Beer Reviews Analysis

Yang Peng

23 September 2018

Summary of the Dataset

This dataset consists of approximately 1.5 million beer reviews from Beer Advocate.

Number of Instances:

```
dim(beer.reviews)[1]
```

```
## [1] 1586614
```

Number of Attributes:

```
dim(beer.reviews)[2]
```

```
## [1] 13
```

Dataset example:

```
kable(beer.reviews[1:5,])
```

brewery_id	brewery_name	review_time	review_overall	review_aroma	review_appearance	review_profilename	beer_style	review_palate	review
10325	Vecchio Birraio	1234817823	1.5	2.0	2.5	stcules	Hefeweizen	1.5	
10325	Vecchio Birraio	1235915097	3.0	2.5	3.0	stcules	English Strong Ale	3.0	
10325	Vecchio Birraio	1235916604	3.0	2.5	3.0	stcules	Foreign / Export Stout	3.0	
10325	Vecchio Birraio	1234725145	3.0	3.0	3.5	stcules	German Pilsener	2.5	
1075	Caldera Brewing Company	1293735206	4.0	4.5	4.0	johnmichaelsen	American Double / Imperial IPA	4.0	

Five number summarys

```
summary(beer_abv)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.01    5.20    6.50    7.04    8.50   57.70   67785
```

```
summary(review_overall)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   3.500   4.000   3.816   4.500   5.000
```

```
summary(review_aroma)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   3.500   4.000   3.736   4.000   5.000
```

```
summary(review_appearance)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   3.500   4.000   3.842   4.000   5.000
```

```
summary(review_palate)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   3.500   4.000   3.744   4.000   5.000
```

```
summary(review_taste)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   3.500   4.000   3.793   4.500   5.000
```

From the summary above, we know that there are many missing values in beer_abv and the review score is ranging from 0 to 5.

Data Cleaning and Checking

Check the count of missing or empty values for beer_abv and brewery_name in Qestion 1:

```
sum(is.na(beer_abv)|beer_abv=="")
```

```
## [1] 67785
```

```
sum(is.na(brewery_name)|brewery_name=="")
```

```
## [1] 15
```

Remove NA values and save into d2:

```
d2 <- na.omit(beer.reviews)#remove rows contains NA value
dim(d2)
```

```
## [1] 1518829      13
```

```
(dim(beer.reviews)[1]-dim(d2)[1])/dim(beer.reviews)[1] #check the percent of the removed rows.
```

```
## [1] 0.04272306
```

It shows that the cleaned parts is less than 5%, so we used the cleaned dataset d2 for the following analysis.

Check the length of unique brewery_name and brewery_id:

```
length(unique(d2$brewery_name))
```

```
## [1] 5156
```

```
length(unique(d2$brewery_id))
```

```
## [1] 5232
```

It turns out the number of unique brewey_id is greater than the number of unique brewery_name, which means different id may have same brewery_name. It can be caused by cutting off by the length of charater.Same as beer_id and beer_name:

```
length(unique(d2$beer_name))
```

```
## [1] 44085
```

```
length(unique(d2$beer_beerid))
```

```
## [1] 49012
```

So here I used brewery_id and beer_beerid as the index of different categories.

1. Which brewery produces the strongest beers by ABV%?

Find the average beer_adv group by brewery_id:

```
a <- aggregate(d2$beer_abv, list(d2$brewery_id), mean,na.omit=T)
dim(a)
```

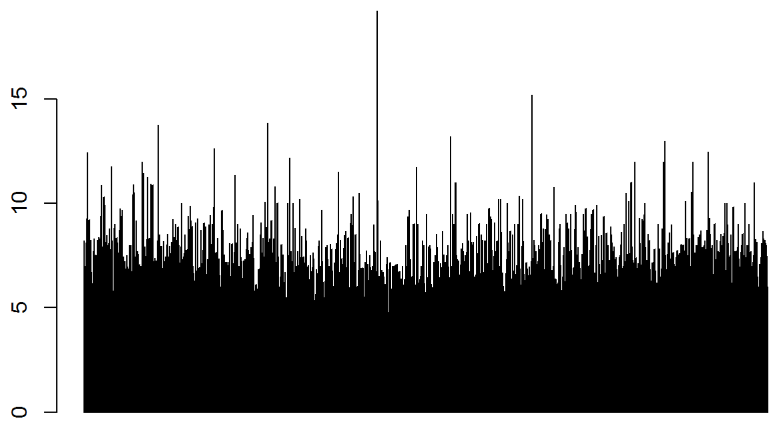
```
## [1] 5232    2
```

Find the unique brewery name which has the max beer_abv:

```
ii <- a$Group.1[a$x==max(a$x)]
unique(d2$brewery_name[d2$brewery_id==ii])#[1] SchorschbrÃ¤u
```

```
## [1] SchorschbrÃ¤u
## 5743 Levels:  't Hofbrouwerijke ... Zywiec Breweries PLC (Heineken)
```

```
barplot(a$x)
```

