DDS_Project_JR_LM

Jeff Reed, Logan Miller

6/10/2021

Contents

Import Relevant Packages

```
library(ggplot2)
library(magrittr)
library(dplyr)
library(GGally)
library(tibble)
library(class)
library(caret)
library(e1071)
library(readr)
library(ggthemes)
```

Load Data

```
Breweries = read.csv('C:/Users/L/Downloads/Breweries.csv')

Beers = read.csv('C:/Users/L/Downloads/Beers.csv')
```

Inspect Data

```
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                  Style Ounces
## 1             Chugach Session Ale     919 0.048  NA        494              Cream Ale     12
## 2              Snowshoe White Ale     587 0.048  12        224                Witbier     12
## 3            King Street Blonde Ale    1665 0.049  NA        103     American Blonde Ale     12
## 4        Urban Wilderness Pale Ale      30 0.049  NA        558        English Pale Ale     12
## 5        Northern Lights Amber Ale     921 0.050  15        494 American Amber / Red Ale     12
## 6 Peninsula Brewers Reserve (PBR)    1187 0.050  15        459     American Blonde Ale     12
```

Counting the breweries in each state

```
Breweries %>% count(State, sort = TRUE)
```

```
##    State  n
## 1     CO 47
## 2     CA 39
## 3     MI 32
## 4     OR 29
## 5     TX 28
## 6     PA 25
## 7     MA 23
## 8     WA 23
## 9     IN 22
## 10    WI 20
## 11    NC 19
## 12    IL 18
## 13    NY 16
## 14    VA 16
## 15    FL 15
## 16    OH 15
## 17    MN 12
## 18    AZ 11
## 19    VT 10
## 20    ME  9
## 21    MO  9
## 22    MT  9
```

```
## 23    CT  8
## 24    AK  7
## 25    GA  7
## 26    MD  7
## 27    OK  6
## 28    IA  5
## 29    ID  5
## 30    LA  5
## 31    NE  5
## 32    RI  5
## 33    HI  4
## 34    KY  4
## 35    NM  4
## 36    SC  4
## 37    UT  4
## 38    WY  4
## 39    AL  3
## 40    KS  3
## 41    NH  3
## 42    NJ  3
## 43    TN  3
## 44    AR  2
## 45    DE  2
## 46    MS  2
## 47    NV  2
## 48    DC  1
## 49    ND  1
## 50    SD  1
## 51    WV  1
```

Merging beer data with brewery data

```
# Change column name in Brewery file to clarify names

Beers = Beers %>% rename(Beer_Name = Name, Brew_ID = Brewery_id)

colnames(Breweries)[2] = "Brewery_Name"

Beer_Brew_Combo = merge(Beers,Breweries,c("Brew_ID"))
```

First and last 6 rows

```
head(Beer_Brew_Combo, 6)
```

```
##   Brew_ID     Beer_Name Beer_ID   ABV IBU                               Style Ounces       Brewery_Na
## 1       1       Pumpion    2689 0.060  38                         Pumpkin Ale     16 NorthGate Brewi
## 2       1 Maggie's Leap    2691 0.049  26                  Milk / Sweet Stout     16 NorthGate Brewi
## 3       1   Parapet ESB    2687 0.056  47 Extra Special / Strong Bitter (ESB)     16 NorthGate Brewi
## 4       1    Stronghold    2688 0.060  25                     American Porter     16 NorthGate Brewi
```

```
## 5       1    Wall's End    2690 0.048  19                   English Brown Ale     16 NorthGate Brewi
## 6       1  Get Together    2692 0.045  50                         American IPA     16 NorthGate Brewi
```

```r
tail(Beer_Brew_Combo, 6)
```

```
##      Brew_ID                 Beer_Name Beer_ID   ABV IBU                   Style Ounces
## 2405     556              Pilsner Ukiah      98 0.055  NA         German Pilsener     12         Ukia
## 2406     557  Heinnieweisse Weissebier      52 0.049  NA              Hefeweizen     12        Butter
## 2407     557        Porkslap Pale Ale      49 0.043  NA American Pale Ale (APA)     12        Butter
## 2408     557          Snapperhead IPA      51 0.068  NA            American IPA     12        Butter
## 2409     557        Moo Thunder Stout      50 0.049  NA      Milk / Sweet Stout     12        Butter
## 2410     558 Urban Wilderness Pale Ale      30 0.049  NA        English Pale Ale     12 Sleeping Lad
##      State
## 2405    CA
## 2406    NY
## 2407    NY
## 2408    NY
## 2409    NY
## 2410    AK
```

