Analysis of the Yelp Dataset

Jerry Tsien

Monday, November 16, 2015

Introduction

Based on the Yelp dataset [575 MB], this analysis tries to find the relationship between the rating of a business (how many stars it has) and the number of reviews it receives on the website of Yelp. For simplicity's sake, the dataset has already been downloaded and pre-processed, and the exact steps are listed in the Appendix.

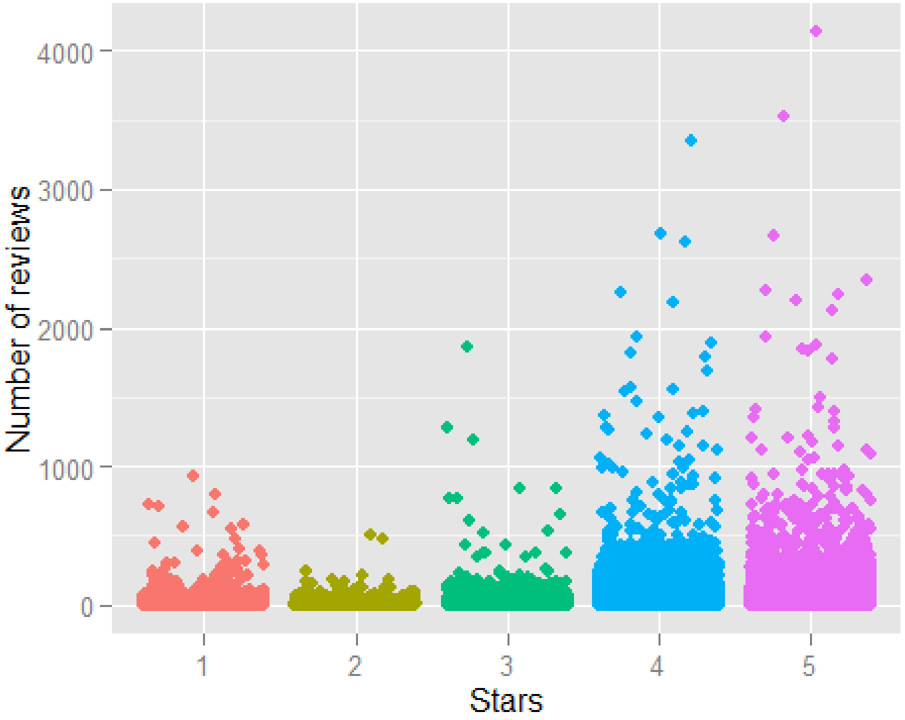
```r
# Loading the pre-processed data.
suppressWarnings(library(ggplot2))
biz <- readRDS("biz.rds")
rev <- readRDS("rev.rds")
```

Methods and Data

The overall rating of a business is determined by majority voting (selecting the mode of all the individual ratings), and the number of reviews are counted regardless of their content. For the purpose of this analysis, rating/stars and review count are calculated from the review file, rather than from the business file.

The relationship between the rating/stars and the number of reviews can be seen from the following graph.

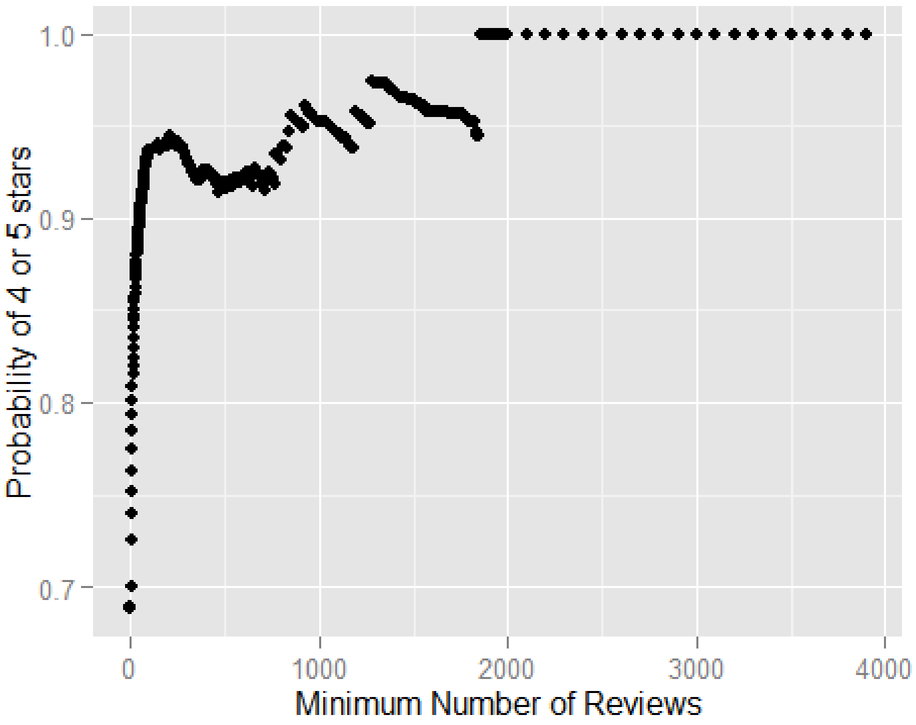
```r
# Number of reviews by number of stars:
p <- ggplot(rev, aes(rating, rev_cnt)) +
  geom_jitter(aes(color = rating)) +
  scale_x_discrete("Stars") +
  scale_y_continuous("Number of reviews") +
  theme(legend.position = "none")
p
```



```
# Calculate the percentage of 4 and 5 stars.
r4n5 <- rev[rev$rating %in% c("4", "5"), ]
p4n5 <- NROW(r4n5) / NROW(rev) * 100
p4n5 <- format(p4n5, digits = 2)
```