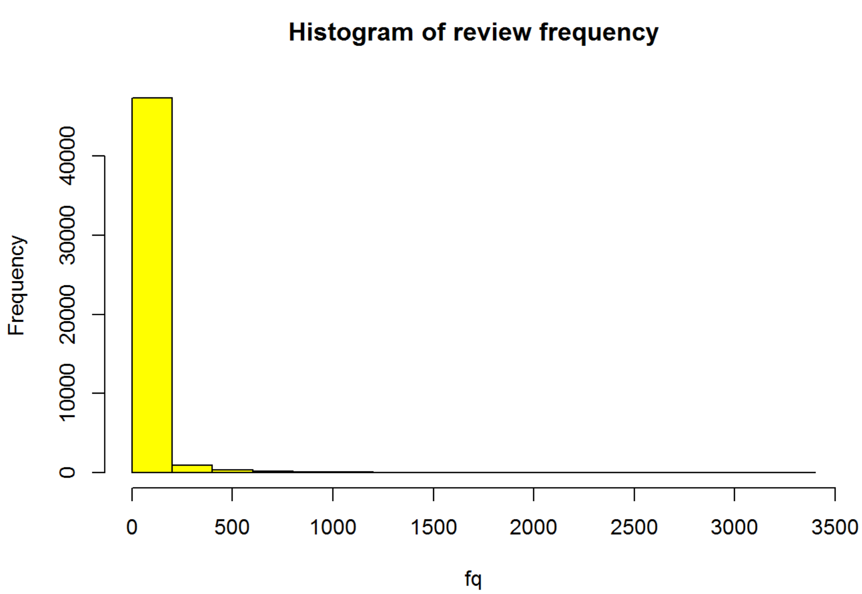


It shows SchorschbrÃ¤u has the max beer ABV% and we can see from the barplot that only one brewery has the noticeable highest abv, which is SchorschbrÃ¤u. **So SchorschbrÃ¤u produces the strongest beers by ABV%.**

2. If you had to pick 3 beers to recommend using only this data, which would you pick?

Find the review frequency histogram group by beer_beerid:

```
fq <- table(d2$beer_beerid)
hist(fq,col="yellow",breaks = 20,main="Histogram of review frequency")
```



Use mean review_overall to determine the popularity:

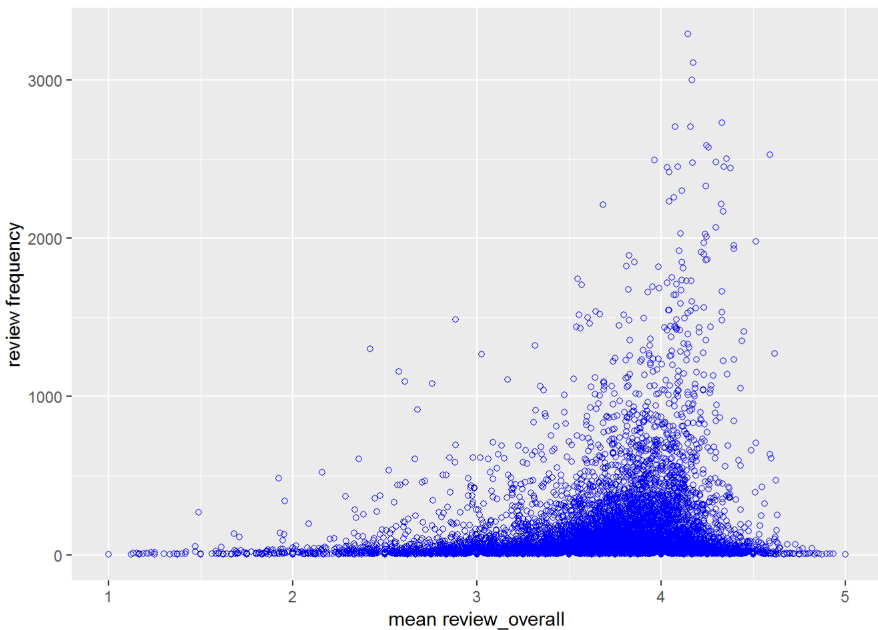
```
table(d2$review_overall)
```

```
##
##      0      1    1.5      2    2.5      3    3.5      4    4.5      5
##      7  10211  12035  35755  54696 155902 286972 559869 314365  89017
```

There are 89017 reviews has 5.