Check missing values in each column

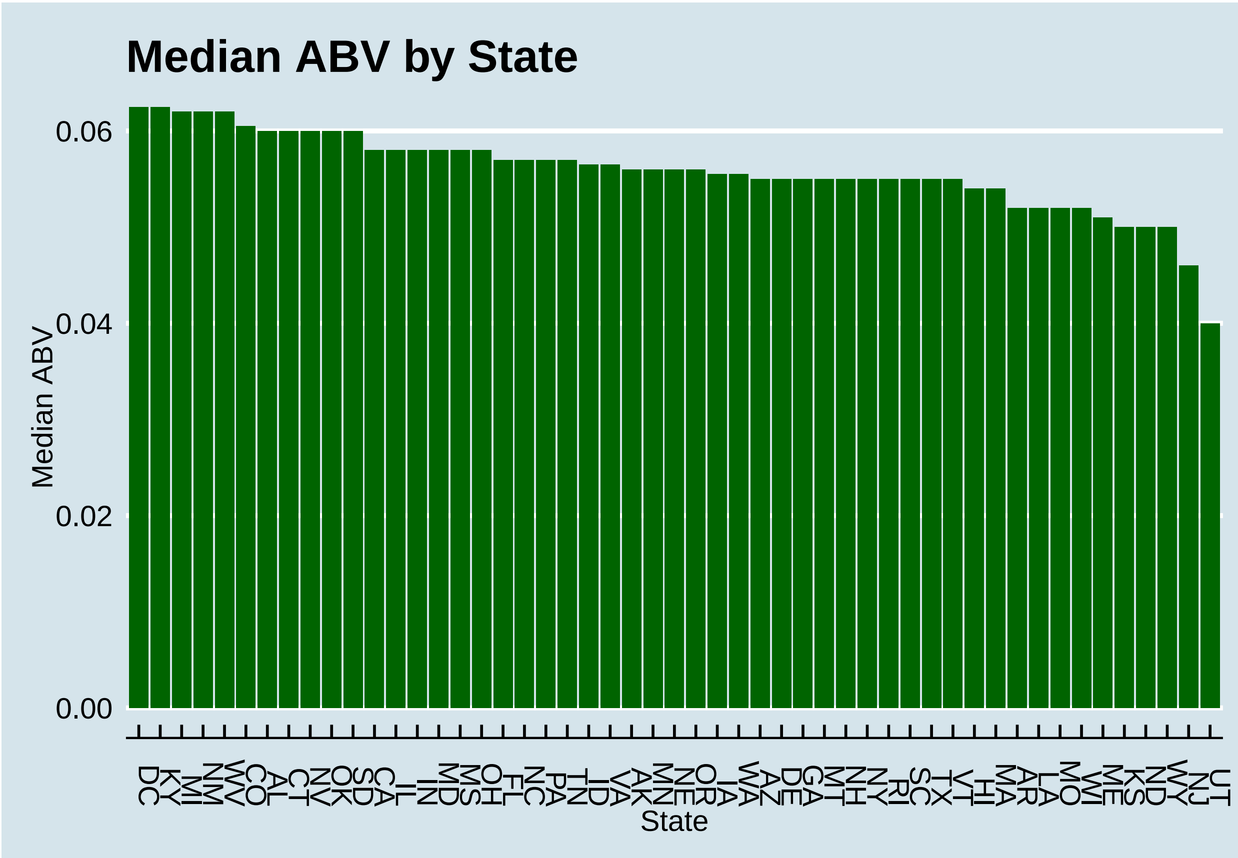
```r
sapply(Beer_Brew_Combo, function(x) sum(is.na(x)))
```

```
##      Brew_ID    Beer_Name      Beer_ID          ABV          IBU        Style       Ounces Brewery_Na
##            0            0            0           62         1005            0            0
```

We can see that only the ABV and IBU columns have missing values. For now we will keep them in but they may need to be filtered out later to avoid misleading results.

