Data Wrangling and Husbandry Final Project Report

29th April 2020

Data Wrangling and Husbandry (Stat 597)

Final Project Report

By: Aayush Mandhyan (am2447)

Abstract

To explore and mine hidden patterns within beer datasets (brewery, beer, beer review) and answer questions which come into mind when one thinks about a variety of beers.

Introduction

Covid-19 The first dataset which comes into one's mind in today's time, is very important but we are already looking at its analysis on the news like 10 times a day. Which made me think in the opposite direction and seek datasets which can be challenging as well as fun at the same time to display my skills gained in R.

It reminded me of last year's spring break, me and my friends took a road trip South. Towards Virginia, North Carolina to enjoy the best nature had to offer i.e. "The Smoky Mountains". And taste different varieties of beer we could come across. During this trip we would stop to seek out famous local breweries and try different varieties of beer they had to offer. The process would always be to seek out breweries, in the area we were in, via the internet or through recommendation by locals; and learn about their types of beers, ratings and so on before visiting the place out. Now thinking back to those times I wanted to take a data Scientist approach to the same questions and seek answers using the below data sources.

Data Source

- https://www.kaggle.com/nickhould/craft-cans#beers.csv (https://www.kaggle.com/nickhould/craft-cans#beers.csv) (beers, breweries)
- https://data.world/socialmediadata/beeradvocate (https://data.world/socialmediadata/beeradvocate) (beer_ratings)

Data Preprocessing

Loading the required packages

Loading the dataset

```
reviews <- read.csv("D:/Classes/Spring 2020/Stat597/Final Project/dataset/beer_reviews.cs
  v")
beers <- read.csv("D:/Classes/Spring 2020/Stat597/Final Project/dataset/beers.csv")
breweries <- read.csv("D:/Classes/Spring 2020/Stat597/Final Project/dataset/breweries.cs
  v")
```

Let’s take a look at the column information of the dataset:

Beer dataset

- abv: The alcoholic content by volume with 0 being no alcohol and 1 being pure alcohol
- ibu: International bittering units, which describe how bitter a drink is
- name: The name of the beer
- style: Beer style (lager, ale, IPA, etc.)
- brewery_id: Unique identifier for brewery that produces this beer
- ounces: Size of beer in ounces

Breweries dataset

- brewery_id: Unique identifier for brewery that produces this beer
- name: Name of the brewery
- city: City that the brewery is located in
- state: State that the brewery is located in

Review dataset

- brewery_id: Unique identifier for brewery that produces this beer
- brewery_name: Name of the brewery
- review_time: Timestamp of the review
- review_overall: Aggregated review score
- review_aroma: Score for beer aroma
- review_appearance: Score for beer appearance
- review_profilename: Reviewer Profile Name
- beer_style: Beer style (lager, ale, IPA, etc.)
- review_palate Score for beer palate
- review_taste: Score for beer taste
- beer_name: The name of the beer
- beer_abv: The alcoholic content by volume with 0 being no alcohol and 1 being pure alcohol
- beer_beerid: Unique identifier for the beer

First 2 dataset are from a single source and have same brewery_id, where as review dataset is sources from a different source hence has a different id for brewries. We need a normalized dataset consisting of all the meta data for every beer type available, for further exploratory data analysis.

Normalizing the dataset

To create a Normalized data set, first we take beer and brewery dataset into consideration as they come from a single source and have same id representation of breweries. So we inner join the first two dataset on brewery_id, as shown below:

```
beers_breweries <- beers %>% select(-X, -id) %>% inner_join(breweries, by = c('brewery_i
  d' = 'X'))
beers_breweries <- beers_breweries %>% rename(beer = name.x, brewery = name.y)
head(beers_breweries)
```

```
##     abv ibu                beer                          style brewery_id
## 1 0.050  NA            Pub Beer            American Pale Lager        408
## 2 0.066  NA         Devil's Cup        American Pale Ale (APA)        177
## 3 0.071  NA Rise of the Phoenix                   American IPA        177
## 4 0.090  NA            Sinister American Double / Imperial IPA        177
## 5 0.075  NA       Sex and Candy                   American IPA        177
## 6 0.077  NA        Black Exodus                  Oatmeal Stout        177
##   ounces                   brewery city state
## 1     12 10 Barrel Brewing Company Bend    OR
## 2     12       18th Street Brewery Gary    IN
## 3     12       18th Street Brewery Gary    IN
## 4     12       18th Street Brewery Gary    IN
## 5     12       18th Street Brewery Gary    IN
## 6     12       18th Street Brewery Gary    IN
```

After the first two datasets have been merged we have to encorporate review dataset. This review dataset has a different source, thus it has different id representation for breweries. So in this case we can not use brewery_id to join these two datasets. But on looking closer at the data, we see that the brewery name remains the same which we can leverage to merge these two datasets. That’s what we do in the next step.

```
beer_df <- reviews %>% inner_join(beers_breweries, by = c('beer_name' = 'beer'))
```

```
## Warning: Column `beer_name`/`beer` joining factors with different levels,
## coercing to character vector
```

```
head(beer_df)
```

```
##   brewery_id.x            brewery_name review_time review_overall
## 1         1075 Caldera Brewing Company  1251327677            4.0
## 2         1075 Caldera Brewing Company  1250928902            2.5
## 3         1075 Caldera Brewing Company  1249866208            4.0
## 4         1075 Caldera Brewing Company  1310259440            4.5
## 5         1075 Caldera Brewing Company  1249847121            4.5
## 6         1075 Caldera Brewing Company  1249556277            4.5
##   review_aroma review_appearance review_profilename
## 1          3.5               3.5         NJpadreFan
## 2          3.0               3.5              vacax
## 3          3.5               4.0           d0ggnate
## 4          4.5               4.5          Cyberkedi
## 5          3.5               4.0         babyhobbes
## 6          3.5               4.0            mdagnew
##                beer_style review_palate review_taste         beer_name
## 1 American Pale Ale (APA)           4.0          4.0  Caldera Pale Ale
## 2 American Pale Ale (APA)           3.5          2.5  Caldera Pale Ale
## 3 American Pale Ale (APA)           4.0          3.5  Caldera Pale Ale
## 4         American Porter           4.5          5.0 Pilot Rock Porter
## 5 American Pale Ale (APA)           3.5          4.0  Caldera Pale Ale
## 6 American Pale Ale (APA)           4.0          4.0  Caldera Pale Ale
##   beer_abv beer_beerid   abv ibu                   style brewery_id.y
## 1      5.5       25414 0.056  55 American Pale Ale (APA)          155
## 2      5.5       25414 0.056  55 American Pale Ale (APA)          155
## 3      5.5       25414 0.056  55 American Pale Ale (APA)          155
## 4      5.8       10788 0.060  NA         American Porter          155
## 5      5.5       25414 0.056  55 American Pale Ale (APA)          155
## 6      5.5       25414 0.056  55 American Pale Ale (APA)          155
##   ounces                 brewery    city state
## 1     12 Caldera Brewing Company Ashland    OR
## 2     12 Caldera Brewing Company Ashland    OR
## 3     12 Caldera Brewing Company Ashland    OR
## 4     12 Caldera Brewing Company Ashland    OR
## 5     12 Caldera Brewing Company Ashland    OR
## 6     12 Caldera Brewing Company Ashland    OR
```

We can observe that in the resulting dataset above consists of duplicate and non-essential features. So, next step is to keep the ones which are relevant to the premise of our analysis.

Thus, we only keep the following columns:

- beer_name: The name of the beer
- style: Beer style (lager, ale, IPA, etc.)
- abv: The alcoholic content by volume with 0 being no alcohol and 1 being pure alcohol
- ounces: Size of beer in ounces
- brewery_name: Name of the brewery
- city: City that the brewery is located in
- state: State that the brewery is located in

- review_overall: Aggregated review score
- review_appearance: Score for beer appearance
- review_taste: Score for beer taste
- review_aroma: Score for beer aroma
- review_palate Score for beer palate

```
beer_df <- beer_df %>% select(beer_name, style, abv, ounces, brewery_name, city, state, r
  eview_overall, review_appearance, review_taste, review_aroma, review_palate)
head(beer_df)
```

```
##           beer_name                   style   abv ounces
## 1  Caldera Pale Ale American Pale Ale (APA) 0.056     12
## 2  Caldera Pale Ale American Pale Ale (APA) 0.056     12
## 3  Caldera Pale Ale American Pale Ale (APA) 0.056     12
## 4 Pilot Rock Porter         American Porter 0.060     12
## 5  Caldera Pale Ale American Pale Ale (APA) 0.056     12
## 6  Caldera Pale Ale American Pale Ale (APA) 0.056     12
##              brewery_name    city state review_overall review_appearance
## 1 Caldera Brewing Company Ashland    OR            4.0               3.5
## 2 Caldera Brewing Company Ashland    OR            2.5               3.5
## 3 Caldera Brewing Company Ashland    OR            4.0               4.0
## 4 Caldera Brewing Company Ashland    OR            4.5               4.5
## 5 Caldera Brewing Company Ashland    OR            4.5               4.0
## 6 Caldera Brewing Company Ashland    OR            4.5               4.0
##   review_taste review_aroma review_palate
## 1          4.0          3.5           4.0
## 2          2.5          3.0           3.5
## 3          3.5          3.5           4.0
## 4          5.0          4.5           4.5
## 5          4.0          3.5           3.5
## 6          4.0          3.5           4.0
```

We, now have our final dataset ready to be cleaned.