Find the average review_overall group by beer_beerid. Plot for “mean review_overall” vs “review frequency”:

```
b <- aggregate(d2$review_overall, list(d2$beer_beerid), mean,na.omit=T)
t <- as.data.frame(cbind(b[,2],fq))
colnames(t) <- c("mean review_overall","review frequency")
ggplot(t, aes(x=`mean review_overall`, y=`review frequency`)) + geom_point(shape=1,color="blue")
```



A good beer should has a reasonable amount of reviews and has a high review score as well. In other words, large number of the reviews, large mean of the review_overall and small variance of review_overall consist of the criterion of ranking beers.

So, here I used the lower bound of 95% confidence interval as the measurement of ranking beers:x_bar-t*sd/sqrt(n).

Moreover, the number of reviews cannot be too small, based on the scatter plot above, here I use 100 as the threshold:

```
ss <- split.data.frame(d2[,c(4,13)],d2$beer_beerid)

getCI <- function(x){
  if(dim(x)[1]<100){
    lowci=0
  }else{
    b <- t.test(x$review_overall)
    lowci <- b$conf.int[1]
  }

}

top3 <- sort(sapply(ss,getCI),decreasing = T)[1:3]
unique(d2$beer_name[d2$beer_beerid %in% names(top3)])
```

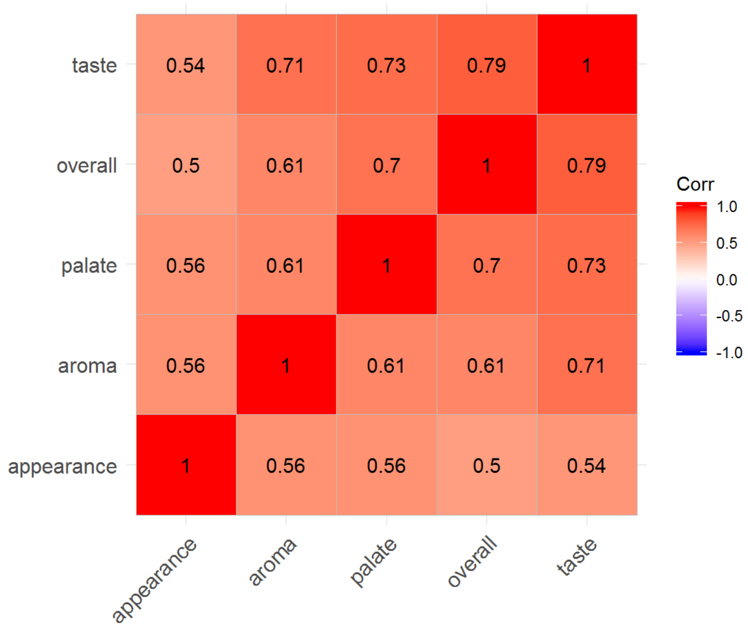
```
## [1] Citra DIPA                Heady Topper
## [3] Trappist Westvleteren 12
## 56857 Levels: '55' Lager '71 Pale Ale ... ZZ Lager
```

It shows that “Citra DIPA”, “Heady Topper” and “Trappist Westvleteren 12” are the three beers that I want to recommended based on this dataset.

3. Which of the factors (aroma, taste, appearance, palette) are most important in determining the overall quality of a beer?

First, I used the correlations between these fators to figure out which one is most important:

```
reviewMt <- cbind(d2$review_overall,d2$review_aroma,d2$review_appearance,d2$review_palate,d2$review_taste)
revcor <-round(cor(reviewMt),2)
colnames(revcor) <- rownames(revcor) <- c("overall","aroma","appearance","palate","taste")
ggcorrplot(revcor, hc.order = TRUE,lab = TRUE)
```



From the correlation heat plot we can see that taste is the most important.

Second, I incoporate linear model and t-test to support the above result:

```
# fit a linear model for diff reviews
lmfit <- lm(d2$review_overall~d2$review_aroma+d2$review_appearance+d2$review_palate+d2$review_taste)
summary(lmfit)
```

```
##
## Call:
## lm(formula = d2$review_overall ~ d2$review_aroma + d2$review_appearance +
##     d2$review_palate + d2$review_taste)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.8681 -0.2570 -0.0067  0.2453  3.6610
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)          0.4477379  0.0023713  188.82   <2e-16 ***
## d2$review_aroma      0.0476247  0.0007406   64.31   <2e-16 ***
## d2$review_appearance 0.0357281  0.0007168   49.84   <2e-16 ***
## d2$review_palate     0.2579818  0.0007785  331.37   <2e-16 ***
## d2$review_taste      0.5498900  0.0007957  691.08   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4219 on 1518824 degrees of freedom
## Multiple R-squared:  0.6541, Adjusted R-squared:  0.6541
## F-statistic: 7.18e+05 on 4 and 1518824 DF,  p-value: < 2.2e-16
```

Based on the p-value and t-statistic can also get the same conclusion.