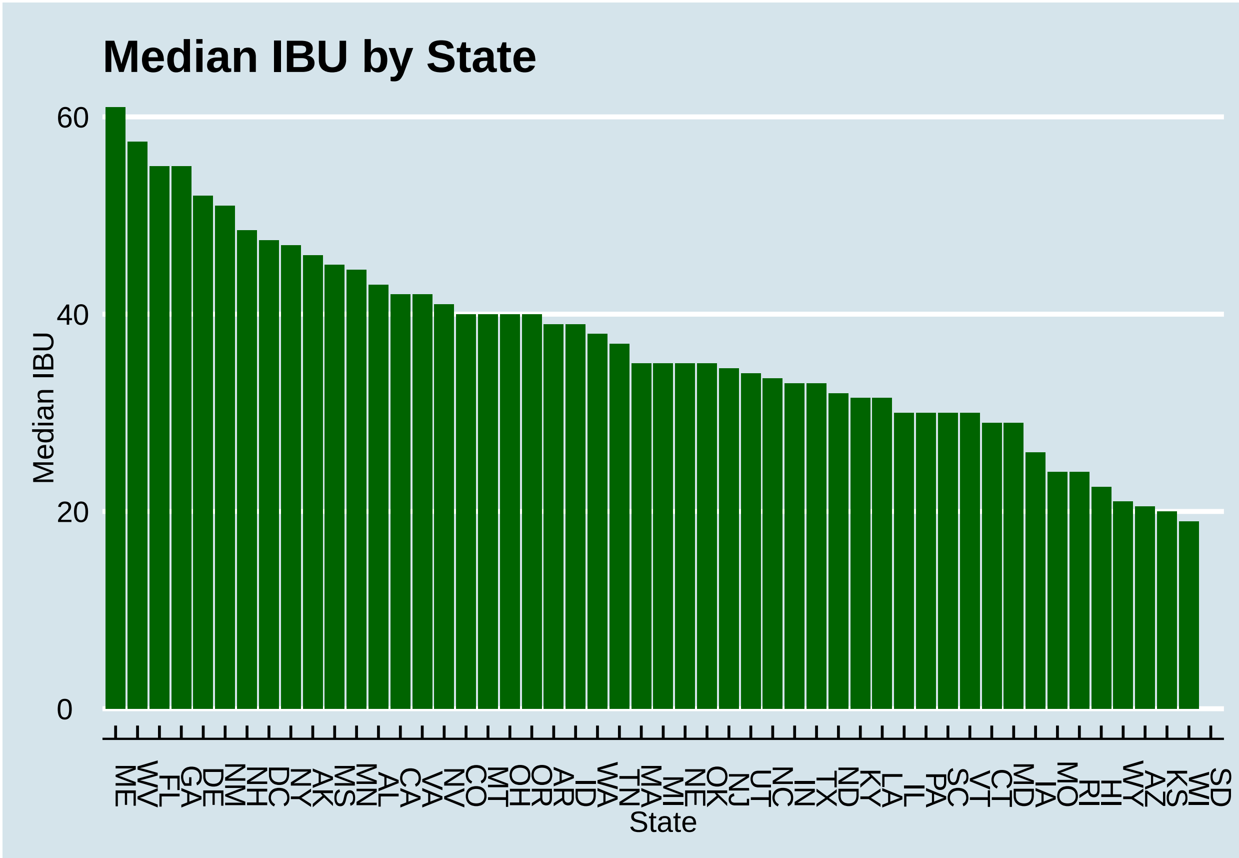
Computing and plotting median ABV and IBU

```r
ABV_IBU_Medians = Beer_Brew_Combo %>%
  group_by (State) %>%
  summarise(Median_ABV = median(ABV, na.rm = TRUE), # na.rm filters out the missing values when computi
            Median_IBU = median(IBU, na.rm = TRUE))

ABV_IBU_Medians %>%
  ggplot(aes(x = reorder(State, -Median_ABV,), y = Median_ABV)) +
  geom_bar(stat = 'identity', fill = 'dark green') +
  labs(x = 'State', y = 'Median ABV', title = 'Median ABV by State') +
  theme_economist() +
  theme(legend.position = 'none', axis.text.x = element_text(angle = 270, hjust = 1))
```




```
ABV_IBU_Medians %>%
  ggplot(aes(x = reorder(State, -Median_IBU,), y = Median_IBU)) +
  geom_bar(stat = 'identity', fill = 'dark green') +
  labs(x = 'State', y = 'Median IBU', title = 'Median IBU by State') +
  theme_economist() +
  theme(legend.position = 'none', axis.text.x = element_text(angle = 270, hjust = 1))
```

```
## Warning: Removed 1 rows containing missing values (position_stack).
```



Which state has the maximum alcoholic beer? Which state has the most bitter beer?

```
Beer_Brew_Combo %>% arrange(-ABV) %>% select(Beer_Name, ABV, State) %>% head(1)
```

```
##                                              Beer_Name   ABV State
## 1 Lee Hill Series Vol. 5 - Belgian Style Quadrupel Ale 0.128    CO
```

```
Beer_Brew_Combo %>% arrange(-IBU) %>% select(Beer_Name, IBU, State) %>% head(1)
```

```
##                    Beer_Name IBU State
## 1 Bitter Bitch Imperial IPA 138    OR
```

Colorado has the beer with the highest ABV at 0.128.

Oregon has the beer with the highest IBU at 138.

