pizza restaurants analysis

Sowmya Chavali

02/04/2021

Pizza! It's one of America's most beloved dishes, eaten anywhere, anytime, and by everyone. The objective of this project is to find excellent pizza based on the location, review and price, and finally build a pizza-map focused on New York, which is known as the "Pizza Capital of the U.S."

I am using the data sources from Jared, Barstool, and Datafiniti. Jared's data is from top NY pizza restaurants, with a 6-point Likert scale survey on ratings. The Barstool sports dataset has critic, public, and the Barstool Staff's rating as well as pricing, location, and geo-location. Datafiniti includes 10000 pizza places, their price ranges and geo-locations.

```
library(readr)
library(dplyr)
library(tidyverse)
library(ggplot2)
library(ggmap)
library(car)
library(plotly)
library(bootstrap)
library(cluster)
library(factoextra)
library(gridExtra)
library(leaflet)
library(DataExplorer)
```

```
barstool <- readr::read_csv("https://raw.githubusercontent.com/rfordatascience/tidytuesday/master/data/
datafiniti <- readr::read_csv("https://raw.githubusercontent.com/rfordatascience/tidytuesday/master/data
jared <- readr::read_csv("https://raw.githubusercontent.com/rfordatascience/tidytuesday/master/data/201
```

Importing Data The purpose of the data is to find the best pizza restaurant, focusing New York City. To achieve that Tyler Ricards recorded the web traffic coming through the OneBite application.

```
dim(barstool)
```

Source data breakdown

```
## [1] 463  22
```

```
dim(datafiniti)
```

```
## [1] 10000    10
```

```
dim(jared)
```

```
## [1] 375   9
```

Identifying missing values

```
colSums(is.na(barstool))  # 2 missing observations - latitude and longitude
```

```
##                                 name                             address1
##                                    0                                    0
##                                 city                                  zip
##                                    0                                    0
##                              country                             latitude
##                                    0                                    2
##                            longitude                          price_level
##                                    2                                    0
##                      provider_rating                provider_review_count
##                                    0                                    0
##       review_stats_all_average_score                review_stats_all_count
##                                    0                                    0
##         review_stats_all_total_score review_stats_community_average_score
##                                    0                                    0
##         review_stats_community_count   review_stats_community_total_score
##                                    0                                    0
##    review_stats_critic_average_score            review_stats_critic_count
##                                    0                                    0
##      review_stats_critic_total_score      review_stats_dave_average_score
##                                    0                                    0
##              review_stats_dave_count        review_stats_dave_total_score
##                                    0                                    0
```

```
unique_datafiniti <- datafiniti %>% distinct()
colSums(is.na(unique_datafiniti)) # No missing observations
```

```
##            name         address            city         country        province
##               0               0               0               0               0
##        latitude       longitude      categories price_range_min price_range_max
##               0               0               0               0               0
```

```
colSums(is.na(jared))   # 5 missing observations - percent
```

```
##   polla_qid      answer       votes    pollq_id    question       place
##           0           0           0           0           0           0
##        time total_votes     percent
##           0           0           5
```

No of Zero values in each column

```
colSums(barstool == 0)
```

```
##                                  name                                 address1
##                                     0                                        0
##                                  city                                      zip
##                                     0                                        0
##                               country                                 latitude
##                                     0                                       NA
##                             longitude                               price_level
##                                    NA                                       21
##                       provider_rating                     provider_review_count
##                                     0                                        0
##       review_stats_all_average_score                    review_stats_all_count
##                                     0                                        0
##         review_stats_all_total_score      review_stats_community_average_score
##                                     0                                       41
##         review_stats_community_count        review_stats_community_total_score
##                                    41                                       41
##    review_stats_critic_average_score                 review_stats_critic_count
##                                   401                                      401
##      review_stats_critic_total_score           review_stats_dave_average_score
##                                   401                                        0
##              review_stats_dave_count             review_stats_dave_total_score
##                                     0                                        0
```

```
colSums(unique_datafiniti == 0)
```

```
##            name         address            city         country        province
##               0               0               0               0               0
##        latitude       longitude      categories price_range_min price_range_max
##               0               0               0            1852               0
```

```
colSums(jared == 0)
```

```
##   polla_qid      answer       votes    pollq_id    question       place
##           0           0         104           0           0           0
##        time total_votes     percent
##           0           5          NA
```

Summary of each dataset

```
summary(barstool)
```

```
##      name             address1            city                zip
##  Length:463         Length:463         Length:463         Min.   : 1748
##  Class :character   Class :character   Class :character   1st Qu.:10009
##  Mode  :character   Mode  :character   Mode  :character   Median :10019
##                                                           Mean   :18531
##                                                           3rd Qu.:11234
##                                                           Max.   :94133
##
```

```
##    country             latitude       longitude        price_level
##  Length:463         Min.   :25.79   Min.   :-122.41   Min.   :0.00
##  Class :character   1st Qu.:40.72   1st Qu.: -74.09   1st Qu.:1.00
##  Mode  :character   Median :40.75   Median : -73.99   Median :1.00
##                     Mean   :40.19   Mean   : -77.44   Mean   :1.46
##                     3rd Qu.:40.78   3rd Qu.: -73.97   3rd Qu.:2.00
##                     Max.   :45.00   Max.   : -70.09   Max.   :3.00
##                     NA's   :2       NA's   :2
##  provider_rating provider_review_count review_stats_all_average_score
##  Min.   :2.000   Min.   :   2.0        Min.   :0.100
##  1st Qu.:3.500   1st Qu.:  74.0        1st Qu.:6.240
##  Median :3.500   Median : 169.0        Median :7.162
##  Mean   :3.671   Mean   : 386.1        Mean   :6.876
##  3rd Qu.:4.000   3rd Qu.: 392.0        3rd Qu.:7.809
##  Max.   :5.000   Max.   :5797.0        Max.   :9.079
##
##  review_stats_all_count review_stats_all_total_score
##  Min.   :  1.00         Min.   :   0.10
##  1st Qu.:  4.00         1st Qu.:  23.65
##  Median :  8.00         Median :  54.10
##  Mean   : 19.02         Mean   : 149.93
##  3rd Qu.: 19.00         3rd Qu.: 140.20
##  Max.   :568.00         Max.   :5045.60
##
##  review_stats_community_average_score review_stats_community_count
##  Min.   : 0.000                       Min.   :  0.00
##  1st Qu.: 6.075                       1st Qu.:  3.00
##  Median : 7.225                       Median :  7.00
##  Mean   : 6.457                       Mean   : 17.87
##  3rd Qu.: 7.873                       3rd Qu.: 18.00
##  Max.   :10.000                       Max.   :567.00
##
##  review_stats_community_total_score review_stats_critic_average_score
##  Min.   :   0.00                    Min.   : 0.0000
##  1st Qu.:  15.65                    1st Qu.: 0.0000
##  Median :  47.30                    Median : 0.0000
##  Mean   : 142.28                    Mean   : 0.9717
##  3rd Qu.: 135.10                    3rd Qu.: 0.0000
##  Max.   :5036.30                    Max.   :11.0000
##
##  review_stats_critic_count review_stats_critic_total_score
##  Min.   :0.0000            Min.   : 0.000
##  1st Qu.:0.0000            1st Qu.: 0.000
##  Median :0.0000            Median : 0.000
##  Mean   :0.1425            Mean   : 1.023
##  3rd Qu.:0.0000            3rd Qu.: 0.000
##  Max.   :5.0000            Max.   :29.800
##
##  review_stats_dave_average_score review_stats_dave_count
##  Min.   : 0.080                  Min.   :1
##  1st Qu.: 6.200                  1st Qu.:1
##  Median : 7.100                  Median :1
##  Mean   : 6.623                  Mean   :1
##  3rd Qu.: 7.800                  3rd Qu.:1
```

```
##  Max.   :10.000                  Max.   :1
##
##  review_stats_dave_total_score
##  Min.   : 0.080
##  1st Qu.: 6.200
##  Median : 7.100
##  Mean   : 6.623
##  3rd Qu.: 7.800
##  Max.   :10.000
##
```

```
summary(datafiniti)
```

```
##      name              address              city              country
##  Length:10000       Length:10000       Length:10000       Length:10000
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##    province            latitude       longitude        categories
##  Length:10000       Min.   :21.42   Min.   :-157.80   Length:10000
##  Class :character   1st Qu.:34.42   1st Qu.:-104.80   Class :character
##  Mode  :character   Median :40.12   Median : -82.91   Mode  :character
##                     Mean   :38.37   Mean   : -90.06
##                     3rd Qu.:40.91   3rd Qu.: -75.19
##                     Max.   :64.85   Max.   : -71.95
##  price_range_min  price_range_max
##  Min.   : 0.000   Min.   : 7.00
##  1st Qu.: 0.000   1st Qu.:25.00
##  Median : 0.000   Median :25.00
##  Mean   : 4.655   Mean   :27.76
##  3rd Qu.: 0.000   3rd Qu.:25.00
##  Max.   :50.000   Max.   :55.00
```

```
summary(jared)
```

```
##    polla_qid        answer              votes           pollq_id
##  Min.   : 2.00   Length:375         Min.   : 0.000   Min.   : 2.00
##  1st Qu.:21.00   Class :character   1st Qu.: 0.000   1st Qu.:21.00
##  Median :40.00   Mode  :character   Median : 2.000   Median :40.00
##  Mean   :39.93                      Mean   : 2.832   Mean   :39.93
##  3rd Qu.:59.00                      3rd Qu.: 4.000   3rd Qu.:59.00
##  Max.   :77.00                      Max.   :26.000   Max.   :77.00
##
##    question            place                time            total_votes
##  Length:375         Length:375         Min.   :1.344e+09   Min.   : 0.00
##  Class :character   Class :character   1st Qu.:1.395e+09   1st Qu.: 7.00
##  Mode  :character   Mode  :character   Median :1.467e+09   Median :12.00
##                                        Mean   :1.459e+09   Mean   :14.16
##                                        3rd Qu.:1.519e+09   3rd Qu.:19.00
##                                        Max.   :1.569e+09   Max.   :67.00
##
```

```
##     percent
##  Min.   :0.0000
##  1st Qu.:0.0000
##  Median :0.1667
##  Mean   :0.2000
##  3rd Qu.:0.3333
##  Max.   :1.0000
##  NA's   :5
```

```
table(barstool$price_level)
```

```
##
##   0   1   2   3
##  21 216 218   8
```