Cleaning the dataset

Before we start analyzing our dataset, we need to make sure that the data set is void of any impurity i.e. NaN values.

```
map_df(beer_df, ~ sum(is.na(.)))
```

```
## # A tibble: 1 x 12
##   beer_name style   abv ounces brewery_name  city state review_overall
##       <int> <int> <int>  <int>        <int> <int> <int>          <int>
## 1         0     0   377      0            0     0     0              0
## # ... with 4 more variables: review_appearance <int>, review_taste <int>,
## #   review_aroma <int>, review_palate <int>
```

```
nrow(beer_df)
```

```
## [1] 99324
```

```
beer_df <- beer_df %>% drop_na()
```

We observe that out of 98947 rows we have NaN’s in 377 rows of abv column only. This we can clean up by just removing these rows without having a significant impact on our dataset.

Exploratory Data Analysis

Till, now we worked on merging our dataset, cleaning it for the purpose of analyzing it and finding actionable and valuable insights out of it. In this section we will be doing just that, and will try to answer some of the pertaining question which pop into ones mind when the think about beer.

Review Summary

First off let us take a look at various statistics of the review scores. These gives us insights into the distribution of scores for each criteria for a beer such:

- Appearance: How does the beer look, its color, its thickness, etc> This characteristics is the first thing a person observes about a beer.
- Aroma: The aroma of the beer. After appearance people smell the beer once they have openned the bottle. If its aroma is bad no one will consume it.
- Taste: Its the most important thing about a beer. Once people have gotten over the first two characteristics of the beer they want to taste it.
- Palate: This characteristic includes taste, aroma and over all texture of a beer. Which defines it in overall sense.
- Overall: Average score of the above individual characteristics.

```
cat('Appearance Score:\n')
```

```
## Appearance Score:
```

```
summary(beer_df$review_appearance)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   1.000   3.500   4.000   3.788   4.000   5.000
```

```
cat('\nTaste Score: \n')
```

```
## 
## Taste Score:
```

```
summary(beer_df$review_taste)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   1.000   3.500   4.000   3.746   4.000   5.000
```

```
cat('\nAroma Score: \n')
```

```
## 
## Aroma Score:
```

```
summary(beer_df$review_aroma)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   1.000   3.500   3.500   3.666   4.000   5.000
```

```
cat('\nPalate Score: \n')
```

```
## 
## Palate Score:
```

```
summary(beer_df$review_palate)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   1.000   3.500   4.000   3.699   4.000   5.000
```

```
cat('\nOverall Score: \n')
```

```
##
## Overall Score:
```

```
summary(beer_df$review_overall)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00    3.50    4.00    3.85    4.50    5.00
```

We observe that average ratings of our beers are quite high in all cateorgies, with a mean value between 3.67 - 3.85. Which is safe to say that the beers in our data set are quite good.

Find top 10 beer based on appearance score

Lets take a look at top 10 beers based on their appearance, also find the brewery which produces them and the brewery's location:

```
top_10_beers <- beer_df %>% group_by(beer_name) %>% summarise(avg=mean(review_appearanc
  e)) %>% arrange(desc(avg)) %>% head(10)

df_appearance <- data.frame(Beer = character(), Score = numeric(), Brewery_Name = charact
  er(), City = character(), State = character())
for (beer in top_10_beers$beer_name)
{
  brewery <- beer_df %>% filter(beer_name == beer) %>% head(1)
  score <- top_10_beers %>% filter(beer_name == beer) %>% head(1)
  df_appearance <- df_appearance %>% add_row(Beer = beer, Score = score$avg, Brewery_Name
  = brewery$brewery_name, City = brewery$city, State = brewery$state)
}
df_appearance
```

```
##                         Beer    Score                         Brewery_Name
## 1          Quakertown Stout 5.000000                  Armadillo Ale Works
## 2  Becky's Black Cat Porter 4.625000                 Seven Brides Brewing
## 3                    Car 21 4.500000            Pateros Creek Brewing Co.
## 4              Flagship Ale 4.500000    Grey Sail Brewing of Rhode Island
## 5              Midnight Oil 4.500000   John Harvard's Brewery & Ale House
## 6       Rusty Nail Pale Ale 4.500000 Huske Hardware House Brewing Company
## 7          Steel Wheels ESB 4.500000                Blue Mountain Brewery
## 8                Undertaker 4.500000          Wincle Beer Company Limited
## 9                 Beelzebub 4.473684          The Alchemist Pub & Brewery
## 10          Lost Sailor IPA 4.400000       Black Forest Brew Haus & Grill
##               City State
## 1           Denton    TX
## 2        Silverton    OR
## 3     Fort Collins    CO
## 4         Westerly    RI
## 5      Gainesville    FL
## 6      Garden City    ID
## 7            Afton    VA
## 8     Grand Rapids    MI
## 9        Waterbury    VT
## 10 South Deerfield    MA
```

Find top 10 beer based on aroma score

Lets take a look at top 10 beers based on their aroma, also find the brewery which produces them and the brewery’s location:

```
top_10_beers <- beer_df %>% group_by(beer_name) %>% summarise(avg=mean(review_aroma)) %>%
  arrange(desc(avg)) %>% head(10)

df_aroma <- data.frame(Beer = character(), Score = numeric(), Brewery_Name = character(),
  City = character(), State = character())
for (beer in top_10_beers$beer_name)
{
  brewery <- beer_df %>% filter(beer_name == beer) %>% head(1)
  score <- top_10_beers %>% filter(beer_name == beer) %>% head(1)
  df_aroma <- df_aroma %>% add_row(Beer = beer, Score = score$avg, Brewery_Name = brewery
  $brewery_name, City = brewery$city, State = brewery$state)
}
df_aroma
```

```
##                       Beer    Score                         Brewery_Name
## 1            Heady Topper 4.660981                        The Alchemist
## 2             The Crusher 4.625000           The Alchemist Pub & Brewery
## 3                    Jade 4.605263            Foothills Brewing Company
## 4            Abrasive Ale 4.500888                Surly Brewing Company
## 5                 Abigale 4.500000                     Sixpoint Brewery
## 6           Descender IPA 4.500000 GoodLife Brewing Company & Bier Hall
## 7   Graham Cracker Porter 4.500000                      Denver Beer Co.
## 8  Reprise Centennial Red 4.500000                  4 Hands Brewing Co.
## 9            Sunset Amber 4.500000                Coast Brewing Company
## 10         Hoponius Union 4.454545                  Jack's Abby Brewing
##               City State
## 1        Waterbury    VT
## 2        Waterbury    VT
## 3             Gary    IN
## 4  Brooklyn Center    MN
## 5         Brooklyn    NY
## 6             Bend    OR
## 7           Denver    CO
## 8      Saint Louis    MO
## 9             Hilo    HI
## 10      Framingham    MA
```

Find top 10 beer based on taste score

Lets take a look at top 10 beers based on their taste, also find the brewery which produces them and the brewery's location:

```
top_10_beers <- beer_df %>% group_by(beer_name) %>% summarise(avg=mean(review_taste)) %>%
  arrange(desc(avg)) %>% head(10)


df_taste <- data.frame(Beer = character(), Score = numeric(), Brewery_Name = character(),
  City = character(), State = character())
for (beer in top_10_beers$beer_name)
{
  brewery <- beer_df %>% filter(beer_name == beer) %>% head(1)
  score <- top_10_beers %>% filter(beer_name == beer) %>% head(1)
  df_taste <- df_taste %>% add_row(Beer = beer, Score = score$avg, Brewery_Name = brewery
  $brewery_name, City = brewery$city, State = brewery$state)
}
df_taste
```

```
##                         Beer    Score                         Brewery_Name
## 1             Descender IPA 4.625000 GoodLife Brewing Company & Bier Hall
## 2               The Crusher 4.625000          The Alchemist Pub & Brewery
## 3              Heady Topper 4.608742                        The Alchemist
## 4              Focal Banger 4.600000          The Alchemist Pub & Brewery
## 5            Tumbleweed IPA 4.538462        Lewis & Clark Brewing Company
## 6                 Beelzebub 4.526316          The Alchemist Pub & Brewery
## 7              Bitter Bitch 4.500000              Portneuf Valley Brewing
## 8        Black Walnut Wheat 4.500000          Piney River Brewing Company
## 9  Bridal Veil Rye Pale Ale 4.500000                Telluride Brewing Co.
## 10                   Car 21 4.500000            Pateros Creek Brewing Co.
##            City State
## 1          Bend    OR
## 2     Waterbury    VT
## 3     Waterbury    VT
## 4     Waterbury    VT
## 5        Helena    MT
## 6     Waterbury    VT
## 7          Gary    IN
## 8       Bucryus    MO
## 9     Telluride    CO
## 10 Fort Collins    CO
```