The graph shows that reviews are concentrated on the businesses with higher ratings. As a matter of fact, 69% of all the reviews are already focused on the 4-star and 5-star businesses. And the percentage continues to rise when businesses receive more reviews:

```
# Calculate the percentage of 4 and 5 stars at different levels:
df <- data.frame(lev = c(0:100,
                         seq(101, 2000, 10),
                         seq(2001, 4000, 100)),
                 n45 = 0, nt = 0, prob = 0)
for(i in 1:NROW(df)) {
  df$n45[i] <- NROW(r4n5[r4n5$rev_cnt > df$lev[i], ])
  df$nt[i] <- NROW(rev[rev$rev_cnt > df$lev[i], ])
}
df$prob <- df$n45 / df$nt
# Probability of higher rating as the number of reviews increases:
p2 <- ggplot(df, aes(lev, prob)) +
  geom_point() +
  scale_x_continuous("Minimum Number of Reviews") +
  scale_y_continuous("Probability of 4 or 5 stars")
p2
```



```
# The relationship is approximately linear when the number of reviews
is lower than 100.
df2 <- df[c(0:100), ]
m <- lm(prob ~ lev, data = df2)
summary(m)

##
## Call:
## lm(formula = prob ~ lev, data = df2)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.107227 -0.012163  0.006236  0.021345  0.027168
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 7.941e-01  5.634e-03  140.96   <2e-16 ***
## lev         1.777e-03  9.831e-05   18.07   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.02838 on 98 degrees of freedom
## Multiple R-squared:  0.7692, Adjusted R-squared:  0.7669
## F-statistic: 326.7 on 1 and 98 DF,  p-value: < 2.2e-16
```

```r
# Get the number of reviewes needed to increase the probability by 1%.
p100 <- 0.01 / m$coefficients[2]
p100 <- format(p100, digits = 1)
```

As the number of reviews reaches a certain threshold (50+), more than 90% of these businesses are rated 4-star or 5-star. And when the number of reviews is less than 100, there exists a linear relationship: 6 more reviews will increase the probability of 4 or 5 stars by 1%. However, when the number of reviews is greater than 100, no linear relationship can be found, although the probability still stays above 90%.

Finally, t-test can be used on the 4-star and 5-star businesses.

```r
t.test(rev_cnt ~ rating, paired = F, var.equal = F, data = r4n5)
```

```
## 
##  Welch Two Sample t-test
## 
## data:  rev_cnt by rating
## t = 7.8474, df = 33443.35, p-value = 4.375e-15
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  5.561368 9.264394
## sample estimates:
## mean in group 4 mean in group 5 
##        36.51901        29.10613
```

Although the difference between the *average* number of reviews of 4-star businesses and that of 5-star ones is statistically significant, it's worth noting that their absolute values are quite low, suggesting that the number of reviews approximately follow a Poisson distribution with a long tail. For this reason, 4-star businesses and 5-star one cannot be separated solely by the number of reviews.

Results

Based on the above analysis, the following conclusions can be drawn:

1. More than 2/3 of all the businesses with reviews in the Yelp dataset are rated 4-star or 5-star.

2. The more reviews a business receives, the more likely it'll be rated 4-star or 5-star.

3. On average, 6 additional reviews for a business will increase the probability of its receiving 4 or 5 stars by 1%, when the number of reviews is less than 100.

4. When the number of reviews is above 50, more than 90% of these businesses are rated 4-star or 5-star.

5. It's impossible to separate 4-star businesses and 5-star ones solely by the number of reviews they receive.

Discussion

The analysis shows that it's usually safe to *follow the crowd* for a customer to choose a business. In fact, satisfaction is almost guaranteed (at least 4 stars) for the business with the highest number of reviews.

Appendix

Below is the code for preprocessing the Yelp dataset, which should be downloaded and saved in a subfolder named "yelp" in the R working folder.

```r
# Use the following commands to start pre-processing:
# Preprocess()
Preprocess <- function() {
  suppressPackageStartupMessages(library('BBmisc'))
  pkgs <- c('jsonlite', 'readr', 'plyr', 'stringr', 'doParallel', 'ff',
'ffbase')
  suppressAll(lib(pkgs))
  rm(pkgs)
  registerDoParallel(cores = 16)
  biz <- LoadFile("business")
  biz <- biz[, c("business_id", "categories", "city",
                 "review_count", "state", "stars")]
  saveRDS(biz, "biz.rds")
  # Pagesize should be larger for the review file (taking about 15
minutes).
  rev <- LoadFile("review", 100000)
  rev <- ddply(rev, .(business_id), summarize,
               rating = GetMode(stars),
               rev_cnt = NROW(stars))
  saveRDS(rev, "rev.rds")
}
# Read the original JOSN file only once.
LoadFile <- function(fn, ps = 10000) {
  fname <- paste0(getwd(), '/yelp/yelp_academic_dataset_', fn, '.json')
  dat <- stream_in(file(fname), pagesize = ps)
  dat <- flatten(dat)
  dat
}
# Get the mode (most frequent value) of the variable.
GetMode <- function(var) {
  names(sort(-table(var)))[1]
}
```