0 and 3 price level have few observations

Data Cleaning Generally we employ data cleaning steps to derive relevant insights from the data and to get rid of garbage values There are only two missing values in the latitude and longitude column in the Barstool data. Also, there is a value out of range in Dave's rating.No imputation is being performed as the count is less than 5% of the data in both cases.Critic ratings have 401 zeroes, Community ratings have 41 zeroes, Minimum price range contains 1852 zeroes, Votes have 104 zeroes. Apart from this the data is clean.

Cleaning Datafiniti

Lets create Category column by clubbing similar categories

```
head(unique_datafiniti)
```

```
## # A tibble: 6 x 10
##   name  address city  country province latitude longitude categories
##   <chr> <chr>   <chr> <chr>   <chr>       <dbl>     <dbl> <chr>
## 1 Shot~ 4203 E~ Sher~ US      AR           34.8     -92.2 Pizza,Res~
## 2 Sauc~ 25 E C~ Phoe~ US      AZ           33.5    -112.  Pizza,Piz~
## 3 Mios~ 3703 P~ Cinc~ US      OH           39.1     -84.4 Restauran~
## 4 Hung~ 30495 ~ Madi~ US      MI           42.5     -83.1 Pizza,Car~
## 5 Spar~ 3600 E~ Balt~ US      MD           39.3     -76.6 Pizza,Ame~
## 6 La V~ 1834 E~ Berk~ US      CA           37.9    -122.  Pizza Pla~
## # ... with 2 more variables: price_range_min <dbl>, price_range_max <dbl>
```

```
unique_datafiniti$categories <- toupper(unique_datafiniti$categories)
new_datafiniti <- unique_datafiniti %>%
mutate(category = case_when(str_detect(categories,"BAR|BREW|PUB|CLUB|LOUNGE") ~ 'ALCOHOL SERVING', str_
```

Checking the category column

```
table(new_datafiniti$category)
```

```
##
##         ALCOHOL SERVING                CATERERS                 ITALIAN
##                      63                      61                     357
## NORMAL PIZZA RESTAURANT
##                    1804
```

Lets clean jared data

```
dim(jared)
```

```
## [1] 375   9
```

```
# Removing rows with 0 total votes
jared_rmzero <- jared%>%
               filter(total_votes != 0)
#Checking new data

dim(jared_rmzero)
```

```
## [1] 370   9
```

```
DT::datatable(barstool)
```

```
## PhantomJS not found. You can install it with webshot::install_phantomjs(). If it is installed, pleas
```

```
DT::datatable(unique_datafiniti)
```

```
DT::datatable(jared_rmzero)
```

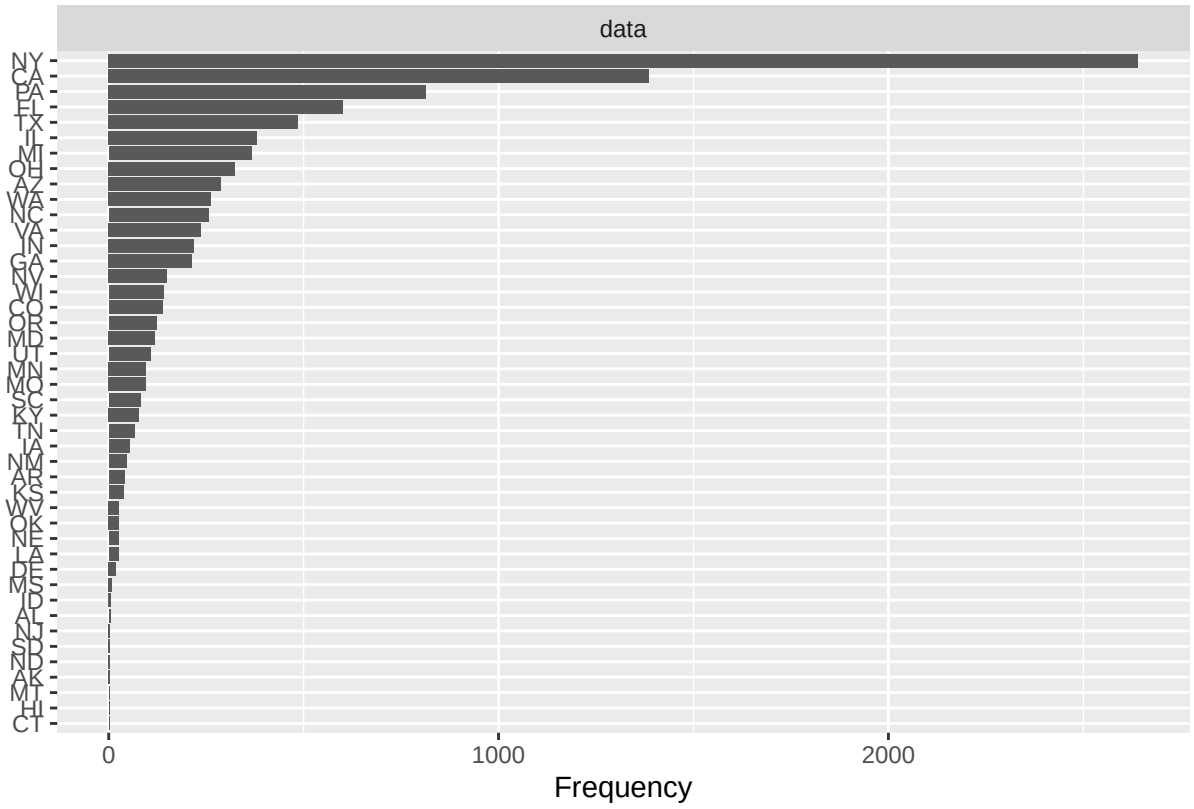
Converting answer to Numerical Rating

```
jared_rmzero <- jared_rmzero %>%
  mutate(Numerical_Rating = case_when(
    answer=="Never Again" ~ 0,
    answer=="Poor" ~ 2,
    answer=="Fair" ~ 4,
    answer=="Average" ~ 6,
    answer=="Good"~ 8,
    answer=="Excellent" ~ 10))

# Calculating weighted numerical rating
jared_ratings <- jared_rmzero %>%
  mutate(Weighted_Rating = Numerical_Rating*votes) %>%
    group_by(place) %>%
    summarise(Final_Rating = sum(Weighted_Rating)/sum(votes))

# Looking at the final Jared Ratings
head(jared_ratings)
```

```
## # A tibble: 6 x 2
##   place                       Final_Rating
##   <chr>                              <dbl>
## 1 5 Boroughs Pizza                    7.33
## 2 Artichoke Basille's Pizza           8
## 3 Arturo's                            7.43
## 4 Bella Napoli                        7.07
## 5 Ben's of SoHo 14th Street           4.8
## 6 Ben's of SoHo Spring Street         6.44
```