Find top 10 beer based on palate score

Lets take a look at top 10 beers based on their palate, also find the brewery which produces them and the brewery’s location:

```
top_10_beers <- beer_df %>% group_by(beer_name) %>% summarise(avg=mean(review_palate)) %>% arrange(desc(avg)) %>% head(10)

df_palate <- data.frame(Beer = character(), Score = numeric(), Brewery_Name = character(), City = character(), State = character())
for (beer in top_10_beers$beer_name)
{
  brewery <- beer_df %>% filter(beer_name == beer) %>% head(1)
  score <- top_10_beers %>% filter(beer_name == beer) %>% head(1)
  df_palate <- df_palate %>% add_row(Beer = beer, Score = score$avg, Brewery_Name = brewery$brewery_name, City = brewery$city, State = brewery$state)
}
df_palate
```

```
##                        Beer    Score
## 1              Focal Banger 4.600000
## 2  Bridal Veil Rye Pale Ale 4.500000
## 3          Quakertown Stout 4.500000
## 4            Southern Cross 4.500000
## 5          Steel Wheels ESB 4.500000
## 6              Heady Topper 4.405117
## 7               Chuli Stout 4.375000
## 8       Dry Dock Hefeweizen 4.375000
## 9                 Beelzebub 4.342105
## 10                     Jade 4.342105
##                                         Brewery_Name      City State
## 1                        The Alchemist Pub & Brewery Waterbury    VT
## 2                              Telluride Brewing Co. Telluride    CO
## 3                                Armadillo Ale Works    Denton    TX
## 4                               Olde Hickory Brewery      Hilo    HI
## 5                              Blue Mountain Brewery     Afton    VA
## 6                                      The Alchemist Waterbury    VT
## 7  Denali Brewing Company / Twister Creek Restaurant Talkeetna    AK
## 8                               Dry Dock Brewing Co.    Aurora    CO
## 9                        The Alchemist Pub & Brewery Waterbury    VT
## 10                         Foothills Brewing Company      Gary    IN
```

Find top 10 beer based on overall score

Lets take a look at top 10 beers based on their overall score, also find the brewery which produces them and the brewery’s location:

```
top_10_beers <- beer_df %>% group_by(beer_name) %>% summarise(avg=mean(review_overall)) %>% arrange(desc(avg)) %>% head(10)


df_overall <- data.frame(Beer = character(), Score = numeric(), Brewery_Name = character(), City = character(), State = character())
for (beer in top_10_beers$beer_name)
{
  brewery <- beer_df %>% filter(beer_name == beer) %>% head(1)
  score <- top_10_beers %>% filter(beer_name == beer) %>% head(1)
  df_overall <- df_overall %>% add_row(Beer = beer, Score = score$avg, Brewery_Name = brewery$brewery_name, City = brewery$city, State = brewery$state)
}
df_overall
```

```
##                         Beer  Score
## 1              Tug Pale Ale 4.7500
## 2              Heady Topper 4.6258
## 3               The Crusher 4.6250
## 4            Bear Ass Brown 4.5000
## 5         Big River Pilsner 4.5000
## 6              Bitter Bitch 4.5000
## 7              Boston Lager 4.5000
## 8  Bridal Veil Rye Pale Ale 4.5000
## 9             Descender IPA 4.5000
## 10      Dry Dock Hefeweizen 4.5000
##                                                Brewery_Name        City
## 1  Marshall Wharf Brewing Company / Three Tides Restaurant     Belfast
## 2                                             The Alchemist   Waterbury
## 3                               The Alchemist Pub & Brewery   Waterbury
## 4                                         Silverton Brewery   Silverton
## 5                                      Florida Beer Company Chattanooga
## 6                                   Portneuf Valley Brewing        Gary
## 7                                    Shenandoah Brewing Co.      Boston
## 8                                     Telluride Brewing Co.   Telluride
## 9                      GoodLife Brewing Company & Bier Hall        Bend
## 10                                    Dry Dock Brewing Co.      Aurora
##    State
## 1     ME
## 2     VT
## 3     VT
## 4     CO
## 5     TN
## 6     IN
## 7     MA
## 8     CO
## 9     OR
## 10    CO
```

Now we see top 10 beers based on various characteristics. Let’s see if there is any beer which has occurrence in all of them.

```
df_intersect <- intersect(df_appearance$Beer, df_aroma$Beer) %>% intersect(df_taste$Beer)
  %>% intersect(df_palate$Beer) %>% intersect(df_overall$Beer)
df_intersect
```

```
## character(0)
```

We observe that none of the beer belong to all 5 top categories. While some of them overlap in most. Example “Bridal Veil Rye Pale Ale” is found in taste, palate and overall top beers.

We have seen beers which rank the highest in these categories, but that about the places which brew them. Do any brewery has a higher score for a particular characteristics. Lets take a look at that.

Find top 10 breweries based on their beer appearance score

Lets take a look at top 10 breweries based on their beer appearance score, also find thier location:

```
top_10 <- beer_df %>% group_by(brewery_name) %>% summarise(avg=mean(review_appearance)) %
  >% arrange(desc(avg)) %>% head(10)


df_appearance <- data.frame(Brewery_Name = character(), Score = numeric(), City = charact
  er(), State = character())
for (b in top_10$brewery_name)
{
  brewery <- beer_df %>% filter(brewery_name == b) %>% head(1)
  score <- top_10 %>% filter(brewery_name == b) %>% head(1)
  df_appearance <- df_appearance %>% add_row(Brewery_Name = brewery$brewery_name, Score =
  score$avg, City = brewery$city, State = brewery$state)
}
df_appearance
```

```
##                            Brewery_Name Score          City State
## 1                   Armadillo Ale Works  5.00        Denton    TX
## 2          Karl Strauss Brewing Company  5.00      Santa Fe    NM
## 3            Original Saratoga Brew Pub  5.00     Pottstown    PA
## 4        Boathouse Brewpub & Restaurant  4.75   Murphysboro    IL
## 5               Bad Bear Brewing Company  4.50   Saint Louis    MO
## 6      Big Buck Brewery & Steakhouse #3  4.50 San Francisco    CA
## 7                      Boston Breweries  4.50        Boston    MA
## 8                 Bray's Brewing Company  4.50       Chatham    NY
## 9                     Buller Brewing Co.  4.50 San Francisco    CA
## 10 Felstar Brewery & Felstead Vineyard  4.50     Lexington    KY
```

Find top 10 breweries based on their beer aroma score

Lets take a look at top 10 breweries based on their beer aroma score, also find thier location:

```
top_10 <- beer_df %>% group_by(brewery_name) %>% summarise(avg=mean(review_aroma)) %>% ar
  range(desc(avg)) %>% head(10)

df_aroma <- data.frame(Brewery_Name = character(), Score = numeric(), City = character(),
  State = character())
for (b in top_10$brewery_name)
{
  brewery <- beer_df %>% filter(brewery_name == b) %>% head(1)
  score <- top_10 %>% filter(brewery_name == b) %>% head(1)
  df_aroma <- df_appearance %>% add_row(Brewery_Name = brewery$brewery_name, Score = scor
  e$avg, City = brewery$city, State = brewery$state)
}
df_aroma
```

```
##                            Brewery_Name Score          City State
## 1                    Armadillo Ale Works  5.00        Denton    TX
## 2          Karl Strauss Brewing Company  5.00      Santa Fe    NM
## 3            Original Saratoga Brew Pub  5.00     Pottstown    PA
## 4        Boathouse Brewpub & Restaurant  4.75   Murphysboro    IL
## 5              Bad Bear Brewing Company  4.50   Saint Louis    MO
## 6      Big Buck Brewery & Steakhouse #3  4.50 San Francisco    CA
## 7                     Boston Breweries  4.50        Boston    MA
## 8               Bray's Brewing Company  4.50       Chatham    NY
## 9                   Buller Brewing Co.  4.50 San Francisco    CA
## 10 Felstar Brewery & Felstead Vineyard  4.50     Lexington    KY
## 11                         G.G. Brewers  4.50 San Francisco    CA
```

Find top 10 breweries based on their beer taste score

Lets take a look at top 10 breweries based on their beer taste also find thier location:

```
top_10 <- beer_df %>% group_by(brewery_name) %>% summarise(avg=mean(review_taste)) %>% ar
  range(desc(avg)) %>% head(10)

df_taste <- data.frame(Brewery_Name = character(), Score = numeric(), City = character(),
  State = character())
for (b in top_10$brewery_name)
{
  brewery <- beer_df %>% filter(brewery_name == b) %>% head(1)
  score <- top_10 %>% filter(brewery_name == b) %>% head(1)
  df_taste <- df_appearance %>% add_row(Brewery_Name = brewery$brewery_name, Score = scor
  e$avg, City = brewery$city, State = brewery$state)
}
df_taste
```

```
##                           Brewery_Name Score          City State
## 1                  Armadillo Ale Works  5.00        Denton    TX
## 2         Karl Strauss Brewing Company  5.00      Santa Fe    NM
## 3           Original Saratoga Brew Pub  5.00     Pottstown    PA
## 4       Boathouse Brewpub & Restaurant  4.75   Murphysboro    IL
## 5             Bad Bear Brewing Company  4.50   Saint Louis    MO
## 6     Big Buck Brewery & Steakhouse #3  4.50 San Francisco    CA
## 7                     Boston Breweries  4.50        Boston    MA
## 8               Bray's Brewing Company  4.50       Chatham    NY
## 9                   Buller Brewing Co.  4.50 San Francisco    CA
## 10 Felstar Brewery & Felstead Vineyard  4.50     Lexington    KY
## 11                    Boston Breweries  4.50        Boston    MA
```