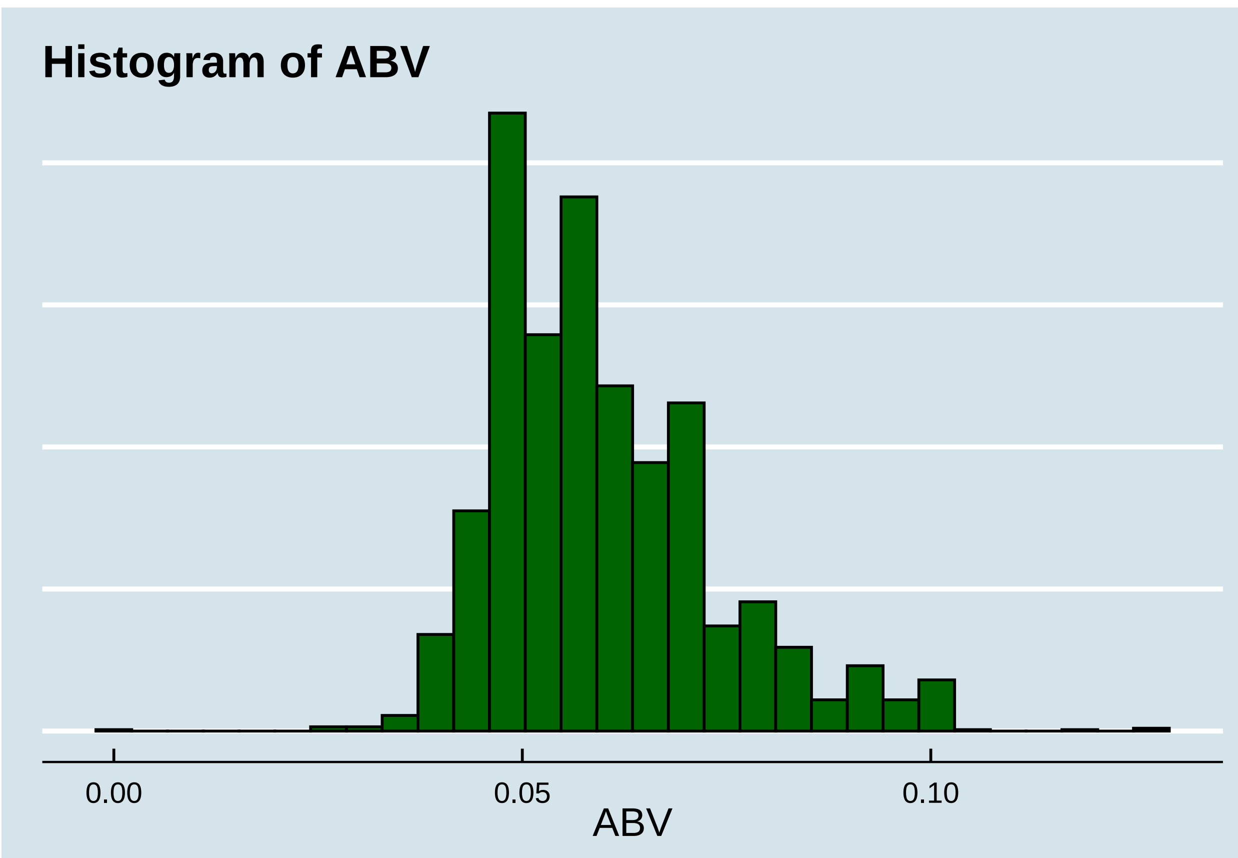
Comment on the summary statistic and distribution of the ABV variable

```
summary(Beer_Brew_Combo$ABV)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
## 0.00100 0.05000 0.05600 0.05977 0.06700 0.12800      62
```