EDA I have applied DataExplorer package as a fast and efficient way to do typical basic EDA. Through the Plot Bar chart of Datafiniti, we found that New York is the dominant state (province) in the dataset.

```
plot_bar(datafiniti$province)
```



Correlation between various pizza ratings

```
barstool_2 <- barstool %>%
  rename(
    all_score = review_stats_all_average_score,
    community_score = review_stats_community_average_score,
    critic_score = review_stats_critic_average_score,
    dave_score = review_stats_dave_average_score
    )

data <- barstool_2 %>% select(provider_rating,community_score,critic_score,dave_score)
data2 <- data[data$critic_score != 0 & data$community_score != 0,]
cor(data2)
```

```
##                 provider_rating community_score critic_score dave_score
## provider_rating      1.00000000       0.4339490    0.2017888 0.05343971
## community_score      0.43394896       1.0000000    0.1832775 0.34432172
## critic_score         0.20178876       0.1832775    1.0000000 0.41823049
## dave_score           0.05343971       0.3443217    0.4182305 1.00000000
```

Correlation between critic score and dave score is 0.42

```
data3 <- data[data$community_score != 0,]
cor(data3)
```

```
##                 provider_rating community_score critic_score  dave_score
## provider_rating      1.00000000      0.31921901  -0.07935913  0.22096952
## community_score      0.31921901      1.00000000  -0.05570681  0.60522594
## critic_score        -0.07935913     -0.05570681   1.00000000 -0.04922468
## dave_score           0.22096952      0.60522594  -0.04922468  1.00000000
```

Correlation between dave score and community score is 0.6, between provider_rating and community score is 0.32 and correlation between provider_rating and dave score is 0.22

Joining Barstool and jared

```
barstool_jared<- jared_ratings %>%
  inner_join(barstool, by = c("place" = "name"))
```

Finding correlation between Jared Final Rating and Barstool All Average Rating

```
cor(barstool_jared$Final_Rating,barstool_jared$review_stats_all_average_score) ## The correlation is no
```

```
## [1] 0.3026819
```

```
Newyork_Barstool <- barstool[str_detect(barstool$city,"York"),]
Rest_of_US_Barstool <-barstool[!str_detect(barstool$city,"York"),]
mean(Newyork_Barstool$review_stats_all_average_score)
```

Comparing pizza ratings in New York with rest of the US

```
## [1] 6.64562
```

```
mean(Rest_of_US_Barstool$review_stats_all_average_score)
```

```
## [1] 7.15211
```

```
mean(Newyork_Barstool$provider_rating)
```

```
## [1] 3.605159
```

```
mean(Rest_of_US_Barstool$provider_rating)
```

```
## [1] 3.748815
```

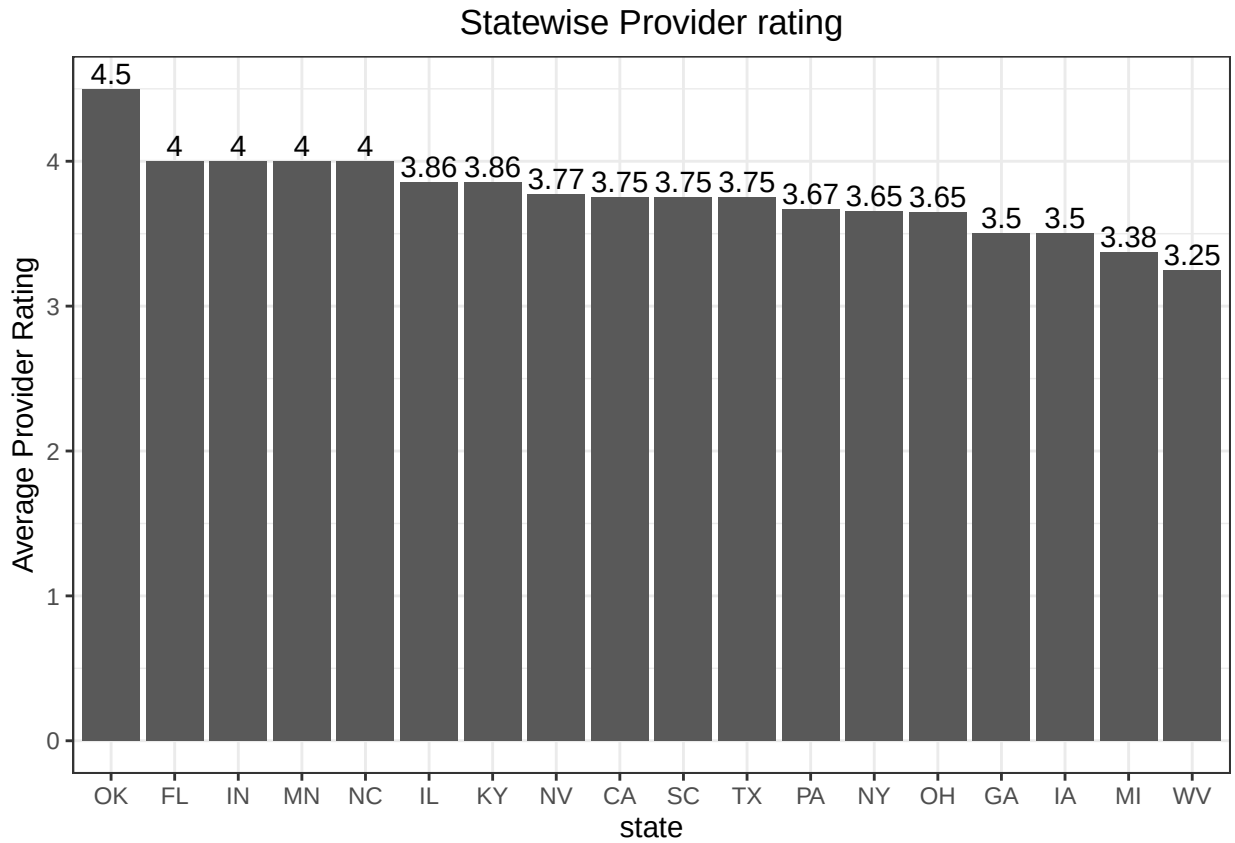
New York has slightly lower provider and average ratings on average as compared to the rest of US

```
table((barstool %>% left_join(new_datafiniti%>% distinct(city,province),by = "city"))$province)     ## A
```