Find top 10 breweries based on their beer palate score

Lets take a look at top 10 breweries based on their beer palate score, also find thier location:

```
top_10 <- beer_df %>% group_by(brewery_name) %>% summarise(avg=mean(review_palate)) %>% a
  rrange(desc(avg)) %>% head(10)


df_palate <- data.frame(Brewery_Name = character(), Score = numeric(), City = character
  (), State = character())
for (b in top_10$brewery_name)
{
  brewery <- beer_df %>% filter(brewery_name == b) %>% head(1)
  score <- top_10 %>% filter(brewery_name == b) %>% head(1)
  df_palate <- df_appearance %>% add_row(Brewery_Name = brewery$brewery_name, Score = sco
  re$avg, City = brewery$city, State = brewery$state)
}
df_palate
```

```
##                           Brewery_Name Score          City State
## 1                  Armadillo Ale Works  5.00        Denton    TX
## 2         Karl Strauss Brewing Company  5.00      Santa Fe    NM
## 3           Original Saratoga Brew Pub  5.00     Pottstown    PA
## 4       Boathouse Brewpub & Restaurant  4.75   Murphysboro    IL
## 5             Bad Bear Brewing Company  4.50   Saint Louis    MO
## 6     Big Buck Brewery & Steakhouse #3  4.50 San Francisco    CA
## 7                     Boston Breweries  4.50        Boston    MA
## 8               Bray's Brewing Company  4.50       Chatham    NY
## 9                   Buller Brewing Co.  4.50 San Francisco    CA
## 10 Felstar Brewery & Felstead Vineyard  4.50     Lexington    KY
## 11                            Hub City  4.50         Boise    ID
```

Find top 10 breweries based on their beer overall score

Lets take a look at top 10 breweries based on their beer overall score, also find thier location:

```
top_10 <- beer_df %>% group_by(brewery_name) %>% summarise(avg=mean(review_overall)) %>%
  arrange(desc(avg)) %>% head(10)


df_overall <- data.frame(Brewery_Name = character(), Score = numeric(), City = character
  (), State = character())
for (b in top_10$brewery_name)
{
  brewery <- beer_df %>% filter(brewery_name == b) %>% head(1)
  score <- top_10 %>% filter(brewery_name == b) %>% head(1)
  df_overall <- df_appearance %>% add_row(Brewery_Name = brewery$brewery_name, Score = sc
  ore$avg, City = brewery$city, State = brewery$state)
}
df_overall
```

```
##                           Brewery_Name Score          City State
## 1                   Armadillo Ale Works  5.00        Denton    TX
## 2          Karl Strauss Brewing Company  5.00      Santa Fe    NM
## 3            Original Saratoga Brew Pub  5.00     Pottstown    PA
## 4        Boathouse Brewpub & Restaurant  4.75   Murphysboro    IL
## 5              Bad Bear Brewing Company  4.50   Saint Louis    MO
## 6      Big Buck Brewery & Steakhouse #3  4.50 San Francisco    CA
## 7                     Boston Breweries  4.50        Boston    MA
## 8              Bray's Brewing Company  4.50       Chatham    NY
## 9                  Buller Brewing Co.  4.50 San Francisco    CA
## 10 Felstar Brewery & Felstead Vineyard  4.50     Lexington    KY
## 11                  Armadillo Ale Works  4.50        Denton    TX
```

Now we see top 10 beers based on various characteristics. Let’s see if there is any brewery which has occurrence in all of them.

```
df_intersect <- intersect(df_appearance$Beer, df_aroma$Beer) %>% intersect(df_taste$Beer)
  %>% intersect(df_palate$Beer) %>% intersect(df_overall$Beer)
df_intersect
```

```
## NULL
```

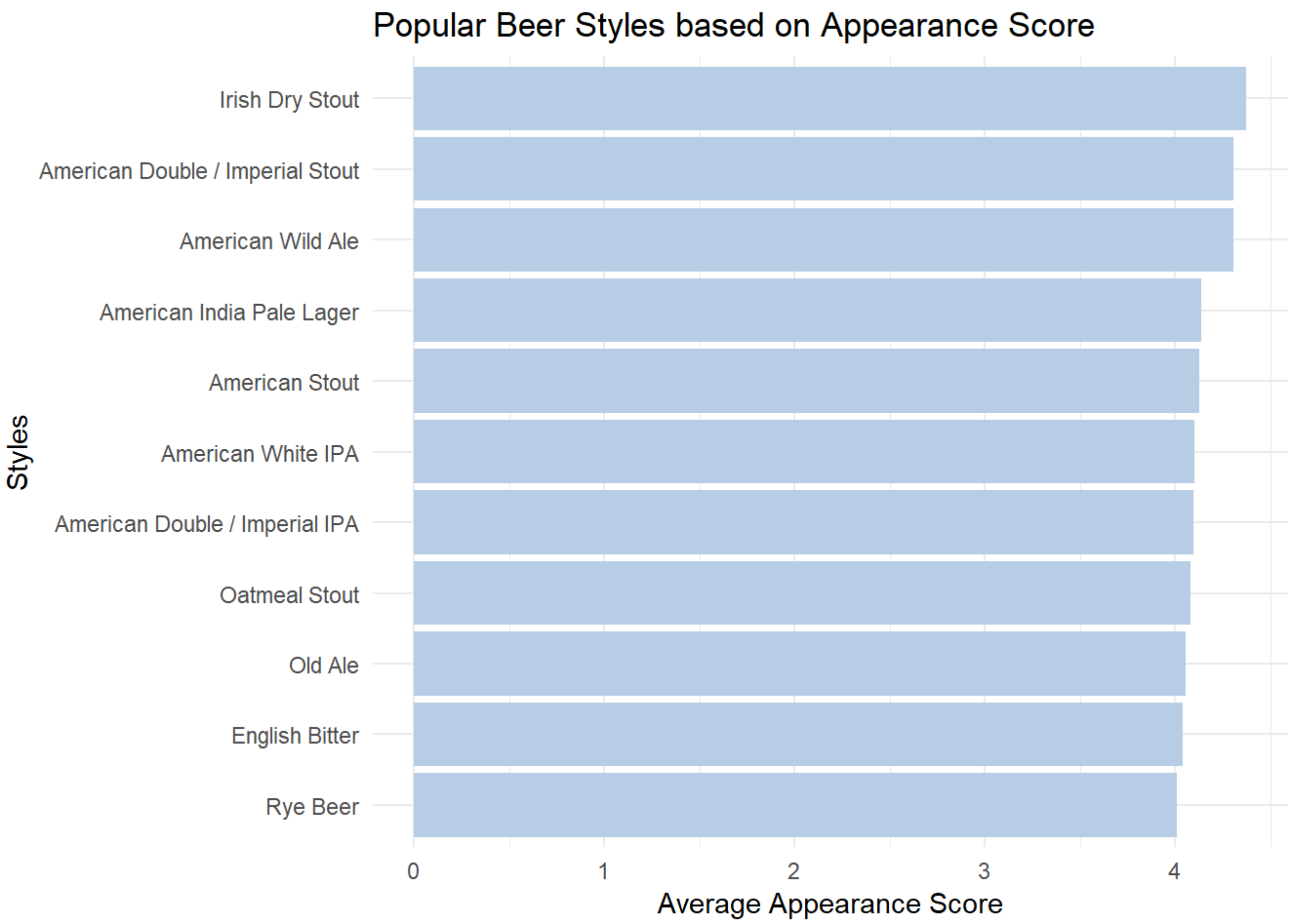
We observe that none of the brewery belong to all 5 top categories.

Next, lets take a look at various beer styles based on review scores. This will give us more insight into which style is better at which characteristics.

Top beer styles based on appearance score

Let’s take a look at the top styles based on their appearance score.