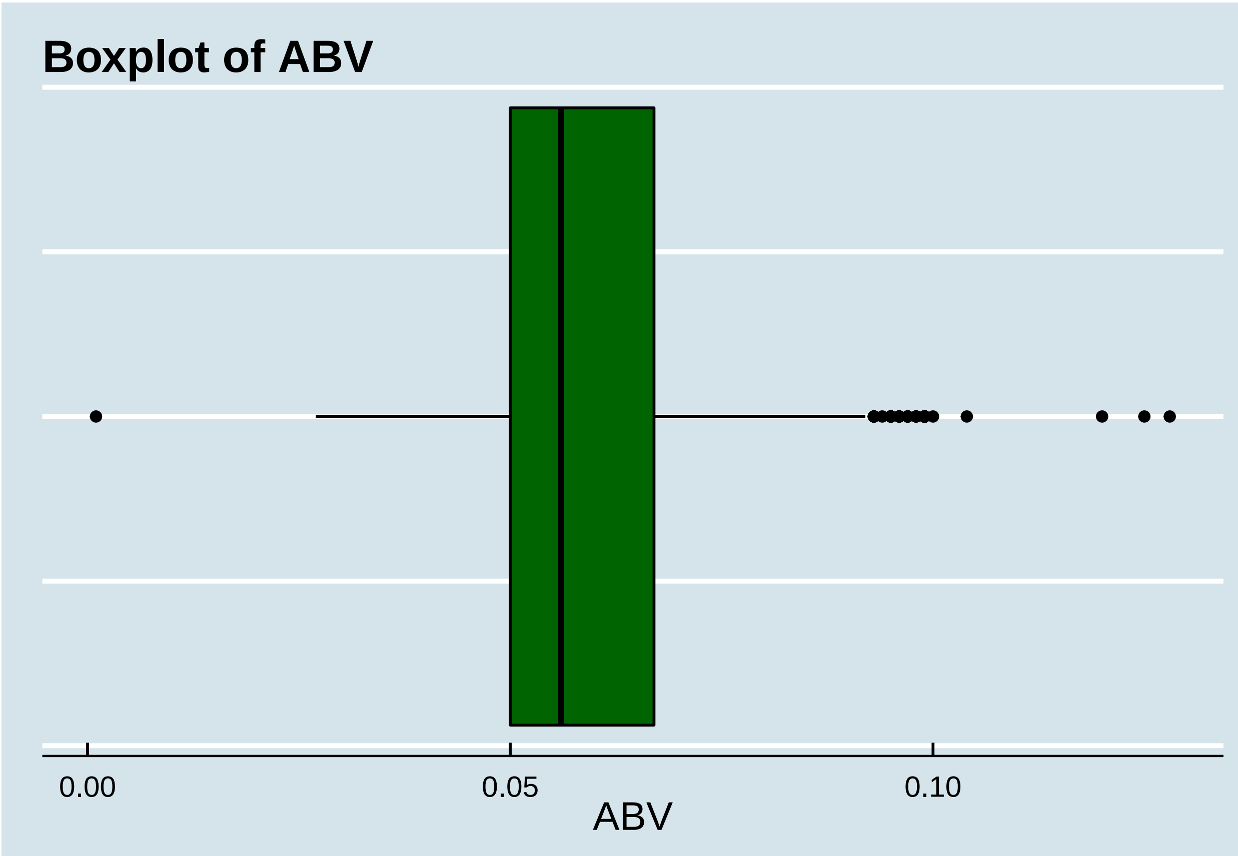
```r
write.csv(data.frame(Mean = round(mean(Beer_Brew_Combo$ABV, na.rm = TRUE), 3),
           Trimmed_Mean = round(mean(Beer_Brew_Combo$ABV, na.rm = TRUE, trim = 0.1), 3),
           Median = median(Beer_Brew_Combo$ABV, na.rm = TRUE),
           Standard_Deviation = round(sd(Beer_Brew_Combo$ABV, na.rm = TRUE), 3))
          , 'C:/Users/L/Downloads/summary.csv'
          , row.names = FALSE)

Beer_Brew_Combo %>%
  ggplot(aes(x = ABV)) +
  geom_histogram(fill = 'dark green', color = 'black') +
  labs(x = 'ABV', y = '',  title = 'Histogram of ABV') +
  theme_economist() +
  theme(axis.text.y = element_blank(), axis.title.x = element_text(size = 15),legend.position = 'none')
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 62 rows containing non-finite values (stat_bin).
```




```r
Beer_Brew_Combo %>%
  ggplot(aes(x = ABV)) +
  geom_boxplot(fill = 'dark green', color = 'black') +
  labs(x = 'ABV', y = '',  title = 'Boxplot of ABV') +
  theme_economist() +
  theme(axis.text.y = element_blank(), axis.title.x = element_text(size = 15),legend.position = 'none')
```

```
## Warning: Removed 62 rows containing non-finite values (stat_boxplot).
```



The lack of difference between the mean and the 10% trimmed mean tells us there are minimal outliers in the data.

Looking at the histogram and boxplot, it is apparent the data is approximately normally distributed