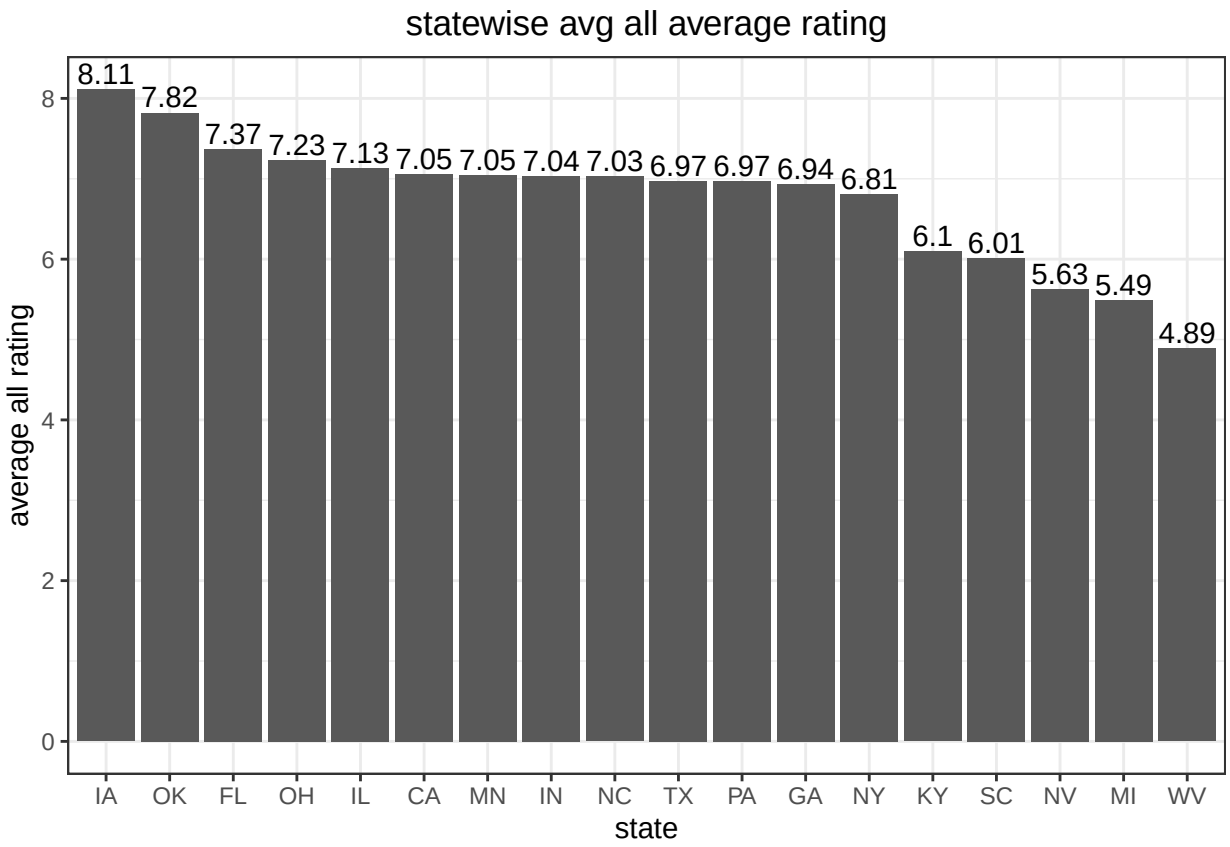
Comparing pizza ratings across states

```
##
##  CA  FL  GA  IA  IL  IN  KY  MI  MN  NC  NV  NY  OH  OK  PA  SC  TX  WV
##   8   6   9   3   7   4   7   4  10   1  11 290  10   1   3   4   4   4
```

```
table1 <- barstool %>% left_join(new_datafiniti%>% distinct(city,province),by = "city") %>% group_by(pr

table1 = na.omit(table1)

## Plotting state wise average provider ratings

ggplot(data = table1, aes(x = reorder(province, -Avg_provider_rating), y = Avg_provider_rating)) +
  geom_bar(stat = "identity") +
    xlab("state") + ylab("Average Provider Rating") +
    ggtitle("Statewise Provider rating") +
    theme_bw() + geom_text(aes(label=round(Avg_provider_rating,2)), position=position_dodge(width=0.9),
  theme(plot.title = element_text(hjust = 0.5))
```



```
table2 <- barstool %>% left_join(new_datafiniti%>% distinct(city,province),by = "city") %>% group_by(pr

table2 = na.omit(table2)

## Plotting state wise All average ratings

ggplot(data = table2, aes(x = reorder(province, -Avg_All_Rating), y = Avg_All_Rating)) +
  geom_bar(stat = "identity",
           size=.2) +
    xlab("state") + ylab("average all rating") +
    ggtitle("statewise avg all average rating") +
    theme_bw() + geom_text(aes(label=round(Avg_All_Rating,2)), position=position_dodge(width=0.9), vjus
  theme(plot.title = element_text(hjust = 0.5))
```



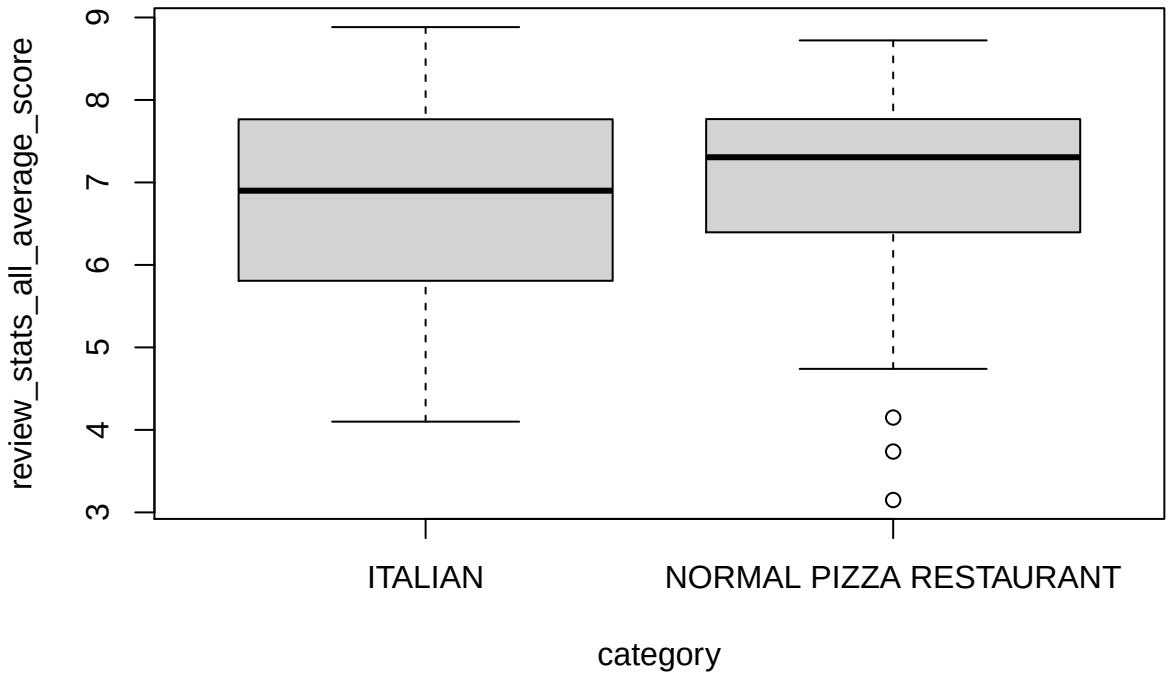
Comparing ratings across categories Joining Datafinity and Barstool data

```
datafiniti_barstool<- new_datafiniti %>%
  inner_join(barstool, by = "name", "city")

dim(datafiniti_barstool)
```

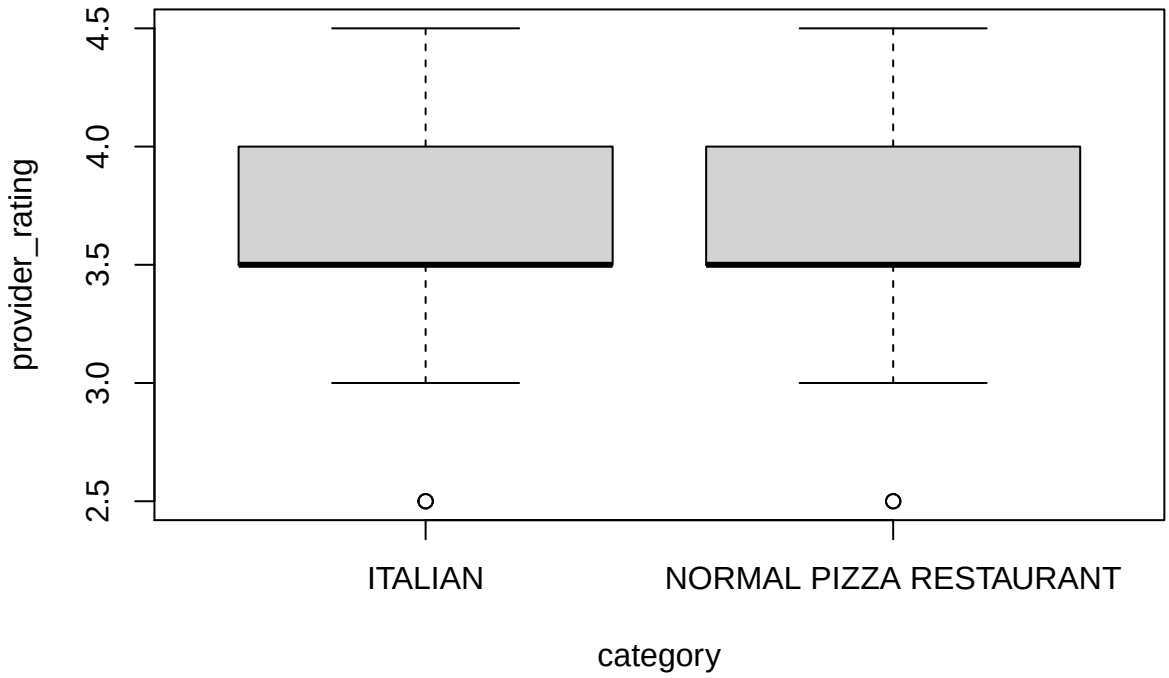
```
## [1] 94 32
```

Analysing ratings across pizza categories

```
boxplot(review_stats_all_average_score~category, data = datafiniti_barstool)
```




```
boxplot(provider_rating~category, data = datafiniti_barstool)
```



