FLS\_8 Working

David

2023-06-17

Load Necessary Packages

# install.packages("naniar")  
library(dplyr)

## Warning: package 'dplyr' was built under R version 4.1.3

##   
## 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)

## Warning: package 'ggplot2' was built under R version 4.1.3

library(ggthemes)

## Warning: package 'ggthemes' was built under R version 4.1.3

library(naniar)

## Warning: package 'naniar' was built under R version 4.1.3

#install.packages("ggsn")  
#install.packages("usmap")  
library(class)  
library(caret)

## Warning: package 'caret' was built under R version 4.1.3

## Loading required package: lattice

library(e1071)

## Warning: package 'e1071' was built under R version 4.1.3

library(tm)

## Warning: package 'tm' was built under R version 4.1.3

## Loading required package: NLP

##   
## Attaching package: 'NLP'

## The following object is masked from 'package:ggplot2':  
##   
## annotate

#library(plyr)  
#library(tidyverse)  
library(mapproj)

## Warning: package 'mapproj' was built under R version 4.1.3

## Loading required package: maps

## Warning: package 'maps' was built under R version 4.1.3

library(maps)  
library(stringr)

## Warning: package 'stringr' was built under R version 4.1.3

library(knitr)  
library(ggsn)

## Warning: package 'ggsn' was built under R version 4.1.3

## Loading required package: grid

library(usmap)

We loaded the data sets and inspected their column types, analyzed useful metrics, and returned the first few observations to ensure the data was loaded correctly and to get a feel for the data

breweries = read.csv(file.choose(), header = TRUE)  
beers = read.csv(file.choose(), header = TRUE)  
  
  
str(breweries)

## 'data.frame': 558 obs. of 4 variables:  
## $ Brew\_ID: int 1 2 3 4 5 6 7 8 9 10 ...  
## $ Name : chr "NorthGate Brewing " "Against the Grain Brewery" "Jack's Abby Craft Lagers" "Mike Hess Brewing Company" ...  
## $ City : chr "Minneapolis" "Louisville" "Framingham" "San Diego" ...  
## $ State : chr " MN" " KY" " MA" " CA" ...

str(beers)

## 'data.frame': 2410 obs. of 7 variables:  
## $ Name : chr "Pub Beer" "Devil's Cup" "Rise of the Phoenix" "Sinister" ...  
## $ Beer\_ID : int 1436 2265 2264 2263 2262 2261 2260 2259 2258 2131 ...  
## $ ABV : num 0.05 0.066 0.071 0.09 0.075 0.077 0.045 0.065 0.055 0.086 ...  
## $ IBU : int NA NA NA NA NA NA NA NA NA NA ...  
## $ Brewery\_id: int 409 178 178 178 178 178 178 178 178 178 ...  
## $ Style : chr "American Pale Lager" "American Pale Ale (APA)" "American IPA" "American Double / Imperial IPA" ...  
## $ Ounces : num 12 12 12 12 12 12 12 12 12 12 ...

head(breweries)

## Brew\_ID Name City State  
## 1 1 NorthGate Brewing Minneapolis MN  
## 2 2 Against the Grain Brewery Louisville KY  
## 3 3 Jack's Abby Craft Lagers Framingham MA  
## 4 4 Mike Hess Brewing Company San Diego CA  
## 5 5 Fort Point Beer Company San Francisco CA  
## 6 6 COAST Brewing Company Charleston SC

head(beers)

## Name Beer\_ID ABV IBU Brewery\_id  
## 1 Pub Beer 1436 0.050 NA 409  
## 2 Devil's Cup 2265 0.066 NA 178  
## 3 Rise of the Phoenix 2264 0.071 NA 178  
## 4 Sinister 2263 0.090 NA 178  
## 5 Sex and Candy 2262 0.075 NA 178  
## 6 Black Exodus 2261 0.077 NA 178  
## Style Ounces  
## 1 American Pale Lager 12  
## 2 American Pale Ale (APA) 12  
## 3 American IPA 12  
## 4 American Double / Imperial IPA 12  
## 5 American IPA 12  
## 6 Oatmeal Stout 12

summary(breweries)

