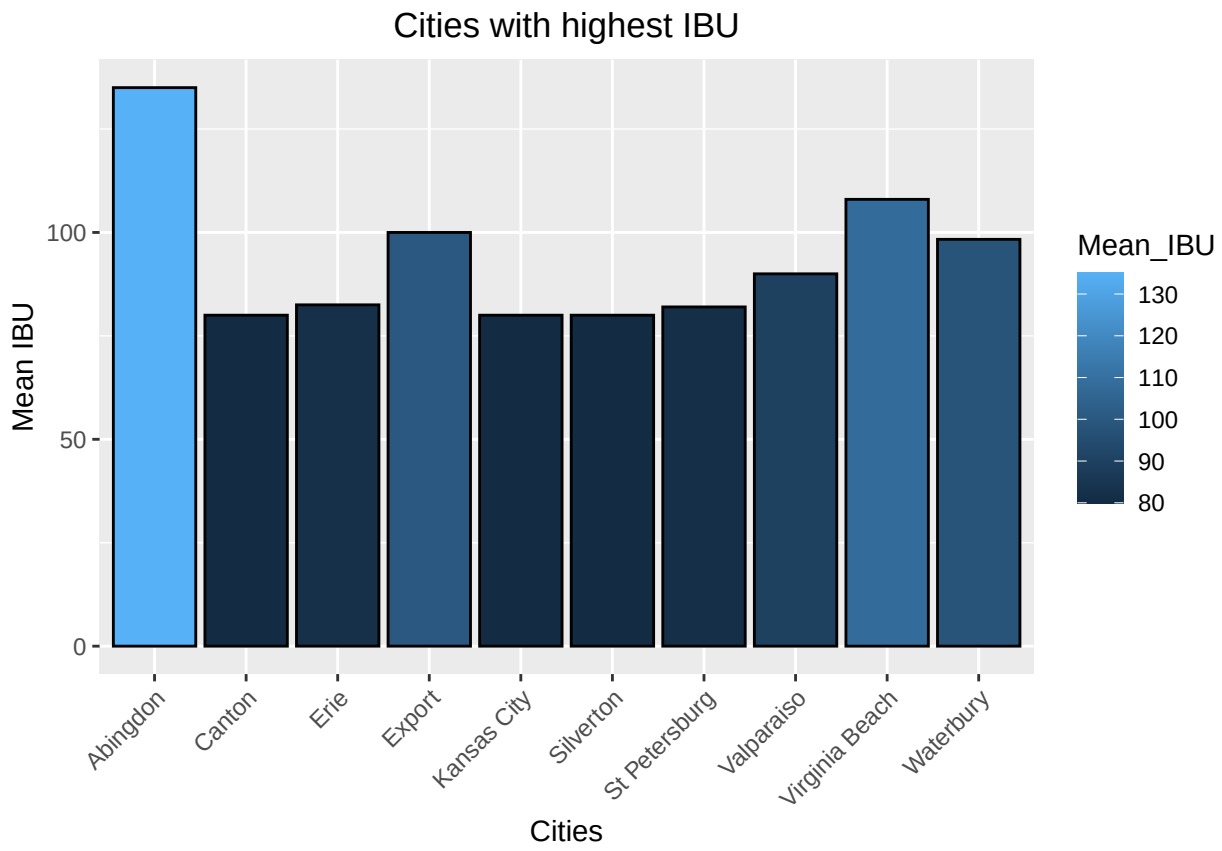
```
## # A tibble: 5 x 2
##   style                   Mean
##   <chr>                  <dbl>
## 1 Berliner Weissbier      7.8
## 2 Gose                    9.43
## 3 American Adjunct Lager 11
## 4 Light Lager            11.7
## 5 Fruit / Vegetable Beer 14.2
```

```r
# getting the state with the highest 10 average ibu
avg_state_ibu <- df %>%
                group_by(state)%>%
                summarize(Mean = mean(ibu, na.rm=TRUE))
average_state_ibu <- avg_state_ibu[order(avg_state_ibu$Mean,decreasing = TRUE),]
average_state_ibu <- head(average_state_ibu,10)


# Plot the states with the highest 10 average ibu in the dataset

ggplot(data = average_state_ibu, aes(x=state, y=Mean, fill=Mean)) +
    geom_bar(stat="identity", color="black", position=position_dodge())+
  theme(axis.text.x = element_text(angle = 45, hjust=1))+
```

```
  # truncates labels
  scale_x_discrete(label = function(x) stringr::str_trunc(x, 20)) + ggtitle("States with highest in Data
  # Adds a theme and adds a centered title
  theme(plot.title = element_text(hjust = 0.5))+
  # Adds detailed caption information alongside a theme
  labs(x = "States with ", y = "Number of Beers")
```




```
# Getting the cities with the highest 10 average ibu
avg_city_ibu <- df %>%
                group_by(city)%>%
                summarize(Mean_IBU= mean(ibu, na.rm=TRUE))
average_city_ibu <- avg_city_ibu[order(avg_city_ibu$Mean_IBU,decreasing = TRUE),]
average_cities_ibu <- head(average_city_ibu,10)

# Plotting the cities with the highest 10 average ibu
ggplot(data = average_cities_ibu, aes(x=city, y=Mean_IBU, fill=Mean_IBU)) +
    geom_bar(stat="identity", color="black", position=position_dodge())+
  theme(axis.text.x = element_text(angle = 45, hjust=1))+
  # truncates labels
  scale_x_discrete(label = function(x) stringr::str_trunc(x, 20)) + ggtitle("Cities with highest IBU") +
  # Adds a theme and adds a centered title
  theme(plot.title = element_text(hjust = 0.5))+
  # Adds detailed caption information alongside a theme
  labs(x = "Cities", y = "Mean IBU")
```



Part 3: Functions/Programming

The third part of the assignment

```
stateAnalysis <- function(state){

  # Creating a
  StateParameters <- list(name=state,
              beer_mean_ibu_state = mean(df[,"ibu"][df$state == state],na.rm=TRUE ),
              beer_mean_abv_state = mean(df[,"abv"][df$state == state], na.rm=TRUE),
              beer_max_ibu_state = max(df[,"ibu"][df$state == state],na.rm=TRUE ),