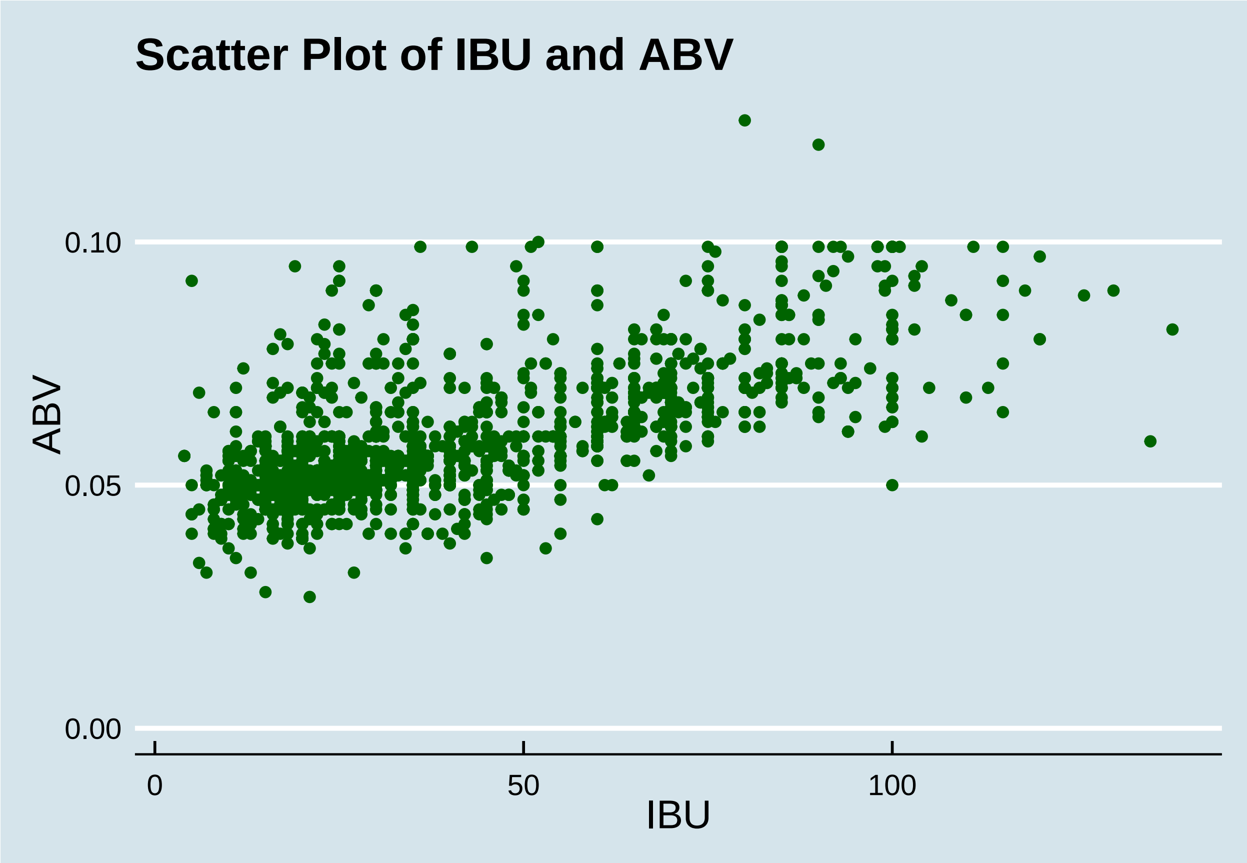
Is there a relationship between ABV and IBU?

```
cor(x = Beer_Brew_Combo$IBU, y = Beer_Brew_Combo$ABV, method = 'pearson', use = 'na.or.complete')
```

```
## [1] 0.6706215
```

```
Beer_Brew_Combo %>%
  ggplot(aes(x = IBU, y = ABV)) +
  geom_point(color = 'dark green') +
  labs(title = 'Scatter Plot of IBU and ABV') +
  theme_economist() +
  theme(axis.title.y = element_text(size = 15), axis.title.x = element_text(size = 15),legend.position =
```

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



Based on a visual inspection of the scatter plot, there appears to be a slight positive correlation between IBU and ABV

This is supported by a linear correlation coefficient of 0.67

Both of the graph and the correlation coefficient suggest that, in general, as IBU increases, so does ABV