Normal Pizza Restaurants have slightly higher All average score as compared to those restaurants which serve Italian pizza, however provider_rating has very similar distribution across both the categories

```
new_datafiniti %>% group_by(category) %>% summarise(AVERAGE_MAX_PRICE = mean(price_range_max))
```

Comparing price range across pizza categories

```
## # A tibble: 4 x 2
##   category                AVERAGE_MAX_PRICE
## * <chr>                               <dbl>
## 1 ALCOHOL SERVING                      32.1
## 2 CATERERS                             27.9
## 3 ITALIAN                              30.9
## 4 NORMAL PIZZA RESTAURANT              27.0
```

```
new_datafiniti[new_datafiniti$price_range_min != 0,] %>% group_by(category) %>% summarise(AVERAGE_MIN_P
```

```
## # A tibble: 4 x 2
##   category                AVERAGE_MIN_PRICE
## * <chr>                               <dbl>
## 1 ALCOHOL SERVING                      26.4
## 2 CATERERS                             23.2
## 3 ITALIAN                              25.4
## 4 NORMAL PIZZA RESTAURANT              24.1
```

Alcohol serving pizza restaurants have the highest average min and max price range followed by Italian pizza restaurants. Caterers and Normal Pizza restaurants have similar min and max price range.

Do higher priced restaurants have better ratings?

Lets analyze provider ratings

```
price_low <- barstool[(barstool$price_level  == 1) | (barstool$price_level  == 0),]
price_high <- barstool[(barstool$price_level  == 2) | (barstool$price_level  == 3),]

mean_high <- mean(price_high$provider_rating)
mean_high
```

```
## [1] 3.710177
```

```
mean_low <- mean(price_low$provider_rating)
mean_low
```

```
## [1] 3.632911
```

```
nrow_high <- nrow(price_high)
nrow_high
```

```
## [1] 226
```

```
nrow_low <- nrow(price_low)
nrow_low
```

```
## [1] 237
```

NULL HYPOTHESIS: mean_high - mean_low <= 0 ALTERNATE HYPOTHESIS mean_high - mean_low > 0

```
se = sqrt(var(price_high$provider_rating)/nrow_high + var(price_low$provider_rating)/nrow_low)
se
```

```
## [1] 0.04753894
```

```
Z = (mean_high-mean_low)/se
Z
```

```
## [1] 1.625312
```

```
Zalpha = qnorm(0.90)
Zalpha
```

```
## [1] 1.281552
```

Z > Zalpha We can reject the NULL HYPOTHESIS with 90% confidence. Hence we can say higher priced restaurants have better mean provider_ratings as compared to lower priced restaurants with 90% confidence

Analyzing All Average Score

```r
u1 <- mean(price_high$review_stats_all_average_score)
u1
```

```
## [1] 7.200656
```

```r
u2 <- mean(price_low$review_stats_all_average_score)
u2
```

```
## [1] 6.567271
```

NULL HYPOTHESIS: u1 - u2 <= 0 ALTERNATE HYPOTHESIS u1 - u2 > 0

```r
se2 = sqrt(var(price_high$review_stats_all_average_score)/nrow_high + var(price_low$review_stats_all_av
se2
```

```
## [1] 0.1288524
```

```r
Z2 = (u1-u2)/se2
Z2
```

```
## [1] 4.915585
```

```r
Zalpha2 = qnorm(0.99)
Zalpha2
```

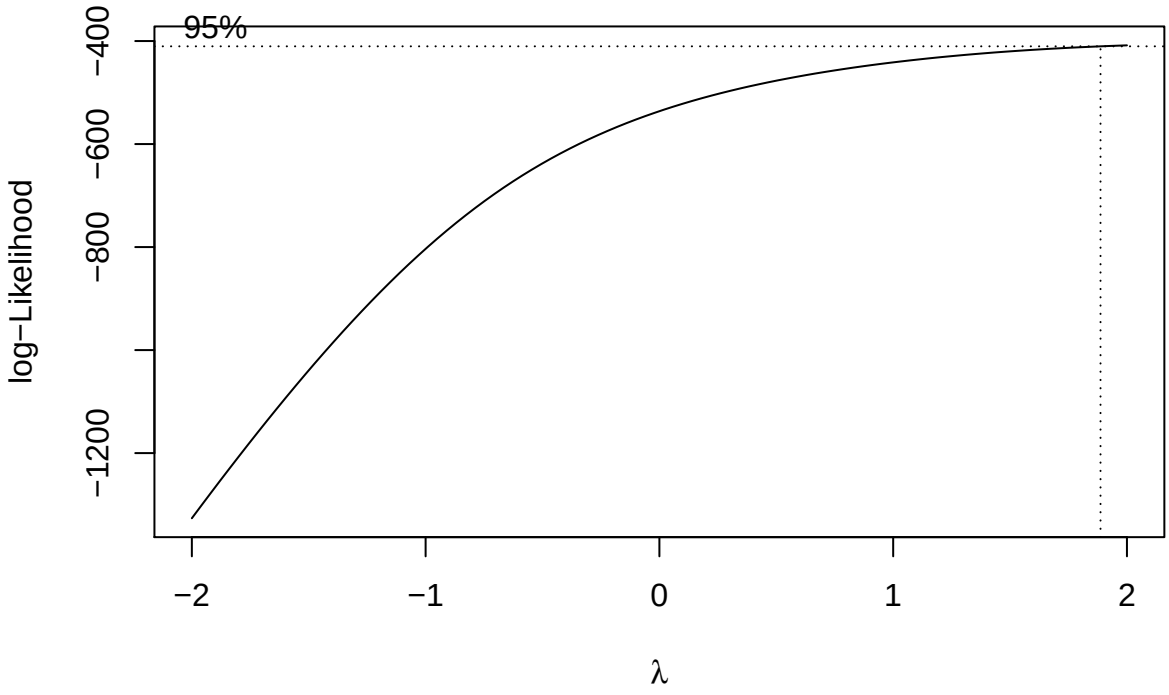
```
## [1] 2.326348
```

Z2 > Zalpha2

We can reject the NULL HYPOTHESIS with 99% confidence. Hence we can say higher priced restaurants have better mean all_average_score as compared to lower priced restaurants with 99% confidence.

Regression for predicting community ratings - OneBite user ratings

```r
# BOXCOX TRANSFORMATION TO GET LAMBDA
boxcox <- MASS::boxcox(community_score ~ dave_score + provider_rating, data = data3)
```




```
lambda <- boxcox$x[which.max(boxcox$y)]
lambda
```

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

```
data3$community_score2 <- ((data3$community_score ^ lambda) - 1) / lambda

## POLYNOMIAL REGRESSION FOR PREDICTING COMMUNITY RATINGS

fit <- lm(community_score2 ~ poly(dave_score,2) + poly(provider_rating,5), data = data3)
```

Adjusted R square of the model is 0.46. Since we are predicting consumer ratings which has very high variation, we can accept this R square value. Also, p value associated with F test and most of the individual t tests in not significant and we can reject the NULL hypothesis at 95% confidence level.

```
summary(fit)
```

```
##
## Call:
## lm(formula = community_score2 ~ poly(dave_score, 2) + poly(provider_rating,
##     5), data = data3)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
```

```
## -24.4405  -2.6747   0.4688   3.3023  23.2799
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 25.3705     0.2885  87.942  < 2e-16 ***
## poly(dave_score, 2)1        93.4312     6.0850  15.354  < 2e-16 ***
## poly(dave_score, 2)2        33.6138     5.9400   5.659 2.85e-08 ***
## poly(provider_rating, 5)1  31.7952     6.0854   5.225 2.77e-07 ***
## poly(provider_rating, 5)2   7.3717     5.9304   1.243   0.2146
## poly(provider_rating, 5)3  -9.4294     5.9264  -1.591   0.1124
## poly(provider_rating, 5)4  12.9945     5.9323   2.190   0.0290 *
## poly(provider_rating, 5)5 -13.2177     5.9295  -2.229   0.0263 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.926 on 414 degrees of freedom
## Multiple R-squared:  0.4633, Adjusted R-squared:  0.4543
## F-statistic: 51.06 on 7 and 414 DF,  p-value: < 2.2e-16
```

