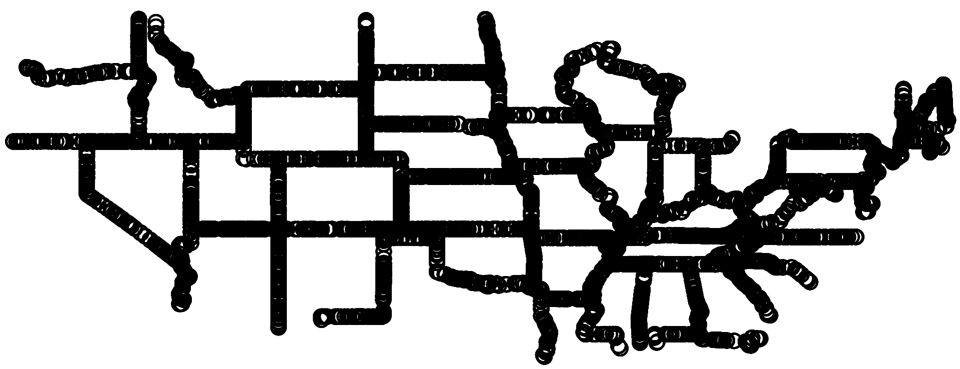
Analysis

Ragip Gurlek

4/17/2020

```r
library(tidyr)
library(dplyr)
library(plyr)
library(sf)
library(knitr)
library(summarytools)
seed <- 8234
```

```r
# Plot the sampled border points ####
point_sample <- list()
for(i in seq(1234, seed, 1000)){
  point_sample <- c(point_sample, readRDS(paste0("point_sample_", seed, ".rds")))
}
plot(st_geometrycollection(point_sample))
```



```r
length(point_sample)
```

```
## [1] 32000
```

```r
remove(point_sample)
```

```r
# Obsorve number of clusters with each nrow value
rest_data <- list()
for(i in seq(1234, seed, 1000)){
  rest_data <- c(rest_data, readRDS(paste0("rest_data_", seed, ".rds")))
}
table(sapply(rest_data, nrow))
```

```
##
##    1    2    3    4    5    6    7    8    9   10   11   12   13   14   15   16
## 2104 1264 1136 1168 1496 1432  944  648  568  616  552  624  488  496  480  456
##   17   18   19   20   21   22   23   24   25   26   27   28   29   30   31   32
##  360  424  312  328  272  208  360  248  328  264  160  184  200  168  152  128
##   33   34   35   36   37   38   39   40   41   42   43   44   45   46   47   48
##  112  192  144   96   88  144   32  120   72  160   80   80  104   56   56   48
##   49   50
##   24 5696
```

```r
sum(sapply(rest_data, nrow))
```

```
## [1] 536432
```

```r
remove(rest_data)
```

```r
# An example from the original data ####
library(yelpr)

key <- readLines("api_key.txt")
radius = 16000 # about 10 miles
longitude <- -85.01723
latitude <- 31.00204
bus_data <- suppressMessages(business_search(
  api_key = key,
  latitude = latitude,
  longitude = longitude,
  radius = radius,
  limit = 50
))
bus_data <- bus_data$businesses
bus_data$categories
```

```
## [[1]]
##      alias    title
## 1 southern Southern
## 2  buffets  Buffets
##
## [[2]]
##     alias     title
## 1 burgers   Burgers
## 2 hotdogs Fast Food
##
## [[3]]
```

```
##           alias         title
## 1       italian       Italian
## 2         pizza         Pizza
## 3 chicken_wings Chicken Wings
##
## [[4]]
##   alias    title
## 1   bbq Barbeque
##
## [[5]]
##      alias    title
## 1  buffets  Buffets
## 2 southern Southern
## 3  burgers  Burgers
##
## [[6]]
##        alias       title
## 1 foodtrucks Food Trucks
## 2        bbq    Barbeque
##
## [[7]]
##     alias   title
## 1 mexican Mexican
##
## [[8]]
##     alias             title
## 1   parks             Parks
## 2  diving            Diving
## 3 rafting Rafting/Kayaking
##
## [[9]]
##        alias       title
## 1 sportsbars Sports Bars
##
## [[10]]
##        alias      title
## 1 sandwiches Sandwiches
##
## [[11]]
##    alias  title
## 1 diners Diners
```