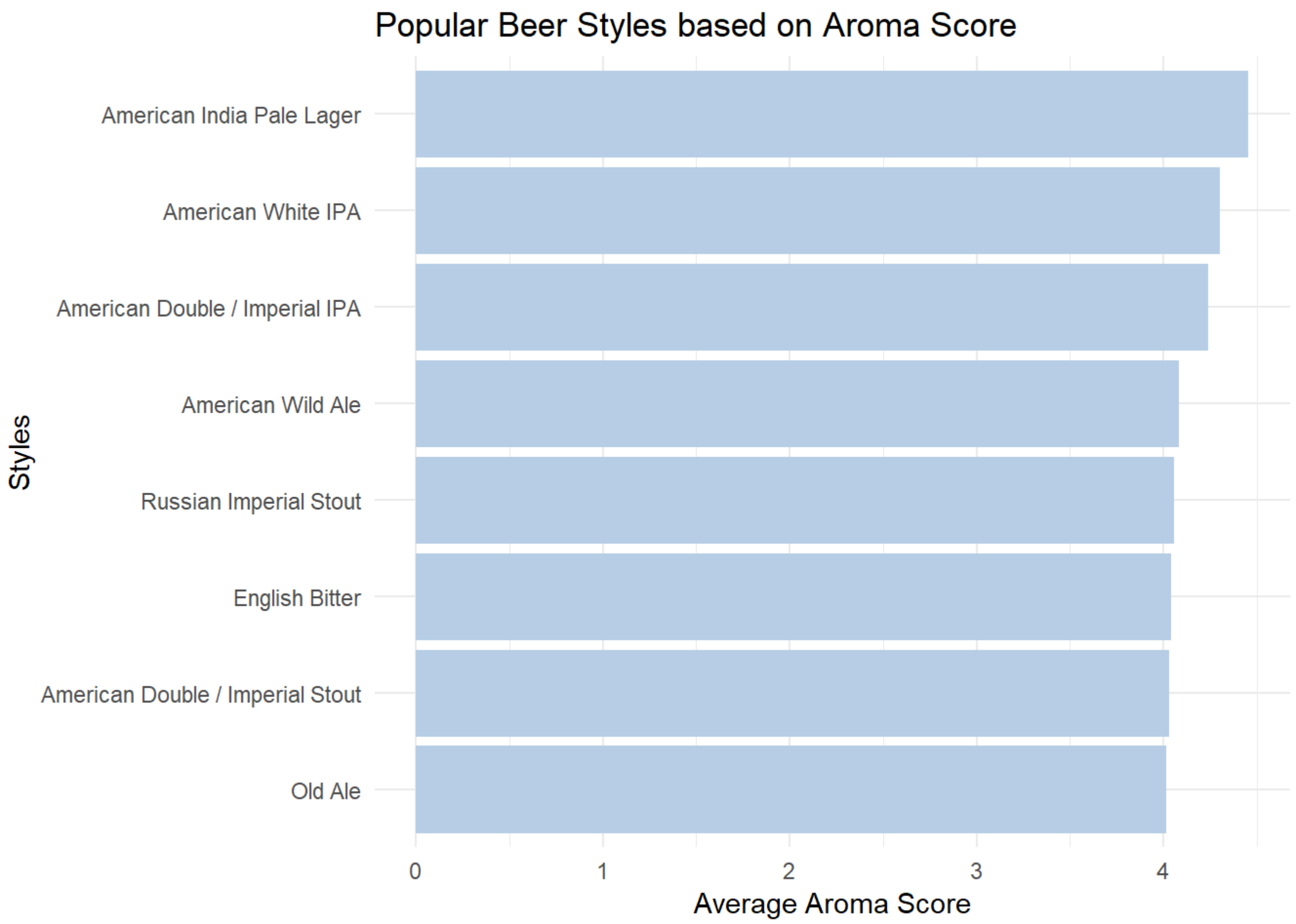
```r
beer_df %>%
  group_by(style) %>% # Group by the beer style
  summarise(avg=mean(review_appearance)) %>% # Count per group
  filter(avg > 4) %>% # Only the larger groups
  ggplot(aes(reorder(style, avg), avg)) + # Reorder the bars
  geom_col(fill = '#B6CDE5') +
  theme_minimal() +
  coord_flip() +
  ylab('Average Appearance Score') +
  xlab('Styles') +
  ggtitle('Popular Beer Styles based on Appearance Score')
```



Top beer styles based on aroma score

Let’s take a look at the top styles based on their aroma score.

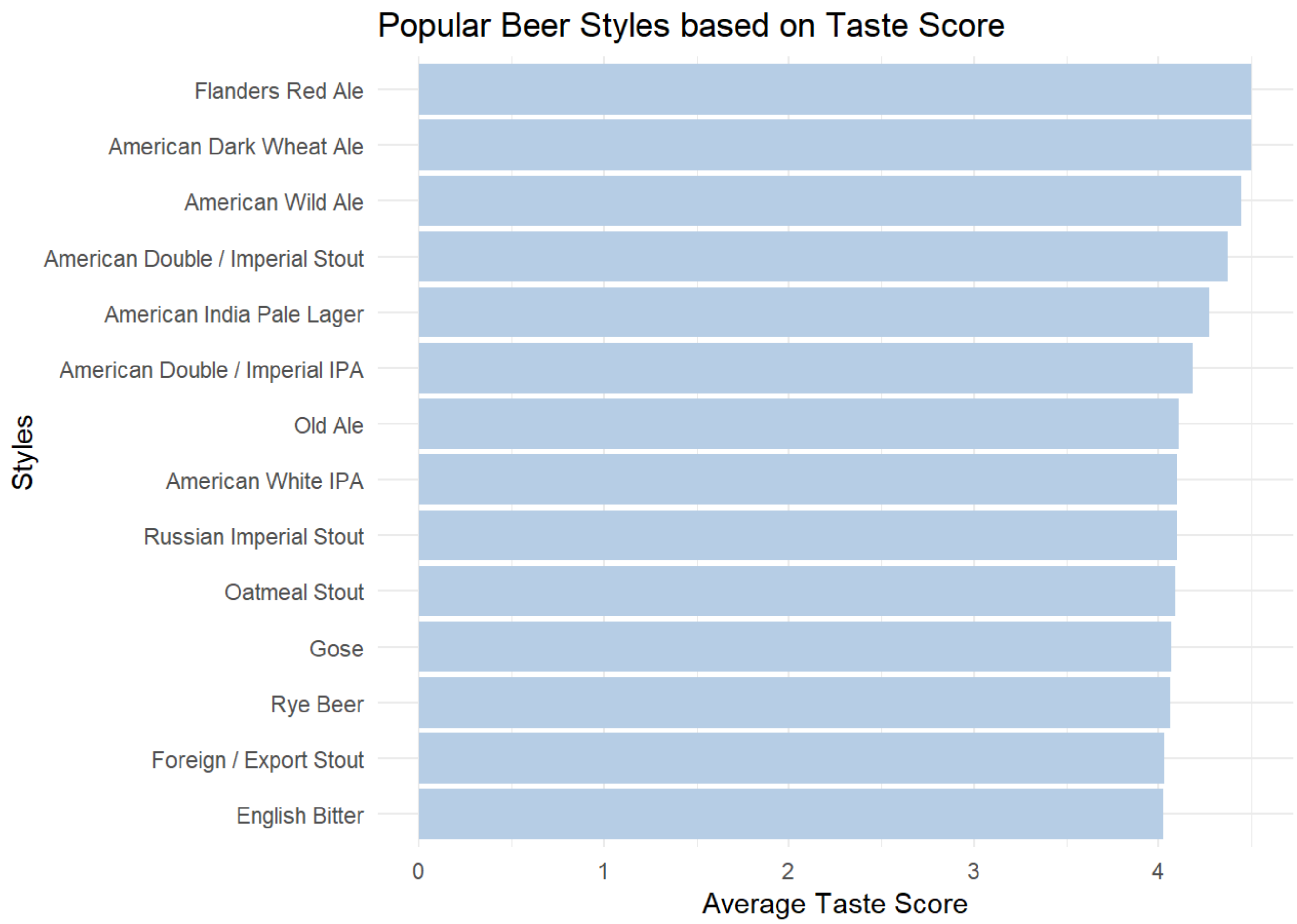
```
beer_df %>%
  group_by(style) %>% # Group by the beer style
  summarise(avg=mean(review_aroma)) %>% # Count per group
  filter(avg > 4) %>% # Only the larger groups
  ggplot(aes(reorder(style, avg), avg)) + # Reorder the bars
  geom_col(fill = '#B6CDE5') +
  theme_minimal() +
  coord_flip() +
  ylab('Average Aroma Score') +
  xlab('Styles') +
  ggtitle('Popular Beer Styles based on Aroma Score')
```



Top beer styles based on taste score

Let’s take a look at the top styles based on their taste score.