## Brew\_ID Name City State   
## Min. : 1.0 Length:558 Length:558 Length:558   
## 1st Qu.:140.2 Class :character Class :character Class :character   
## Median :279.5 Mode :character Mode :character Mode :character   
## Mean :279.5   
## 3rd Qu.:418.8   
## Max. :558.0

summary(beers)

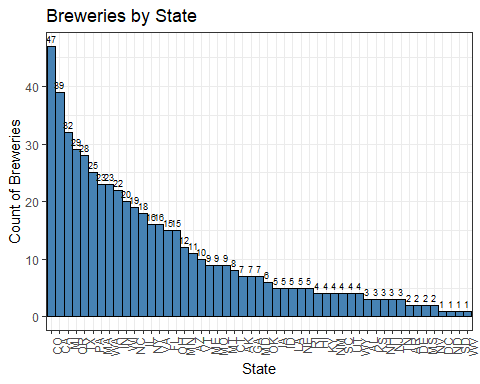
## Name Beer\_ID ABV IBU   
## Length:2410 Min. : 1.0 Min. :0.00100 Min. : 4.00   
## Class :character 1st Qu.: 808.2 1st Qu.:0.05000 1st Qu.: 21.00   
## Mode :character Median :1453.5 Median :0.05600 Median : 35.00   
## Mean :1431.1 Mean :0.05977 Mean : 42.71   
## 3rd Qu.:2075.8 3rd Qu.:0.06700 3rd Qu.: 64.00   
## Max. :2692.0 Max. :0.12800 Max. :138.00   
## NA's :62 NA's :1005   
## Brewery\_id Style Ounces   
## Min. : 1.0 Length:2410 Min. : 8.40   
## 1st Qu.: 94.0 Class :character 1st Qu.:12.00   
## Median :206.0 Mode :character Median :12.00   
## Mean :232.7 Mean :13.59   
## 3rd Qu.:367.0 3rd Qu.:16.00   
## Max. :558.0 Max. :32.00   
##

select(beers)

## data frame with 0 columns and 2410 rows

1. How many breweries are present in each state? Below we find an overview of the number of breweries for each state Colorado is the clear leader with 47 breweries followed by California and Michigan with 39 and 37, respectively 7 states are in the 20s 9 states are in the 10s and the remaining have 9 and below

# Grouping data set by state to obtain number of breweries  
breweriesByState <- breweries %>%   
 group\_by(State) %>%  
 dplyr::summarise(count = n()) %>%  
 arrange(desc(count))  
  
# Plotting data  
breweriesByState %>%  
 ggplot(aes(x = reorder(State, -count), y = count)) +  
 geom\_bar(stat = "identity", fill = "steelblue", width = 1, color = "black") +  
 geom\_text(aes(label = count), vjust = -0.5, size = 2.5) +  
 xlab("State") +  
 ylab("Count of Breweries") +  
 ggtitle("Breweries by State") +  
 theme\_bw() +  
 theme(axis.text.x = element\_text(angle = 90, hjust = 1))



2.We examined the data sets by looking at the first and last observations

#Question\_2 Merge Beer Data with Brewery Data; Outer Join  
BB.Data = merge(beers, breweries, by.x="Brewery\_id", by.y="Brew\_ID", all=TRUE)  
head(BB.Data)

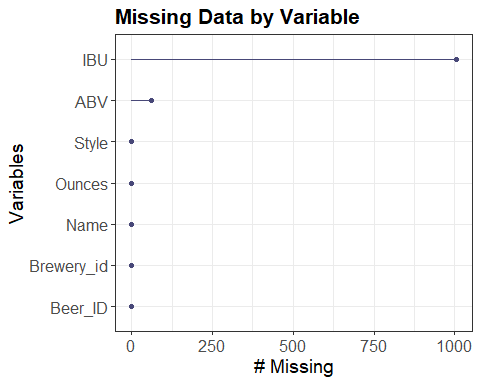
## Brewery\_id Name.x Beer\_ID ABV IBU  
## 1 1 Get Together 2692 0.045 50  
## 2 1 Maggie's Leap 2691 0.049 26  
## 3 1 Wall's End 2690 0.048 19  
## 4 1 Pumpion 2689 0.060 38  
## 5 1 Stronghold 2688 0.060 25  
## 6 1 Parapet ESB 2687 0.056 47  
## Style Ounces Name.y City  
## 1 American IPA 16 NorthGate Brewing Minneapolis  
## 2 Milk / Sweet Stout 16 NorthGate Brewing Minneapolis  
## 3 English Brown Ale 16 NorthGate Brewing Minneapolis  
## 4 Pumpkin Ale 16 NorthGate Brewing Minneapolis  
## 5 American Porter 16 NorthGate Brewing Minneapolis  
## 6 Extra Special / Strong Bitter (ESB) 16 NorthGate Brewing Minneapolis  
## State  
## 1 MN  
## 2 MN  
## 3 MN  
## 4 MN  
## 5 MN  
## 6 MN

tail(BB.Data)