Regression equation

$$community\_score2 = \frac{community\_score^{\lambda} - 1}{\lambda}$$

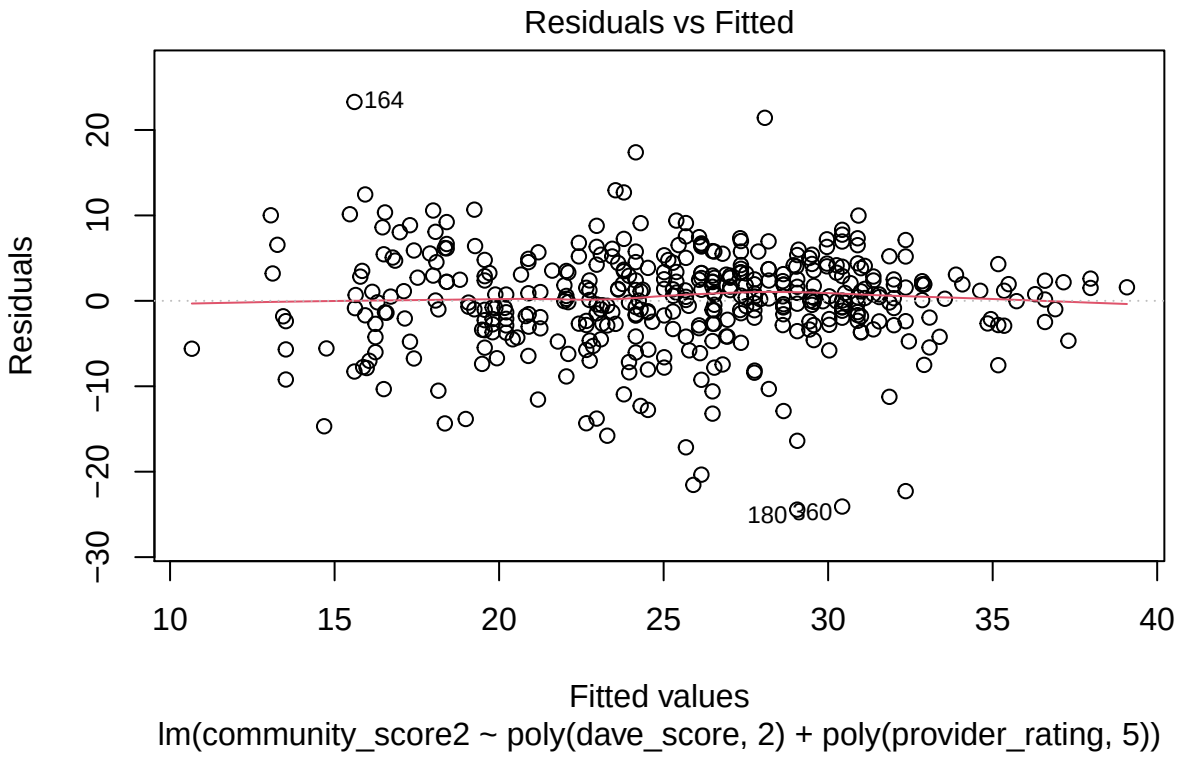
$community\_score2 = 25.3705 + 93.4312 * dave\_score + 33.6138 * dave\_score^2 + 31.7952 * provider\_rating + 7.3717 * provider\_r$

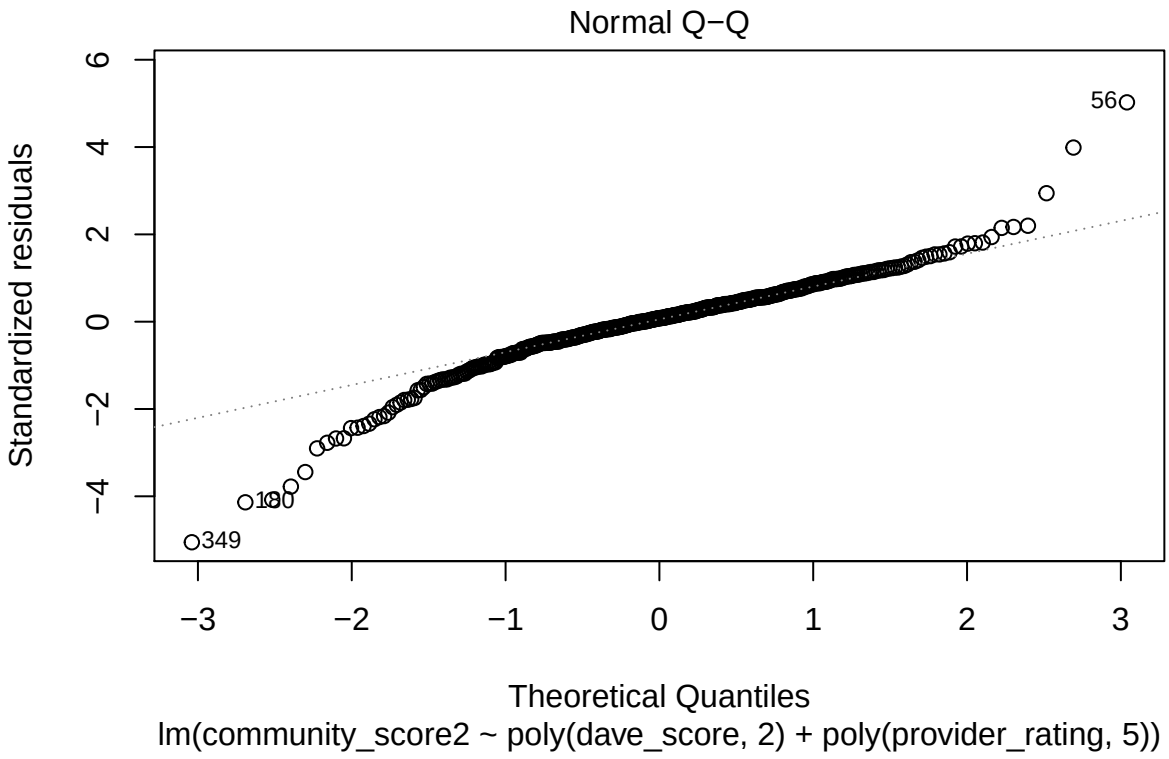
Plotting and analyzing the residuals

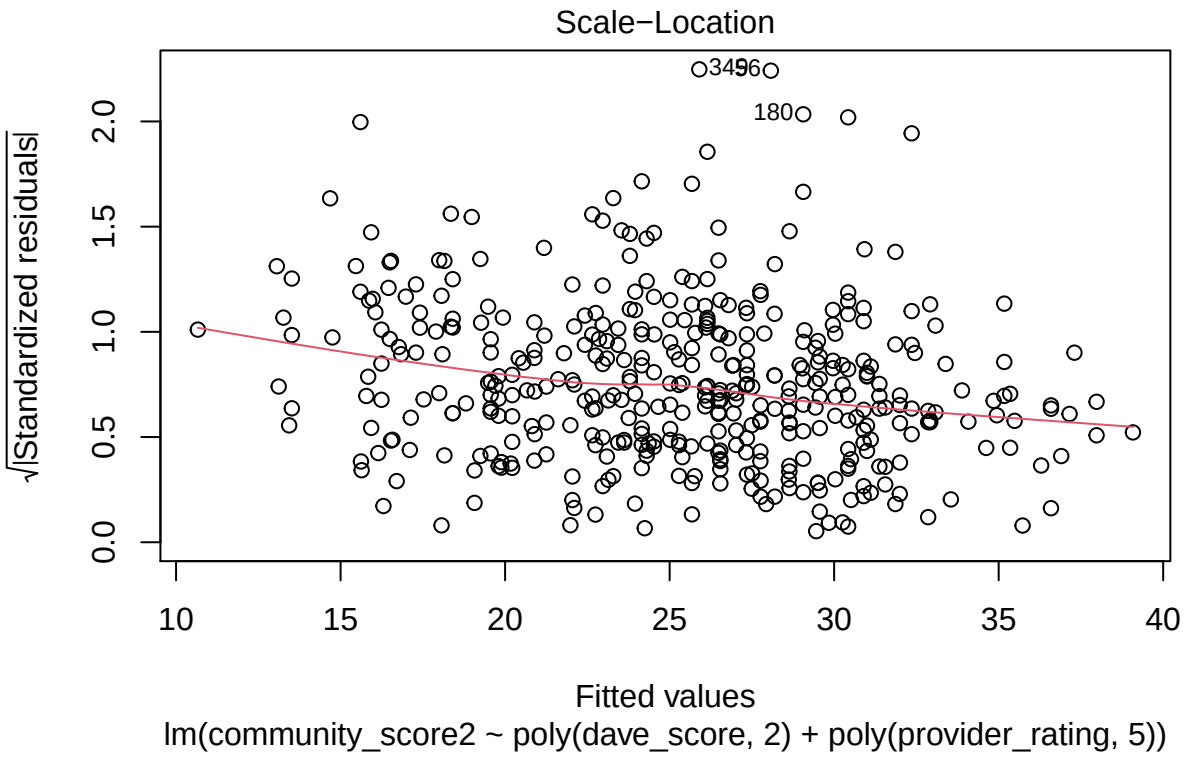
By performing residual diagnostic, we can see that they satisfy our initial regression assumptions of-

