# Yelp Review Classification

Ramiro Romero, 205334455

2022-11-30

## Project Set-up

We start the project by reading in the necessary libraries and reading in the data itself.

```
knitr::opts_chunk$set(echo = TRUE)
library(rjson)
library(ggplot2)
library(zoo)
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
library(tm)
```

```
## Loading required package: NLP
```

```
##
## Attaching package: 'NLP'
```

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

```
library(SnowballC)
library(textcat)

library(caTools)
library(rpart)
library(rpart.plot)
library(randomForest)
```

```
## randomForest 4.7-1.1
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
##
## Attaching package: 'randomForest'

## The following object is masked from 'package:ggplot2':
##
##     margin
```

```
library(caret)
```

```
## Loading required package: lattice
```

```
library(e1071)
library(wordcloud)
```

```
## Loading required package: RColorBrewer
```

```
Sys.setlocale("LC_ALL", "C")
```

```
## [1] "C/C/C/C/C/en_US.UTF-8"
```

```
data <- read.csv("Data_Final")
dim(data)
```

```
## [1] 53845    18
```

## EDA and data reduction

Because our data frame is exceptionally large, containing 53845 observations and 18 variables, we should utilize a subset of the data to save ourselves time in computation and processing.

### row-reduction

```
# data cleaning
data$City <- factor(data$City)

levels(data$City)[14:18] <- "Santa Barbara"

table(data$City)
```

```
##
##      Aliso Viejo      Carpinteria         Cerritos           Goleta
##                1             2557               10             6009
##       Isla Vista      Kings Beach      Los Angeles   Mission Canyon
##             1425                1                3               10
##        Montecito     Port Hueneme      Real Goleta             Reno
##              848                1                4                1
##          Salinas    Santa Barbara      Santa Clara      Santa Maria
##                2            42532               15                3
```

```
## South Lake Tahoe           Sparks       Summerland          Truckee
##                1                2              406               13
##        West Hill
##                1
```

To reduce the number of observations in our data, we can filter out observations outside the city of Santa Barbara.

```
data <- data[data$City == "Santa Barbara",]
dim(data)
```

```
## [1] 42532    18
```

By reducing the dataset to observations exclusively from Santa Barbara, we have reduced the number of observations by over 10,000.