## Brewery\_id Name.x Beer\_ID ABV IBU  
## 2405 556 Pilsner Ukiah 98 0.055 NA  
## 2406 557 Heinnieweisse Weissebier 52 0.049 NA  
## 2407 557 Snapperhead IPA 51 0.068 NA  
## 2408 557 Moo Thunder Stout 50 0.049 NA  
## 2409 557 Porkslap Pale Ale 49 0.043 NA  
## 2410 558 Urban Wilderness Pale Ale 30 0.049 NA  
## Style Ounces Name.y City  
## 2405 German Pilsener 12 Ukiah Brewing Company Ukiah  
## 2406 Hefeweizen 12 Butternuts Beer and Ale Garrattsville  
## 2407 American IPA 12 Butternuts Beer and Ale Garrattsville  
## 2408 Milk / Sweet Stout 12 Butternuts Beer and Ale Garrattsville  
## 2409 American Pale Ale (APA) 12 Butternuts Beer and Ale Garrattsville  
## 2410 English Pale Ale 12 Sleeping Lady Brewing Company Anchorage  
## State  
## 2405 CA  
## 2406 NY  
## 2407 NY  
## 2408 NY  
## 2409 NY  
## 2410 AK

1. Address the missing values in each column Below we explore missing values via the naniar package After exploring that IBU has a large amount of missing values, we decided to dig deeper and see which observations had missing values based on the vis\_miss plot, it seems the IBU values are randomly missing because they are distributed normally across the whole data set. The same seems true for ABV although it only accounts for 3% of the data set. After joining on state. Next, we used the gg\_miss\_upset function to visualize patterns of missingness, or rather the combination and intersection of missingness across cases. There were 62 observations that were both missing an IBU and ABV value We’ve come to the conclusion that the missing values are type MCAR (missing completely at random.)

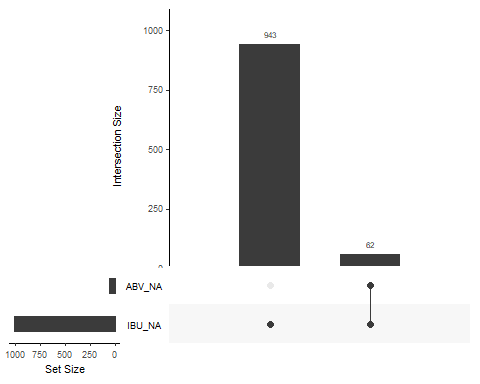
gg\_miss\_var(beers) +  
 ggtitle("Missing Data by Variable") +  
 theme\_bw() + # Set a minimal theme  
 theme(plot.title = element\_text(size = 16, face = "bold"), # Customize title  
 axis.text = element\_text(size = 12), # Customize axis text  
 axis.title = element\_text(size = 14), # Customize axis title  
 legend.title = element\_blank(), # Hide legend title  
 legend.text = element\_text(size = 12)) # Customize legend text



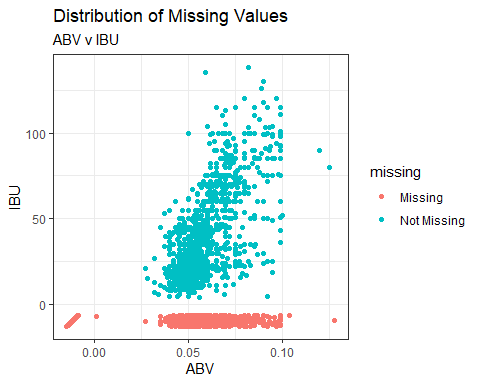
# To see if the values are distributed randomly across the data set  
vis\_miss(beers) +  
 theme\_bw() +  
 ggtitle("Distribution of Missing Data")