Descriptive Stats From Raw Data

```
raw_data <- read.csv("raw_data.csv")
state_counts <- table(raw_data$state)
state_counts <- sort(state_counts, decreasing = T)
length(state_counts)
```

```
## [1] 53
```

```r
descr(raw_data[,c(6:9, 14:19)], stats = "fivenum", style = "rmarkdown")
```

Descriptive Statistics

raw_data **N:** 269880

Table 1: Table continues below

	CityRate	CombinedRate	CountyRate	distance	price	rating
Min	0.00	0.00	0.00	32.93	1.00	1.00
Q1	0.00	0.06	0.00	5423.08	1.00	3.50
Median	0.00	0.07	0.00	8535.86	2.00	4.00
Q3	0.00	0.08	0.01	12278.37	2.00	4.50
Max	0.06	0.11	0.05	159221.29	4.00	5.00

	review_count	SpecialRate	StateRate
Min	1.00	0.00	0.00
Q1	10.00	0.00	0.05
Median	30.00	0.00	0.06
Q3	88.00	0.00	0.06
Max	12206.00	0.04	0.07

```r
# Variable means
summary_table <- aggregate(raw_data[, c("rating", "price", "CombinedRate", "review_count")],
          list(raw_data$is_focal), mean, na.rm = T)[,2:5]
rownames(summary_table) <- c("Non-focal", "Focal")
kable(summary_table, digits = c(3,3,4,2),
      caption = "Non-focal vs Focal Means")
```

Table 3: Non-focal vs Focal Means

	rating	price	CombinedRate	review_count
Non-focal	3.823	1.578	0.0677	92.50
Focal	3.822	1.572	0.0669	92.35

```r
# Variable sd
summary_table <- aggregate(raw_data[, c("rating", "price", "CombinedRate", "review_count")],
          list(raw_data$is_focal), sd, na.rm = T)[,2:5]
rownames(summary_table) <- c("Non-focal", "Focal")
kable(summary_table, digits = c(3,3,4,2),
      caption = "Non-focal vs Focal Standard Deviations")
```

Table 4: Non-focal vs Focal Standard Deviations

	rating	price	CombinedRate	review_count
Non-focal	0.790	0.579	0.0194	261.90
Focal	0.793	0.578	0.0201	273.63

```r
# Rating t-test for focal vs non-focal
focal <- raw_data[raw_data$is_focal, ]
non_focal <- raw_data[!raw_data$is_focal, ]
t.test(focal$rating, non_focal$rating)
```

```
##
##  Welch Two Sample t-test
##
## data:  focal$rating and non_focal$rating
## t = -0.34559, df = 269175, p-value = 0.7297
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.007027672  0.004920877
## sample estimates:
## mean of x mean of y
##  3.821752  3.822805
```

```r
# Price t-test for focal vs non-focal
t.test(focal$price, non_focal$price)
```

```
##
##  Welch Two Sample t-test
##
## data:  focal$price and non_focal$price
## t = -2.5123, df = 269248, p-value = 0.012
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.009962973 -0.001230387
## sample estimates:
## mean of x mean of y
##  1.572434  1.578031
```

```r
# CombinedRate t-test for focal vs non-focal
t.test(focal$CombinedRate, non_focal$CombinedRate)
```

```
##
##  Welch Two Sample t-test
##
## data:  focal$CombinedRate and non_focal$CombinedRate
## t = -10.847, df = 267227, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.0009778442 -0.0006785454
## sample estimates:
##  mean of x  mean of y
## 0.06687031 0.06769850
```

```r
# StateRate t-test for focal vs non-focal
t.test(focal$StateRate, non_focal$StateRate)
```

```
##
##  Welch Two Sample t-test
##
## data:  focal$StateRate and non_focal$StateRate
## t = -5.7067, df = 266820, p-value = 1.153e-08
## alternative hypothesis: true difference in means is not equal to 0
```

```
## 95 percent confidence interval:
##  -0.0004709795 -0.0002301686
## sample estimates:
##  mean of x  mean of y
## 0.05343817 0.05378874
```

```r
# CityRate t-test for focal vs non-focal
t.test(focal$CityRate, non_focal$CityRate)
```

```
##
##  Welch Two Sample t-test
##
## data:  focal$CityRate and non_focal$CityRate
## t = -8.6417, df = 269007, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.0003757692 -0.0002368292
## sample estimates:
##   mean of x   mean of y
## 0.003974872 0.004281171
```

```r
# Review Count t-test for focal vs non-focal
t.test(focal$review_count, non_focal$review_count)
```

```
##
##  Welch Two Sample t-test
##
## data:  focal$review_count and non_focal$review_count
## t = -0.15101, df = 267629, p-value = 0.88
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -2.178388  1.866718
## sample estimates:
## mean of x mean of y
##  92.34609  92.50193
```