Using KNN to investigate IBU vs ABV in IPAs and other Ales

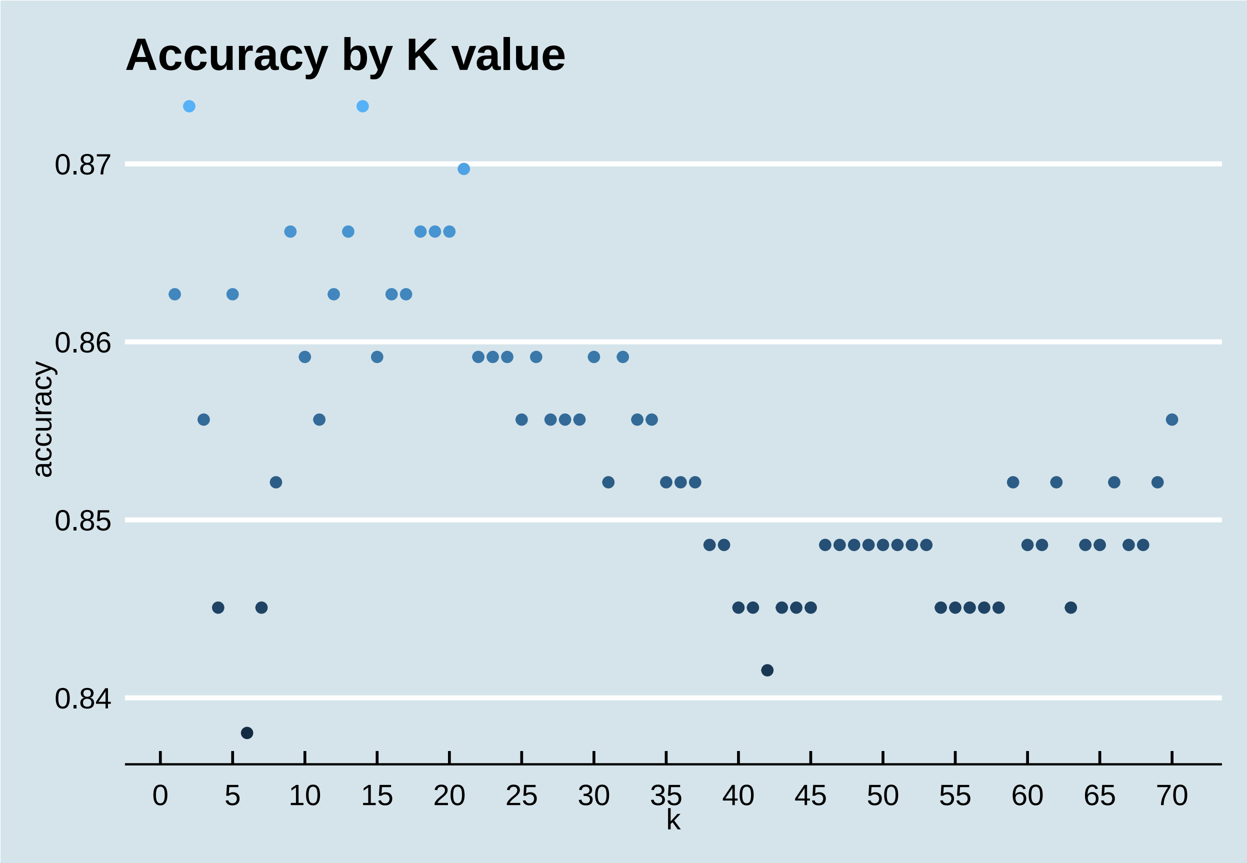
Deciding best K to use

```
ipa_ale_df = Beer_Brew_Combo %>%
  filter(!is.na(ABV) &
           !is.na(IBU) &
           (grepl('\\bIPA\\b', Style, ignore.case = TRUE) | !grepl('\\bIPA\\b', Style) &
           grepl('\\bALE\\b', Style, ignore.case = TRUE))
         ) %>%
  mutate(isIPA = ifelse(grepl('\\bIPA\\b', Style, ignore.case = TRUE), 1, 0),
         scaled_ibu = scale(IBU),
         scaled_abv = scale(ABV)
    )

sample_size = floor(.70 * nrow(ipa_ale_df))
```

```r
set.seed(67)

train_index = sample(seq_len(nrow(ipa_ale_df)), size = sample_size)

train_df = ipa_ale_df[train_index, ]
test_df = ipa_ale_df[-train_index, ]

accuracy_df = data.frame(accuracy = numeric(70), k = numeric(70))

for(i in 1:70)
{
  beer_classifications = knn(train_df[, c(12,13)],
                    test_df[, c(12,13)],
                    train_df$isIPA,
                    prob = TRUE, k = i)

  CM = confusionMatrix(table(beer_classifications, test_df$isIPA))

  accuracy_df$accuracy[i] = CM$overall[1]
  accuracy_df$k[i] = i

}
print(accuracy_df %>% arrange(-accuracy) %>% head())
```

```
##    accuracy  k
## 1 0.8732394  2
## 2 0.8732394 14
## 3 0.8697183 21
## 4 0.8661972  9
## 5 0.8661972 13
## 6 0.8661972 18
```

```r
accuracy_df %>%
  ggplot(aes(x = k, y = accuracy, color = accuracy)) +
  geom_point() +
  scale_x_continuous(breaks = seq(0, 70, 5)) +
  labs(title = 'Accuracy by K value') +
  theme_economist() +
  theme(legend.position = 'none')
```



Highest accuracy occurs when k = 2 or 14, so we will go with 14

```
set.seed(67)

classifications = knn(train_df[, c(12,13)],
                      test_df[, c(12,13)],
                      train_df$isIPA,
                      prob = TRUE, k = 14)

print(confusionMatrix(table(classifications, test_df$isIPA)))
```