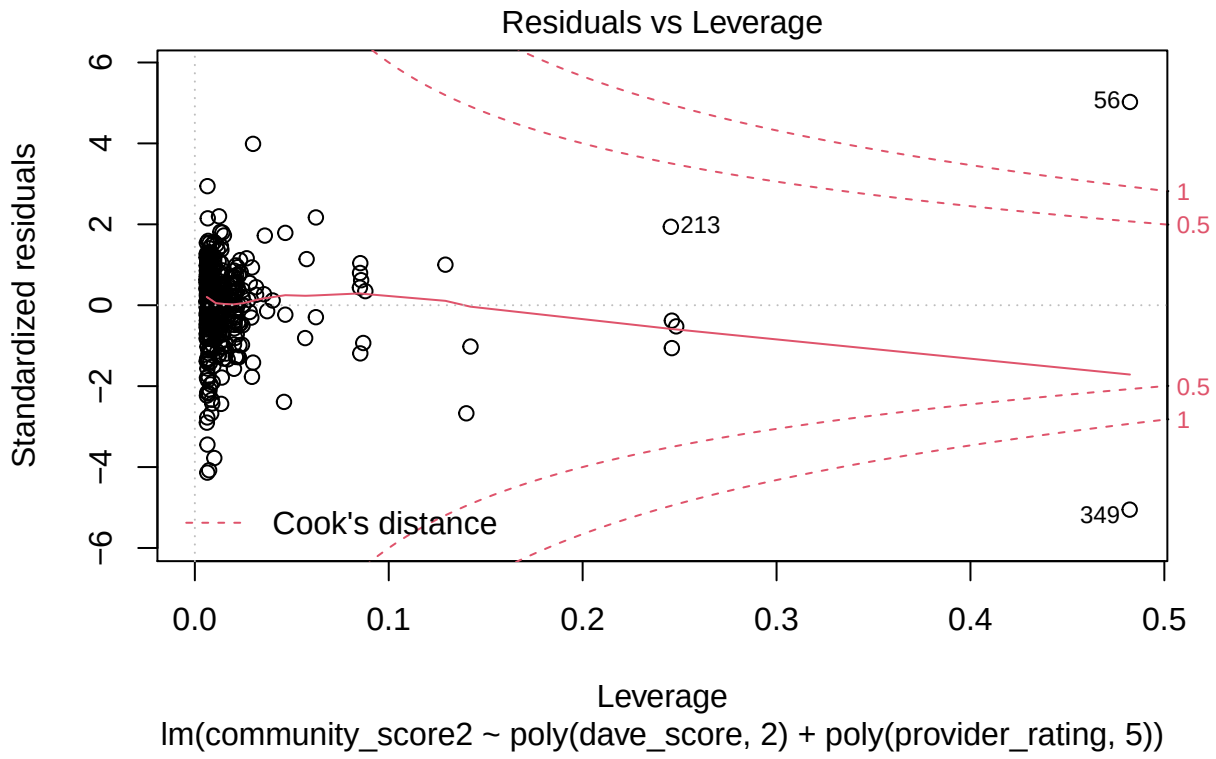
- Normality
- Constant Variance
- Mean 0

```
plot(fit)
```


Residuals vs Fitted
Residuals
Fitted values
lm(community_score2 ~ poly(dave_score, 2) + poly(provider_rating, 5))
164
180
360
20
10
0
−10
−20
−30
10
15
20
25
30
35
40


Normal Q−Q
Standardized residuals
6
4
2
0
−2
−4
56
180
349
−3
−2
−1
0
1
2
3
Theoretical Quantiles
lm(community_score2 ~ poly(dave_score, 2) + poly(provider_rating, 5))


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
10
15
20
25
30
35
40
Fitted values
lm(community_score2 ~ poly(dave_score, 2) + poly(provider_rating, 5))



Clustering

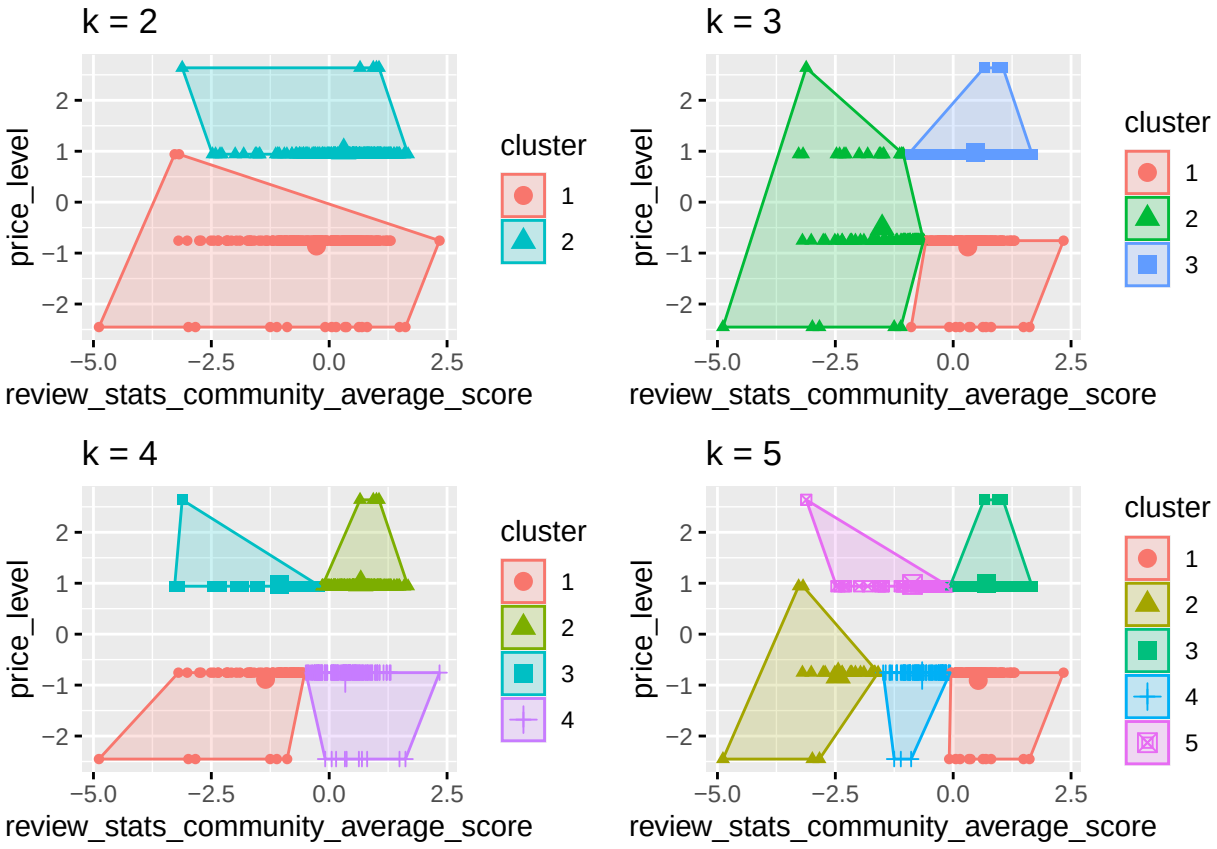
We are going to cluster pizza restaurants on the basis of price level and community ratings

```
#Removing zeroes
barstool_rm <- barstool[barstool$review_stats_community_average_score!=0,]
#Scaling data
barstool_cl <- scale(barstool_rm[c("review_stats_community_average_score","price_level")])

## Creating multiple clusters with different centres

set.seed(5021)

k2 <- kmeans(barstool_cl, centers = 2, nstart = 25)
k3 <- kmeans(barstool_cl, centers = 3, nstart = 25)
k4 <- kmeans(barstool_cl, centers = 4, nstart = 25)
k5 <- kmeans(barstool_cl, centers = 5, nstart = 25)

str(k4)
```