The 65.41% R-square is not satisfactory. Since Review score is from 0 to 5, I transformed the review_overall to be from -Inf to Inf and refit linear model to see if it can improve R-square:

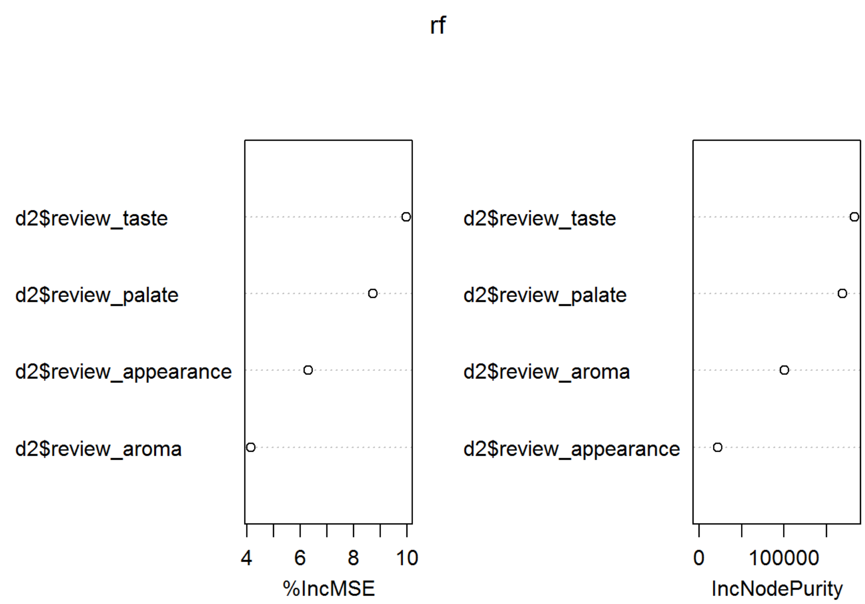
```
y1 <- d2$review_overall/5
y <- log((y1+0.001)/(1.001-y1))
lmfit3 <- lm(y~d2$review_aroma+d2$review_appearance+d2$review_palate+d2$review_taste)
summary(lmfit3)
```

```
##
## Call:
## lm(formula = y ~ d2$review_aroma + d2$review_appearance + d2$review_palate +
##     d2$review_taste)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.2429 -0.5635 -0.2687  0.1513  9.2574
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          -3.744288   0.006856 -546.16   <2e-16 ***
## d2$review_aroma       0.062739   0.002141   29.30   <2e-16 ***
## d2$review_appearance  0.095146   0.002072   45.91   <2e-16 ***
## d2$review_palate      0.441914   0.002251  196.33   <2e-16 ***
## d2$review_taste       0.795824   0.002300  345.94   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.22 on 1518824 degrees of freedom
## Multiple R-squared:  0.3493, Adjusted R-squared:  0.3493
## F-statistic: 2.038e+05 on 4 and 1518824 DF,  p-value: < 2.2e-16
```

Since the R-square reduced to 35%, such transformation is not successful for the model fitting.

Lastly, I fit a nonlinear model, Random Forest, to vote for the most important factor:

```
rf <- randomForest(d2$review_overall~d2$review_aroma+d2$review_appearance+d2$review_palate+d2$review_taste,
                   importance=TRUE,
                   ntree=20)
varImpPlot(rf)
```



The plot above also indicates the variable taste to be the most important factor, which is consistent with the finding in linear model.

In sum, taste is the most important fator based on all the analysis above.

4. Lastly, if I typically enjoy a beer due to its aroma and appearance, which beer style should I try?

Add a new column “subtol2” as the sum of review_aroma and review_appearance into d2:

```
subtol2 <- d2$review_aroma+d2$review_appearance
d2$subtol2 <- subtol2
```

Find the average of “subtol2” group by beer style and get the one with maximum subtotal of review_aroma and review_appearance :

```
ff <- aggregate(d2$subtol2, list(d2$beer_style), mean,na.omit=T)
gg <- ff[ff$x==max(ff$x),]
gg
```

```
##                           Group.1        x
## 12 American Double / Imperial Stout 8.325858
```

```
dim(d2[d2$beer_style==gg$Group.1,])[1]
```

```
## [1] 50146
```

Due to the large amount of reviews for American Double / Imperial Stout, the finding is convincing.

American Double / Imperial Stout is recommended if you typically enjoy a beer due to its aroma and appearance.