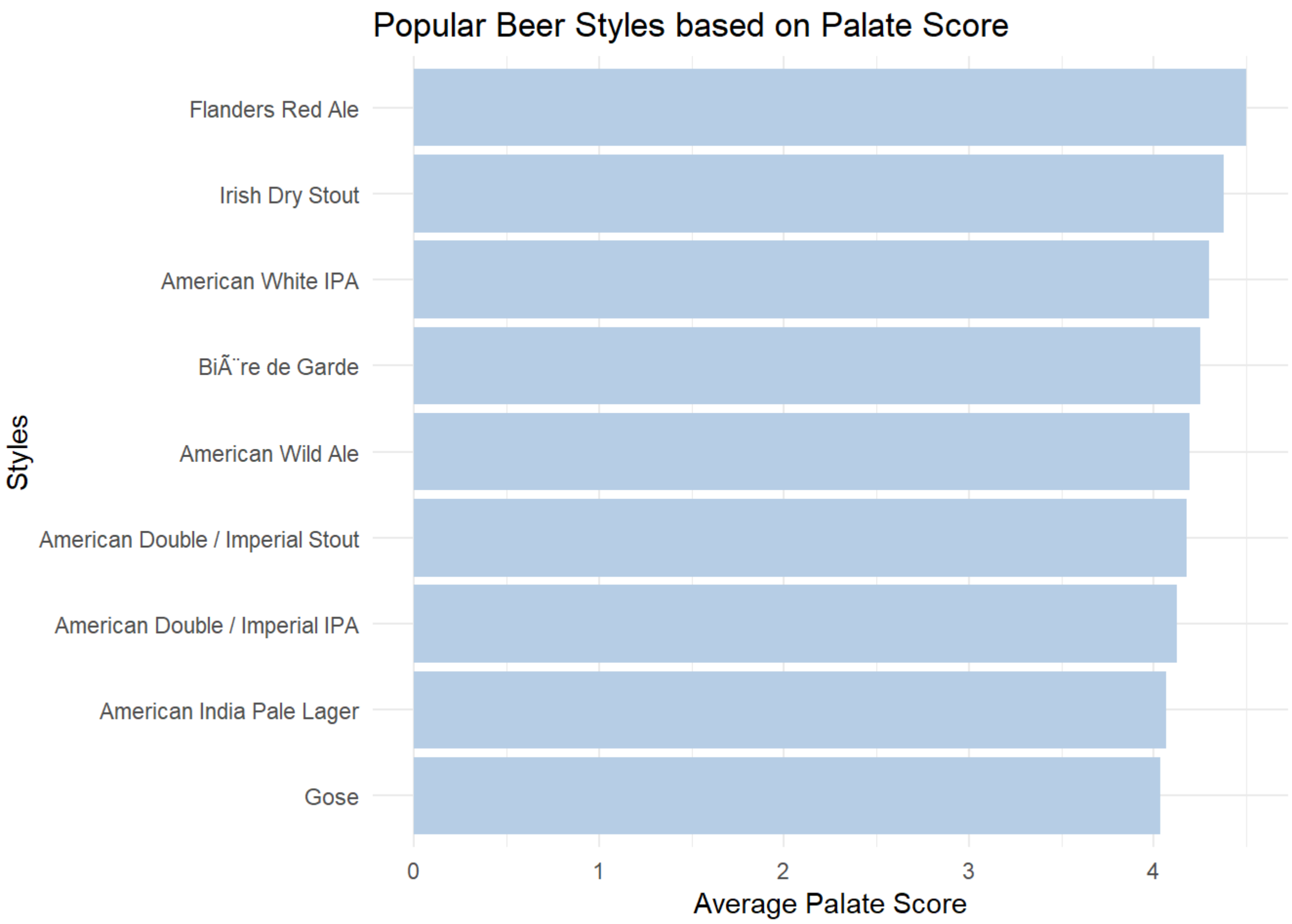
```
beer_df %>%
  group_by(style) %>% # Group by the beer style
  summarise(avg=mean(review_taste)) %>% # Count per group
  filter(avg > 4) %>% # Only the larger groups
  ggplot(aes(reorder(style, avg), avg)) + # Reorder the bars
  geom_col(fill = '#B6CDE5') +
  theme_minimal() +
  coord_flip() +
  ylab('Average Taste Score') +
  xlab('Styles') +
  ggtitle('Popular Beer Styles based on Taste Score')
```



Top beer styles based on palate score

Let’s take a look at the top styles based on their palate score.

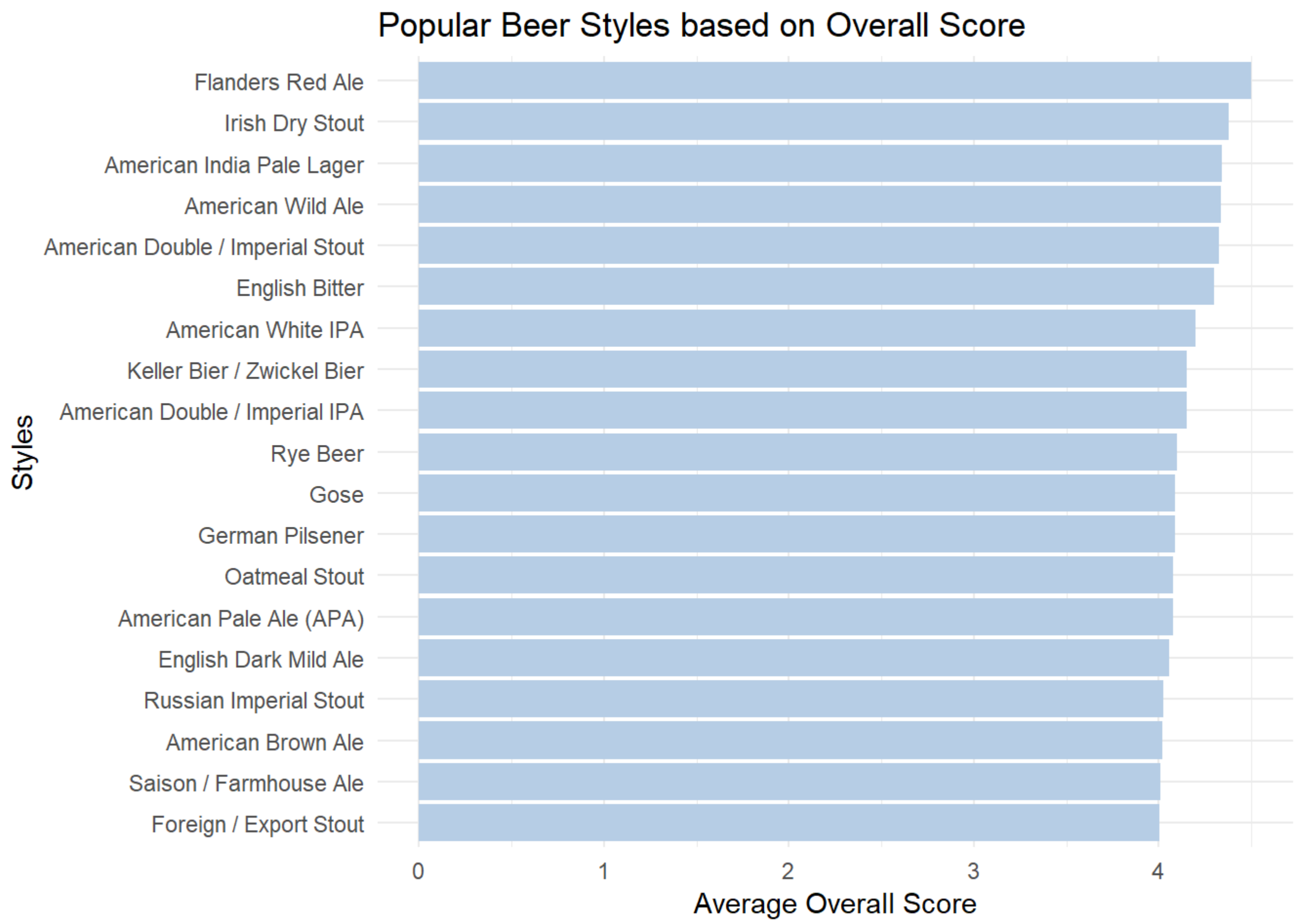
```
beer_df %>%
  group_by(style) %>% # Group by the beer style
  summarise(avg=mean(review_palate)) %>% # Count per group
  filter(avg > 4) %>% # Only the larger groups
  ggplot(aes(reorder(style, avg), avg)) + # Reorder the bars
  geom_col(fill = '#B6CDE5') +
  theme_minimal() +
  coord_flip() +
  ylab('Average Palate Score') +
  xlab('Styles') +
  ggtitle('Popular Beer Styles based on Palate Score')
```



Top beer styles based on overall score

Let’s take a look at the top styles based on their overall score.

```
beer_df %>%
  group_by(style) %>% # Group by the beer style
  summarise(avg=mean(review_overall)) %>% # Count per group
  filter(avg > 4) %>% # Only the larger groups
  ggplot(aes(reorder(style, avg), avg)) + # Reorder the bars
  geom_col(fill = '#B6CDE5') +
  theme_minimal() +
  coord_flip() +
  ylab('Average Overall Score') +
  xlab('Styles') +
  ggtitle('Popular Beer Styles based on Overall Score')
```



Brewery density across states in the United States

Let us take a look at the number of breweries in United States accross variou states. We see this plotted on the United States map.

```
brewery_density <- beer_df %>% group_by(region) %>% count(brewery_name) %>% summarise(val
  ue = n())
brewery_density[is.na(brewery_density)] <- 0
names(brewery_density) = c("region", "value")
plot_brewery_density <- state_choropleth(brewery_density, title  = "Breweries across Unit
  ed States", legend = "Count")
require(gridExtra)
```