**column reduction**

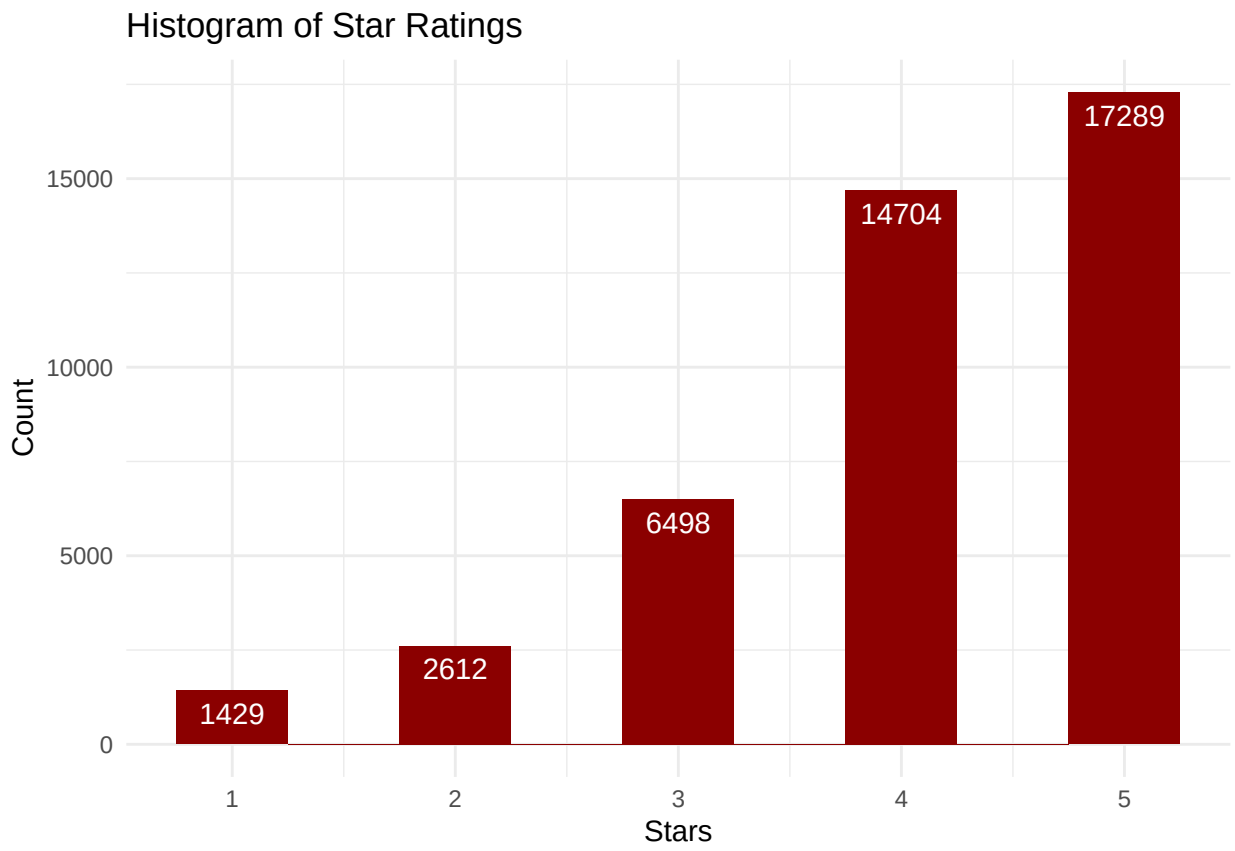
For now, I am only Interested in the text review, the star rating, and the business id. I will create a new data set called yelp which only contains the three columns of interest.

```
yelp <- data[c("Bus_id","Star","Review")]
head(yelp)
```

```
##                   Bus_id Star
## 2 SZU9c8V2GuREDN5KgyHFJw    5
## 3 eL4lyE7LNoXEMvpcJ8WNVw    3
## 4 SZU9c8V2GuREDN5KgyHFJw    5
## 6 0qu0fNTOsSmuREYVIMPuIQ    4
## 7 0qu0fNTOsSmuREYVIMPuIQ    4
## 8 -ujBP1Dw0j1-Ffaz97-LXQ    5
##
## 2
## 3
## 4 I love trying fresh seafood on piers, wharfs and seaside markets. Most of the time, it is a disapp
## 6
## 7
## 8
```

## Sentiment Analysis

```
ggplot(yelp, aes(x=Star))+
  geom_bar(stat="bin", bins= 9, fill="darkred") +
  geom_text(stat='count', aes(label=after_stat(count)), vjust=1.6, color="white") +
  ggtitle("Histogram of Star Ratings") +
  xlab("Stars") + ylab("Count") +
  theme_minimal()
```




Let's check to make sure that all the reviews are written in English

```
# language reduction
languages <- textcat(yelp$Review)
yelp <- yelp[languages == "english",]
dim(yelp)
```

```
## [1] 40457     3
```

Add a column to yelp indicating sentiment

```
yelp$Positive <- as.factor(yelp$Star >= 4)
table(yelp$Positive)
```

```
##
## FALSE  TRUE
##  9873 30583
```

Now lets remove stop words and punctuation from the reviews using metadata.

```
corpus <- VCorpus(VectorSource(yelp$Review))
corpus = tm_map(corpus, content_transformer(tolower))
corpus = tm_map(corpus, removePunctuation)
corpus = tm_map(corpus, removeWords,stopwords("english"))
```

```
corpus = tm_map(corpus, stemDocument)

corpus[[8]]$content
```

```
## [1] "stay bachelorett parti place got job done unfortun hotel santa barbara pretti freak pricey lot l
```

The following technique is called bag of words. It rearranges the data so that each word from the review is a column and each review is a row, with corresponding values for the number of times each respective word appears in each review.

```
frequencies <- DocumentTermMatrix(corpus)
sparse <- removeSparseTerms(frequencies, 0.99)
reviewsSparse <- as.data.frame(as.matrix(sparse))
colnames(reviewsSparse) <- make.names(colnames(reviewsSparse))


reviewsSparse$positive <- yelp$positive
```

## Sentiment analysis

To make this a classification problem, star ratings greater than or equal to 4 are positive and negative otherwise. Unfortunately, this will result in unbalanced data because of the high frequency of positive ratings. However, we won't force balance the data because this is the natural occurrence of the data.

We will create a new binary variable called "Sentiment" which indicates whether a rating is positive or negative.