Regression

```r
reg_data <- read.csv("reg_data.csv")
min(reg_data$review_count)
```

```
## [1] -1950.587
```

```r
library(AER)

model <- ivreg(rating ~ log(review_count+1951) + price | log(review_count+1951) + StateRate +
                 CountyRate + CityRate + SpecialRate, data = reg_data)
summary(model)
```

```
##
## Call:
## ivreg(formula = rating ~ log(review_count + 1951) + price | log(review_count +
##     1951) + StateRate + CountyRate + CityRate + SpecialRate,
##     data = reg_data)
##
## Residuals:
```

```
##      Min       1Q   Median       3Q      Max
## -3.87497 -0.50684  0.05816  0.60856  4.55245
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)              3.02813    0.23986   12.62   <2e-16 ***
## log(review_count + 1951) -0.41725    0.03163  -13.19   <2e-16 ***
## price                    0.41688    0.03710   11.24   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9417 on 128361 degrees of freedom
## Multiple R-Squared: -0.09717,    Adjusted R-squared: -0.09718
## Wald test: 94.79 on 2 and 128361 DF,  p-value: < 2.2e-16
```

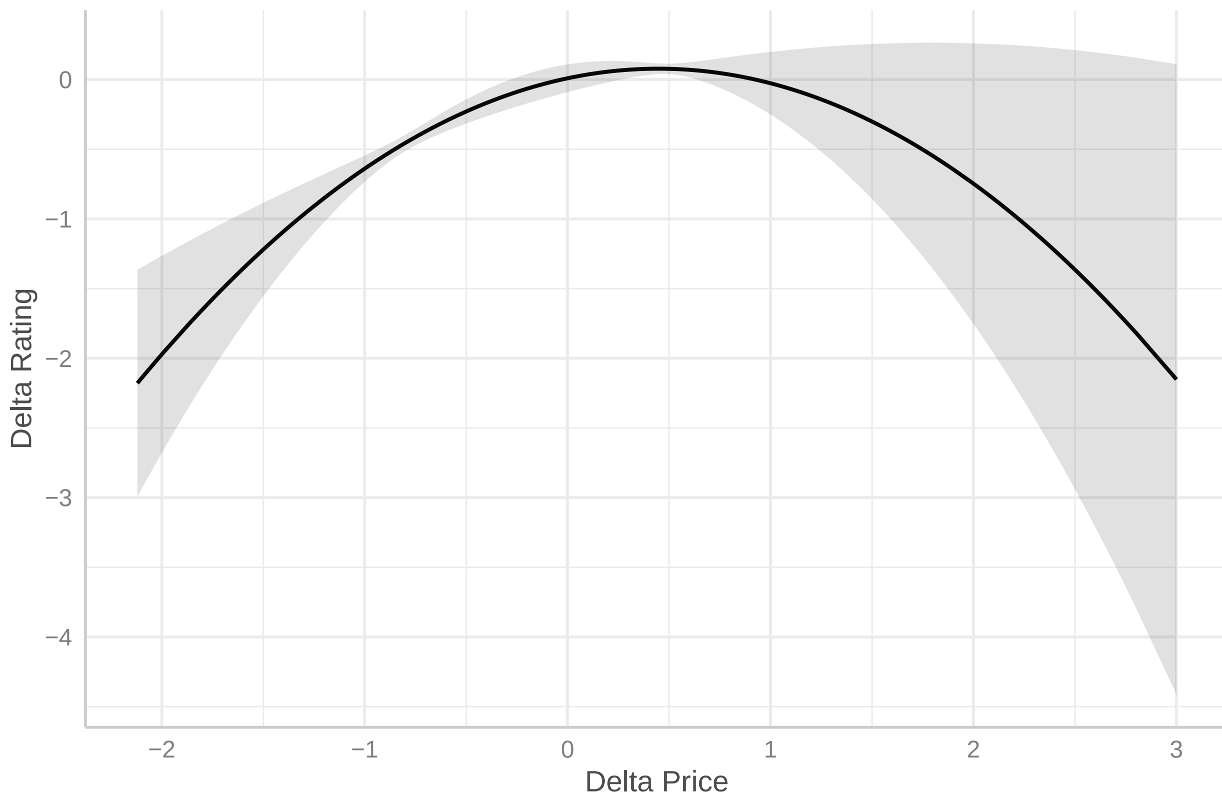
```r
model <- ivreg(rating ~ log(review_count+1951) + price + I(price^2)|
                 log(review_count+1951) + StateRate +
                 CountyRate + CityRate + SpecialRate, data = reg_data)
summary(model)
```

```
##
## Call:
## ivreg(formula = rating ~ log(review_count + 1951) + price + I(price^2) |
##     log(review_count + 1951) + StateRate + CountyRate + CityRate +
##         SpecialRate, data = reg_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.0176 -0.5099  0.0276  0.5801  5.4754
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)              2.64612    0.27555   9.603  < 2e-16 ***