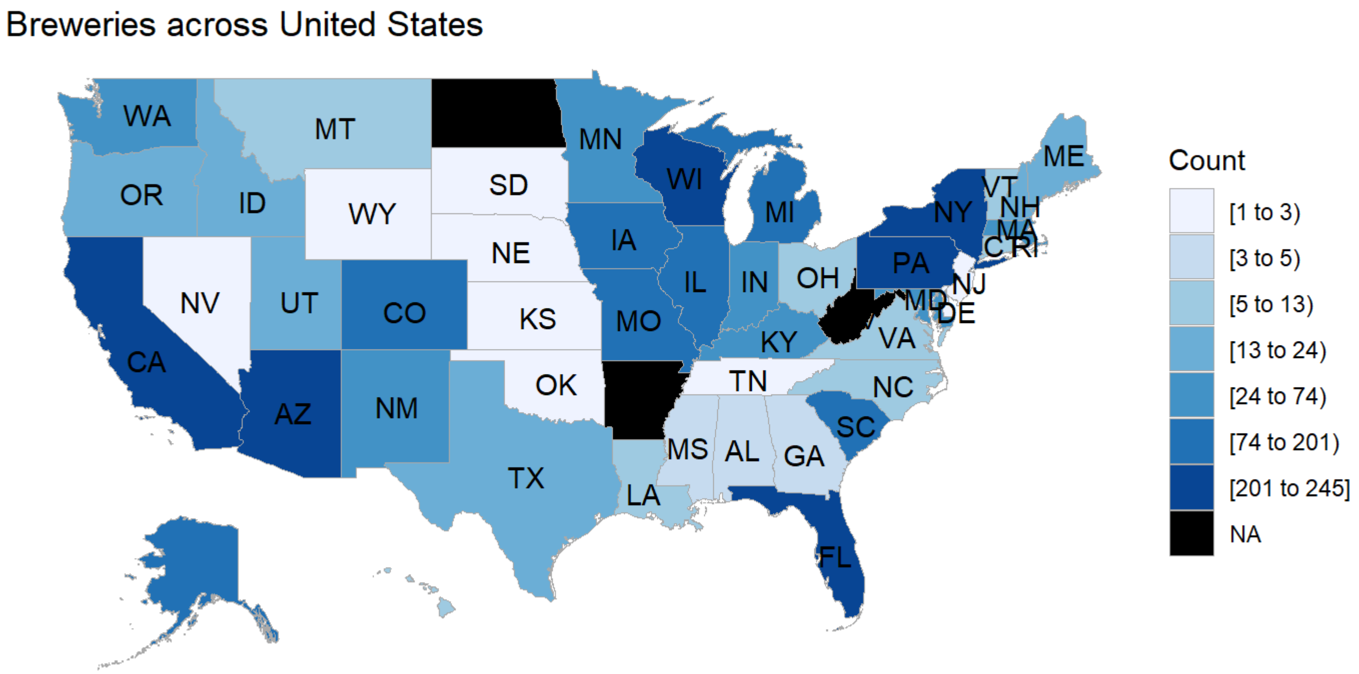
```
## Loading required package: gridExtra
```

```
##
## Attaching package: 'gridExtra'
```

```
## The following object is masked from 'package:acs':
##
##     combine
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
grid.arrange(plot_brewery_density, ncol=1)
```

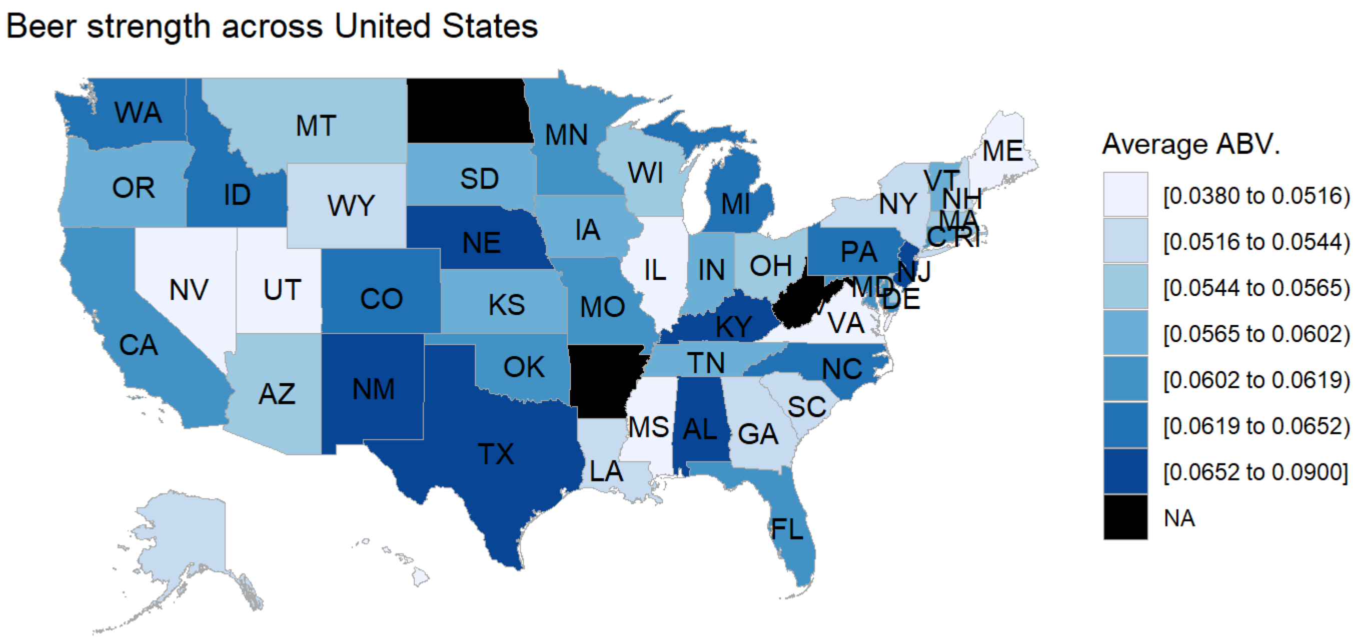


From the above map we see that there are higher number of breweries in California, Arizona, Florida, etc. Where as the number of breweries in central america are very low.

Average alcohol strength of beer by state

Next, we take a look at the strength of beers accross states.

```
df <- beer_df %>% group_by(region) %>% summarise(avg_= mean(abv))
df[is.na(df)] <- 0
names(df) = c("region", "value")
plot_df <- state_choropleth(df, title  = "Beer strength across United States", legend =
  "Average ABV.")
require(gridExtra)
grid.arrange(plot_df, ncol=1)
```

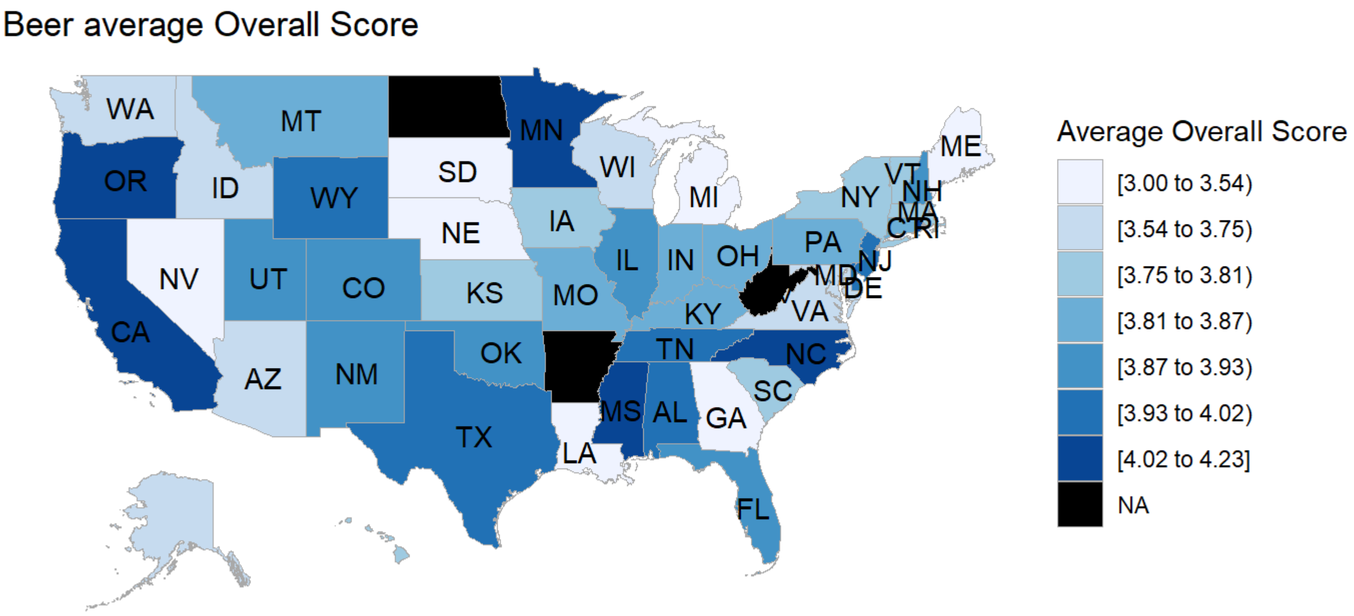


From this map we see that the alcohol content in the beer are higher in middle states of the US. Where as from the previous map we saw that the count of breweries were low in that region.