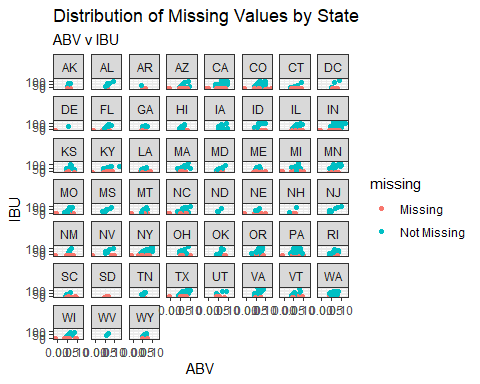
df <- left\_join(breweries, beers, by = c("Brew\_ID" = "Brewery\_id"))  
   
# To see combination and intersections  
gg\_miss\_upset(df)



# gg\_miss to see distribution of missing abv v ibu correlation  
df %>% ggplot(aes(x = ABV, y = IBU)) +  
 geom\_miss\_point() +  
 theme\_bw() +  
 ggtitle(label = "Distribution of Missing Values", subtitle = "ABV v IBU")



# gg\_miss to see distribution of missing abv v ibu correlation  
df %>% ggplot(aes(x = ABV, y = IBU)) +  
 geom\_miss\_point() +  
 theme\_bw() +  
 ggtitle(label = "Distribution of Missing Values by State", subtitle = "ABV v IBU") +  
 facet\_wrap(~State)



1. Building a KNN to see the relationship between beer type with respect to ABV and IBU. First, we broke up the style between “IPA” and “Other” and assigned them to an additional column labeled “type”. Then we created a 70/30 train/test set based on the beers.csv. We excluded NAs on ABV and IBU to get a more accurate representation.Then, we set an iteration to test the k parameter and see what iteration provided the best accuracy. After running it multiple times, we’ve determined that the accuracy is best achieved between k = 30 and 80.

library(class)  
library(caret)  
library(e1071)  
library(dplyr)  
  
#distinguishing between IPA and other  
beers <- beers %>% mutate(type = ifelse(grepl('IPA', Style), "IPA", "Other"))  
  
head(beers)

## Name Beer\_ID ABV IBU Brewery\_id  
## 1 Pub Beer 1436 0.050 NA 409  
## 2 Devil's Cup 2265 0.066 NA 178  
## 3 Rise of the Phoenix 2264 0.071 NA 178  
## 4 Sinister 2263 0.090 NA 178  
## 5 Sex and Candy 2262 0.075 NA 178  
## 6 Black Exodus 2261 0.077 NA 178  
## Style Ounces type  
## 1 American Pale Lager 12 Other  
## 2 American Pale Ale (APA) 12 Other  
## 3 American IPA 12 IPA  
## 4 American Double / Imperial IPA 12 IPA  
## 5 American IPA 12 IPA  
## 6 Oatmeal Stout 12 Other

# Setting percentage of .70 for train  
perc = .7  
trainIndices = sample(1:dim(beers)[1],round(perc \* dim(beers)[1]))  
train = beers[trainIndices,] # Assigning 70% of beers to train  
test = beers[-trainIndices,] # Assigning other 30% of beers to test  
  
train = train %>% filter(is.na(ABV) == FALSE & is.na(IBU) == FALSE) #excluding NAs  
test = test %>% filter(is.na(ABV) == FALSE & is.na(IBU) == FALSE) #exclusing NAs  
  
accs = data.frame(accuracy = numeric(100), k = numeric(100)) # making accuracy data frame  
  
for(i in 1:100) #iterating over 1-90 k's  
{  
 # setting the classifications based on train and test  
 classifications = knn(train[,c(3,4)],test[,c(3,4)],train$type, prob = TRUE, k = i)  
 # returning table to see which ones were right and wrongs  
 table(test$type,classifications)  
 # returning confusion matrix to get overall stats  
 CM = confusionMatrix(table(test$type,classifications))  
 # assigning accuracy at iteration i from overall score  
 accs$accuracy[i] = CM$overall[1]  
 # assigning iteration i to k at [i]  
 accs$k[i] = i  
}  
  
# plotting accuracy of k  
plot(accs$k,accs$accuracy, type = "l", xlab = "k", main = "Accuracy of K")