              beer_max_abv_state = max(df[,"abv"][df$state == state],na.rm=TRUE ),
              beer_min_ibu_state = min(df[,"ibu"][df$state == state],na.rm=TRUE ),
              beer_min_abv_state = min(df[,"abv"][df$state == state], na.rm=TRUE)
              )
  class(StateParameters) <- "state"
  return(StateParameters)
}
stateAnalysis("CO")
```

```
## $name
## [1] "CO"
##
## $beer_mean_ibu_state
## [1] 47.43151
##
```

```
## $beer_mean_abv_state
## [1] 0.063372
##
## $beer_max_ibu_state
## [1] 104
##
## $beer_max_abv_state
## [1] 0.128
##
## $beer_min_ibu_state
## [1] 9
##
## $beer_min_abv_state
## [1] 0.041
##
## attr(,"class")
## [1] "state"
```

```
summary.Region <- function(obj){
  cat("The mean values of region", obj$name, "are as follows:\n")
  cat("Mean Beer per capita: ", obj$beer_mean, "|", "stdev: ", obj$beer_sd, "\n")
  cat("Mean Wine per capita:", obj$wine_mean, "|", "stdev: ", obj$wine_sd, "\n")
  cat("Mean Spirit per capita:", obj$spirit_mean, "|", "stdev: ", obj$spirit_sd,  "\n")

}
```

```
cityAnalysis <- function(city){

  # Creating a
  CityParameters <- list(name=city,
              beer_mean_ibu_city = mean(df[,"ibu"][df$city == city],na.rm=TRUE),
              beer_mean_abv_city = mean(df[,"abv"][df$city == city], na.rm=TRUE),
              beer_max_ibu_city = max(df[,"ibu"][df$city == city],na.rm=TRUE ),
              beer_max_abv_city = max(df[,"abv"][df$city == city],na.rm=TRUE ),
              beer_min_ibu_city = min(df[,"ibu"][df$city == city],na.rm=TRUE ),
              beer_min_abv_city = min(df[,"abv"][df$city == city], na.rm=TRUE)
              )
  class(CityParameters) <- "city"
  return(CityParameters)
}

cityAnalysis("Louisville")
```

```
## $name
## [1] "Louisville"
##
## $beer_mean_ibu_city
## [1] 40.71429
##
## $beer_mean_abv_city
## [1] 0.0646
##
## $beer_max_ibu_city
## [1] 80
##
```

```
## $beer_max_abv_city
## [1] 0.125
##
## $beer_min_ibu_city
## [1] 13
##
## $beer_min_abv_city
## [1] 0.04
##
## attr(,"class")
## [1] "city"
```

```
BeerAnalysis <- function(style){

  # Creating a
  BeerParameters <- list(name=style,
              mean_ibu_for_beer_style = mean(df[,"ibu"][df$style == style],na.rm=TRUE),
              mean_abv_style_for_beer_style = mean(df[,"abv"][df$style == style],na.rm=TRUE)

              )
  class(BeerParameters) <- "beer"
  return(BeerParameters)
}

BeerAnalysis("American Pale Lager")
```

```
## $name
## [1] "American Pale Lager"
##
## $mean_ibu_for_beer_style
## [1] 26.75
##
## $mean_abv_style_for_beer_style
## [1] 0.05121622
##
## attr(,"class")
## [1] "beer"
```