Average beer overall review of beer by state

Next, we take a look at the over all review scores of beers accross states.

```
df <- beer_df %>% group_by(region) %>% summarise(avg_= mean(review_overall))
df[is.na(df)] <- 0
names(df) = c("region", "value")
plot_df <- state_choropleth(df, title  = "Beer average Overall Score", legend = "Average
   Overall Score")
require(gridExtra)
grid.arrange(plot_df, ncol=1)
```

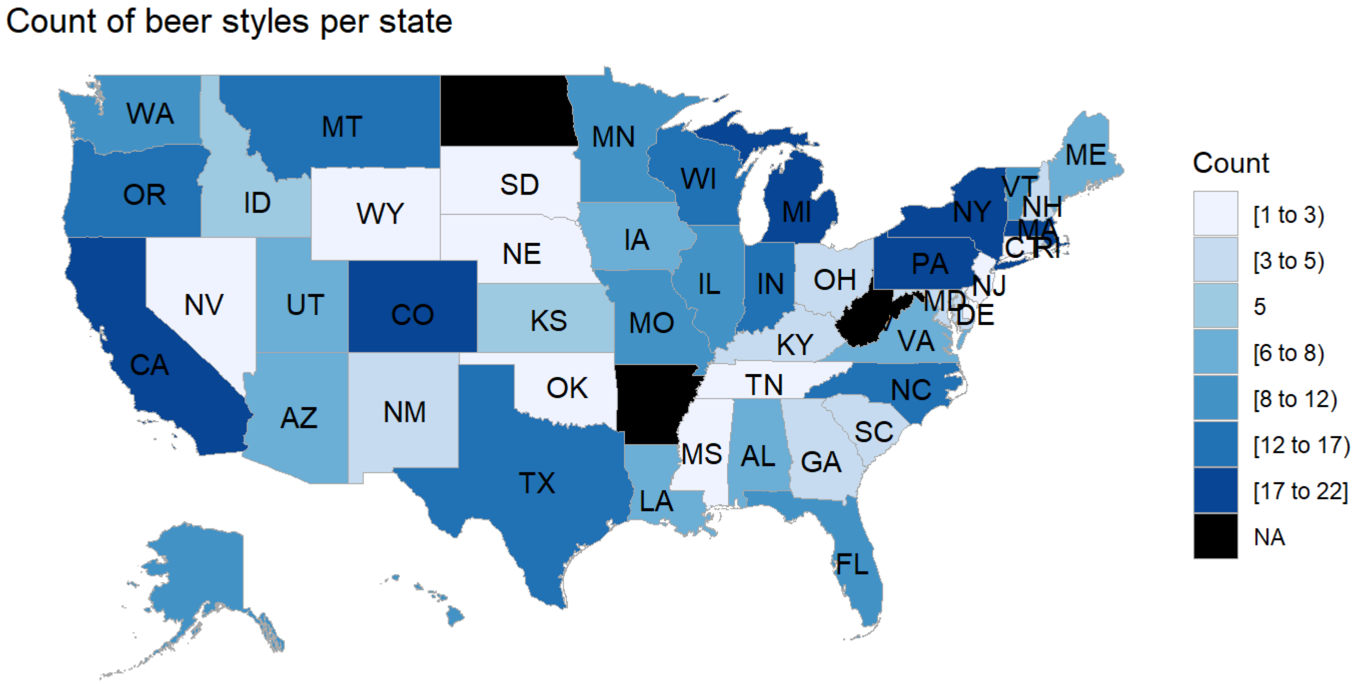


From the above map we observe that the states where there are more number of breweries have higher overall score for the beer. But we also see that the states with higher average abv for a beer also have a higher average overall score.

Number of styles per state

Finally we take a look at the number of different beer styles produced based on state in the US.

```
df <- beer_df
df$style <- as.character(df$style)
df <- within(df, {no.styles<-ave(style,region,FUN=function(x) length(unique(x)))})
df <- subset(df, select=c("region","no.styles"))
df <- unique(df)
df$no.styles <- as.numeric(paste(df$no.styles))
df[is.na(df)] <- 0
names(df) = c("region", "value")
plot_df <- state_choropleth(df, title  = "Count of beer styles per state", legend = "Coun
  t")
require(gridExtra)
grid.arrange(plot_df, ncol=1)
```



And we find that CA, CO, PA, NY, MI, MT, TX have a higher count of beer styles.

Number of styles per city

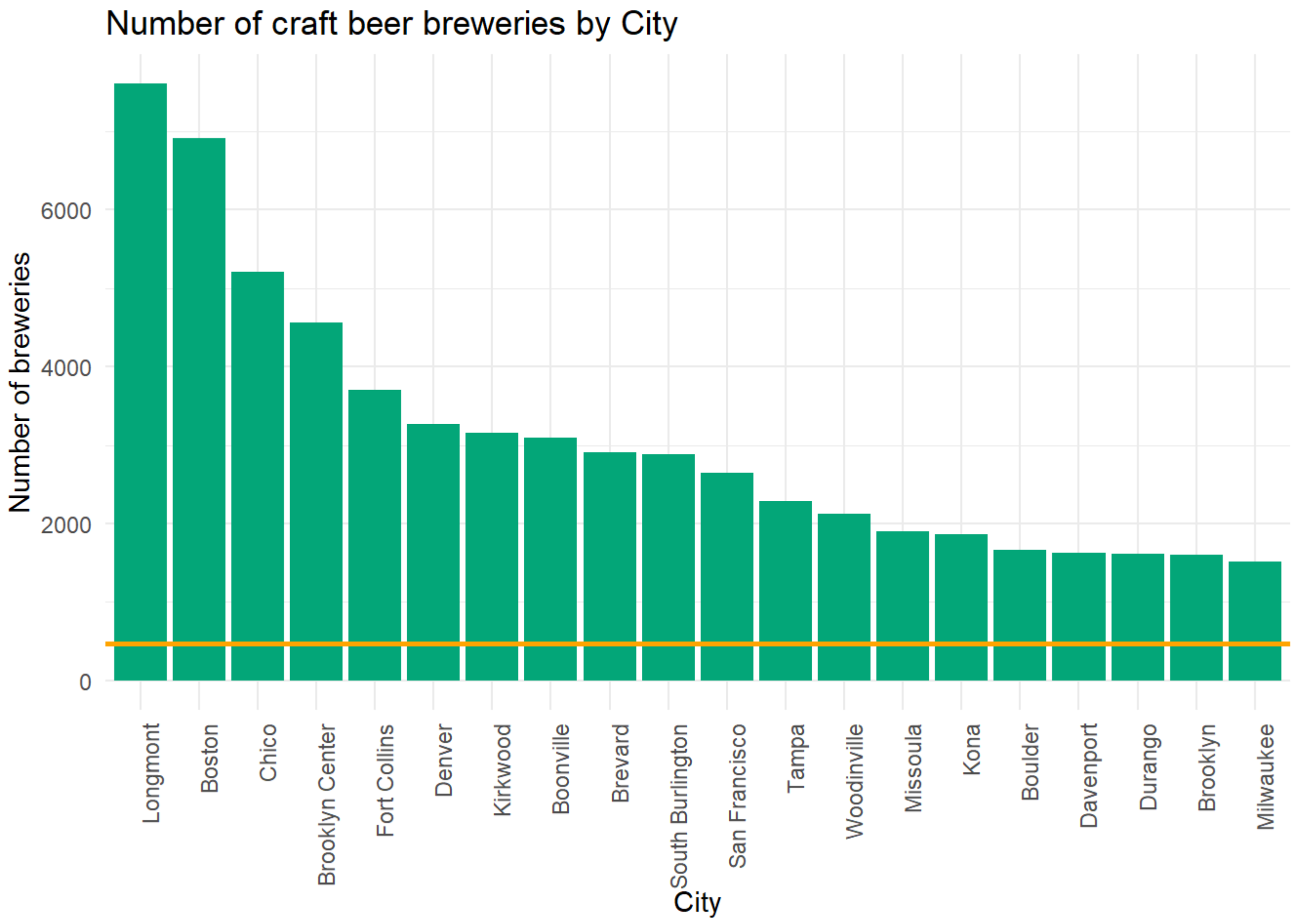
And at last, we take a look at the cities with the most number of beer styles.

```
cities_count <- beer_df %>%
  group_by(city) %>% # Group by city
  summarise(n_breweries = n()) %>% # Count per city
  arrange(desc(n_breweries)) %>% # Sort by count descending
  top_n(20) # Return the top 20 rows
```

```
## Selecting by n_breweries
```

```
# Calculate mean average number of breweries
avg_breweries_city <- beer_df %>%
  group_by(city) %>%
  summarise(n = n()) %>%
  summarise(mean = mean(n)) %>%
  `$`(mean)

ggplot(data = cities_count, aes(x = reorder(city, -n_breweries), y = n_breweries)) +
  geom_col( fill = '#03A678') +
  geom_hline(yintercept = avg_breweries_city, color = '#FFA400', size = 1) + # add horizontal line for average number
  theme_minimal() +
  labs(x = "City", y = "Number of breweries", title = "Number of craft beer breweries by City") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) # rotate the axis text
```



Conclusion

In this analysis, we took a deep dive into our beer data set and found some intresting insights. We got to explore various top beers based on their different characteristics review scores, we got to know which berwery produced it and in in which location. We got to know where are the top rated breweries located.

We got to see locations of various styles of beer, beer abv and beer overall score on the map of United States.

All this insights can be used by the buisness user as well as by an ordinary beer drinker. To seek different styles of beer based on different criteria. They got to know which states to look for a particular type of brew and also how much alcohol content they can expect in their beer based on the state they got the beer from.

Resource

You will find the data set and the .Rmd file for this project on my github repo (link (https://github.com/AayushMandhyan/Data-Wrangling-Beer-Dataset)).

References

- Professor Jason Klusowski's notes and class assignments.
- https://monashbioinformaticsplatform.github.io/2017-11-16-open-science-training/topics/rmarkdown.html (https://monashbioinformaticsplatform.github.io/2017-11-16-open-science-training/topics/rmarkdown.html)
- https://www.kaggle.com/nickhould/craft-cans (https://www.kaggle.com/nickhould/craft-cans)
- https://r4ds.had.co.nz/index.html (https://r4ds.had.co.nz/index.html)

Acknowledgement

I would like to thank Professor Jason Klusowski for his efforts he puts into the class as well as his guidance in learning R by doing hand-on assignments and also in incorporating those learning into this final project.