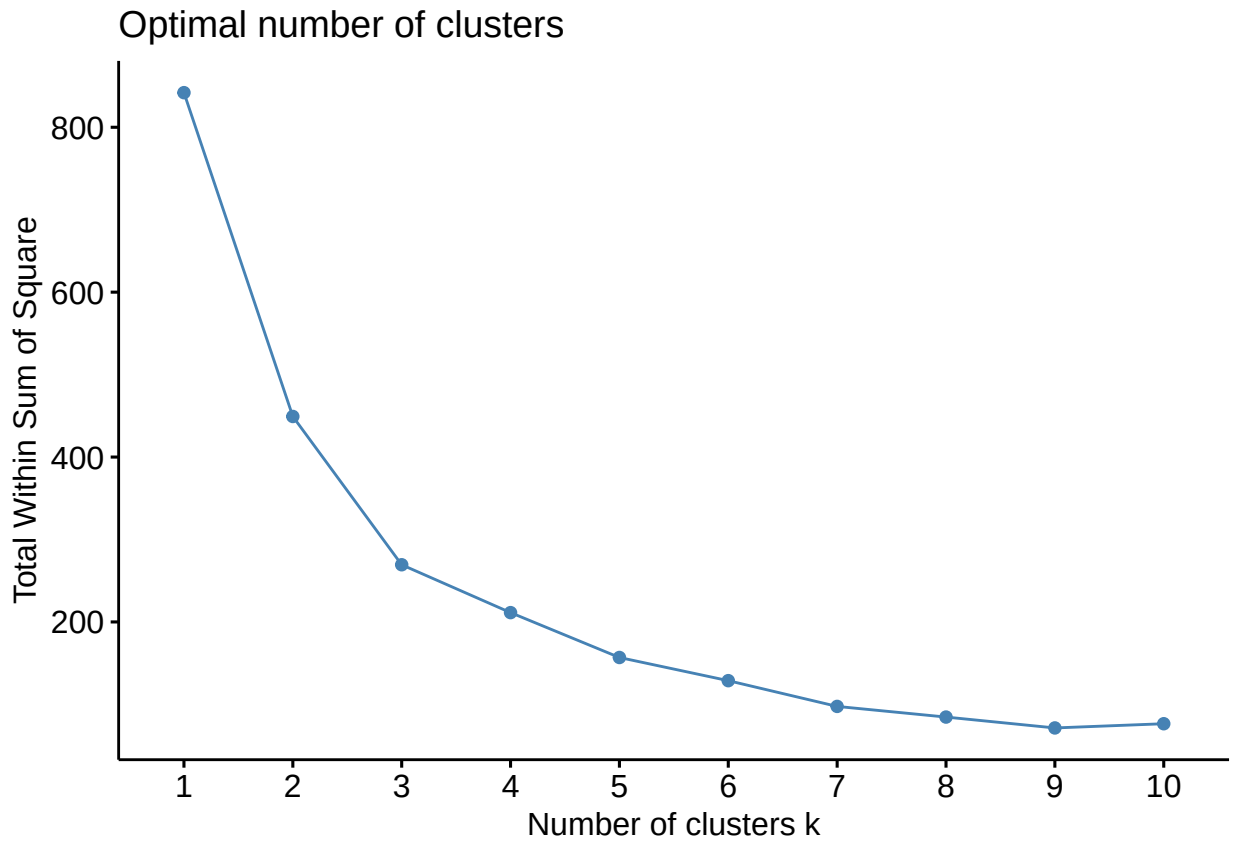
```
## List of 9
##  $ cluster     : int [1:422] 4 4 1 1 1 4 2 3 1 1 ...
##  $ centers     : num [1:4, 1:2] -1.351 0.669 -1.046 0.341 -0.888 ...
##   ..- attr(*, "dimnames")=List of 2
##   .. ..$ : chr [1:4] "1" "2" "3" "4"
##   .. ..$ : chr [1:2] "review_stats_community_average_score" "price_level"
##  $ totss       : num 842
```

```
##  $ withinss    : num [1:4] 64.8 40.3 35.6 60.7
##  $ tot.withinss: num 202
##  $ betweenss   : num 640
##  $ size        : int [1:4] 77 153 46 146
##  $ iter        : int 3
##  $ ifault      : int 0
##  - attr(*, "class")= chr "kmeans"
```

```r
# Visualizing the clusters

p2 <- fviz_cluster(k2, geom = "point", data = barstool_cl) + ggtitle("k = 2")
p3 <- fviz_cluster(k3, geom = "point", data = barstool_cl) + ggtitle("k = 3")
p4 <- fviz_cluster(k4, geom = "point", data = barstool_cl) + ggtitle("k = 4")
p5 <- fviz_cluster(k5, geom = "point", data = barstool_cl) + ggtitle("k = 5")

grid.arrange(p2, p3, p4, p5, nrow = 2)
```




```r
#Elbow Curve to decide the optimum number of clusters looking at the bend

fviz_nbclust(barstool_cl, kmeans, method = "wss")
```




```
# Comparing the clusters

barstool_rm %>%
  select("review_stats_community_average_score","price_level") %>%
  mutate(Cluster = k4$cluster) %>%
  group_by(Cluster) %>%
  summarise_all("mean")
```

```
## # A tibble: 4 x 3
##   Cluster review_stats_community_average_score price_level
## *   <int>                                <dbl>       <dbl>
## 1       1                                 5.40       0.922
## 2       2                                 7.92       2.03
## 3       3                                 5.78       2.02
## 4       4                                 7.51       0.932
```

So finally we have 4 cluster which signify

- Cluster_1 - Low Rating and Low Price restaurants
- Cluster_2 - High Rating and High Price restaurants
- Cluster_3 - Low Rating and High Price restaurants
- Cluster_4 - High Rating and Low Price restaurants

```
#Based on Ratings

barstool_NY <- barstool[barstool$city=='New York',]%>%
  na.omit(barstool_NY)

getColor <- function(barstool_NY) {
    sapply(barstool_NY$review_stats_all_average_score, function(x) {
      if(x <= 4.5) {
        "red"
      } else if(x <= 6.5) {
        "orange"
      } else {
        "green"
      } })
  }

  icons <- awesomeIcons(
    icon = 'ios-close',
    iconColor = 'black',
    library = 'ion',
    markerColor = getColor(barstool_NY)
  )


  leaflet(barstool_NY) %>%
    addTiles() %>%
    addAwesomeMarkers(~longitude, ~latitude, icon=icons, label=~as.character(name))%>%
    addProviderTiles("CartoDB.Positron") %>%
    setView(-73.98, 40.75, zoom = 14)
```

```
  #Based on Clusters

  clustered_data <- cbind(k4$cluster,barstool_rm)%>%
    na.omit(barstool_rm$latitude) %>%
    na.omit(barstool_rm$longitude) %>%
    filter(city=="New York")

clustered_data['cluster'] <- clustered_data['k4$cluster']

dim(clustered_data)
```

Visualization

```
## [1] 216  24
```

```
getColor <- function(clustered_data) {
    sapply(clustered_data$cluster, function(x) {
      if(x == 1) {
        "pink"
      } else if(x == 2) {
```

```
      "green"
    } else if(x == 3) {
      "orange"
    }else {
      "red"
    } })
}

icons <- awesomeIcons(
  icon = 'ios-close',
  iconColor = 'black',
  library = 'ion',
  markerColor = getColor(clustered_data)
)


leaflet(clustered_data) %>%
  addTiles() %>%
  addAwesomeMarkers(~longitude, ~latitude, icon=icons, label=~as.character(name))%>%
  addProviderTiles("CartoDB.Positron") %>%
  setView(-73.98, 40.75, zoom = 12.5)
```

Cluster_1 PINK - Low Rating and Low Price restaurants Cluster_2 GREEN - High Rating and High Price restaurants Cluster_3 ORANGE - Low Rating and High Price restaurants Cluster_4 RED - High Rating and Low Price restaurants

The above exercise helped us understand various trends in pizza ratings. The following is the summary of the analysis:

We find that there is low correlation between commmunity, provider, critic and jared ratings. Community and Dave ratings have moderately high correlation of ~0.6 New York have lower provider and average pizza ratings on average as compared to rest of the US States with high ratings- IA,OK,FL,OH States with low ratings - WV,MI,NV,SC Restaurants serving Italian Pizza have lower ratings on average as compared to Non Italian Pizza restaurants Alcohol serving and Italian pizza restaurants have higher priced pizza as compared to those that do not fall in this category High priced restaurants have better pizza ratings as compared to low priced restaurants