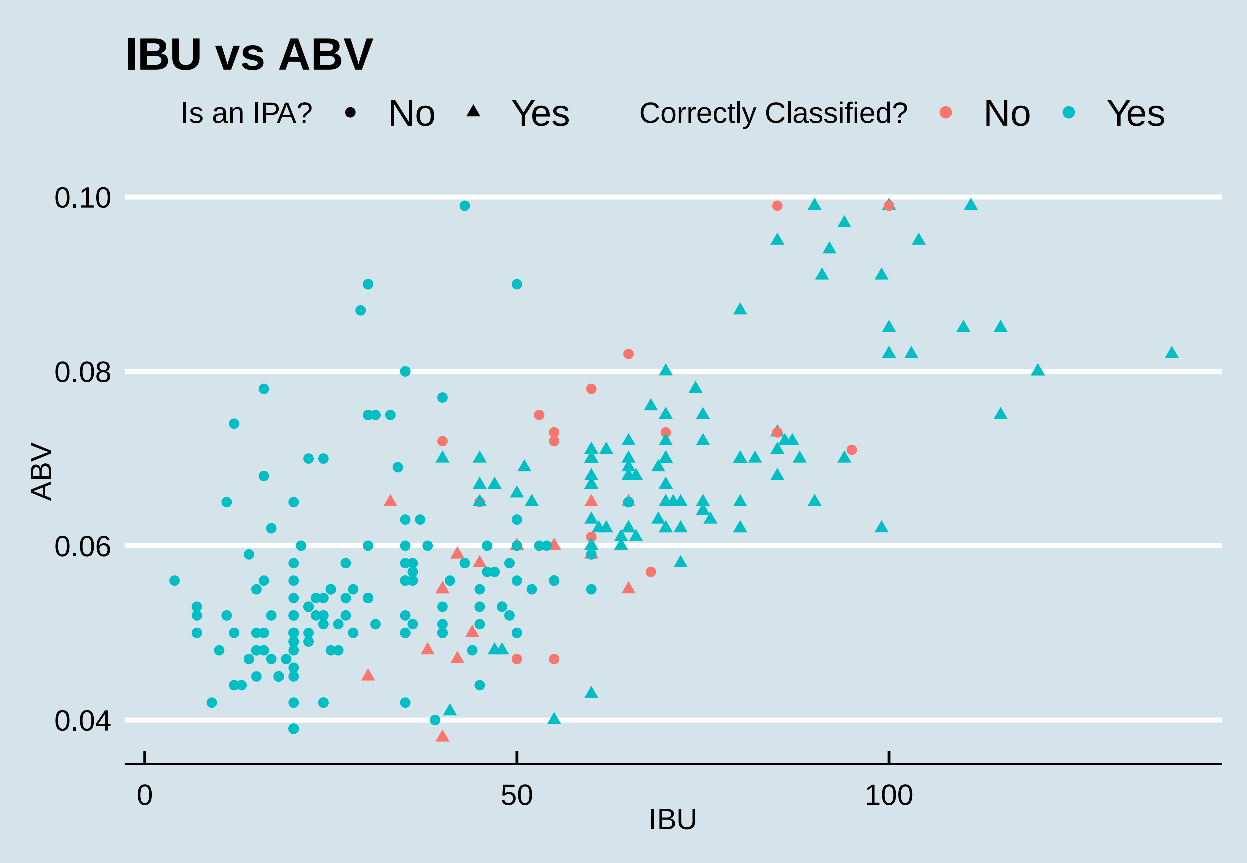
```
## Confusion Matrix and Statistics
##
##
## classifications   0   1
##               0 144  19
##               1  17 104
##
##                Accuracy : 0.8732
##                  95% CI : (0.8289, 0.9096)
##     No Information Rate : 0.5669
##     P-Value [Acc > NIR] : <2e-16
##
##                   Kappa : 0.7414
##
##  Mcnemar's Test P-Value : 0.8676
```

```
##
##             Sensitivity : 0.8944
##             Specificity : 0.8455
##          Pos Pred Value : 0.8834
##          Neg Pred Value : 0.8595
##              Prevalence : 0.5669
##          Detection Rate : 0.5070
##    Detection Prevalence : 0.5739
##       Balanced Accuracy : 0.8700
##
##        'Positive' Class : 0
##
```

Using just ABV and IBU in KNN analysis when k = 14, IPAs were correctly classified 87.3% of the time

Visualizing the above conculusion

```
test_df$classification = classifications


test_df = test_df %>%
  mutate(correct_classification = factor(ifelse(isIPA == classification, 1, 0), labels = c('No', 'Yes')
         isIPA = factor(ifelse(isIPA == 1, 1, 0), labels = c('No', 'Yes')))

test_df %>%
  ggplot(aes(x = IBU, y = ABV, color = correct_classification)) +
  geom_point(aes(shape = isIPA)) +
  theme_economist() +
  labs(title = 'IBU vs ABV', color = 'Correctly Classified?', shape = 'Is an IPA?')
```


IBU vs ABV
Is an IPA? No Yes
Correctly Classified? No Yes
0.10
0.08
0.06
0.04
ABV
0
50
100
IBU