## log(review_count + 1951) -0.34783    0.04003  -8.690  < 2e-16 ***
## price                    0.30604    0.05388   5.680 1.35e-08 ***
## I(price^2)              -0.34221    0.12030  -2.845  0.00445 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9447 on 128360 degrees of freedom
## Multiple R-Squared: -0.104,  Adjusted R-squared: -0.1041
## Wald test:  65.5 on 3 and 128360 DF,  p-value: < 2.2e-16
```

This plot shows the prediction for Delta rating, not the effect (coefficient). It tells us that if I am cheaper than average of my neighbor restaurants, I am rated lower than them. Being a little bit expensive than the average slightly helps. However, after a point, it hurts the ratings. Specifically, being more expensive by a value between 0.26 and 0.64 significantly improves the ratings. The maximum effect is achived when the difference is 0.45, which improves the rating by 0.08. Being more expensive than the aveeage by a value more than 0.92 hurts the rating although this effect is not significant due to high standard errors in that region. Note that these results are obtained after controlling for confounders like quality (thanks to IV and matching).

```r
library(ggeffects)
mydf <- ggpredict(model, terms = "price [all]")
attributes(mydf)$title <- ""
attributes(mydf)$x.title <- "Delta Price"
```

```r
attributes(mydf)$y.title <- "Delta Rating"
plot(mydf)
```




```r
mydf[abs(mydf$predicted) < 0.001, ]
```

```
##
## #
## # x = Delta Price
##
##      x | Predicted |   SE |        95% CI
## ---------------------------------------------
## -0.03 |         0 | 0.05 | [-0.10, 0.10]
## -0.03 |         0 | 0.05 | [-0.10, 0.10]
## -0.03 |         0 | 0.05 | [-0.10, 0.10]
## -0.03 |         0 | 0.05 | [-0.10, 0.10]
## -0.03 |         0 | 0.05 | [-0.10, 0.10]
##  0.92 |         0 | 0.09 | [-0.18, 0.19]
##  0.92 |         0 | 0.09 | [-0.18, 0.19]
##  0.93 |         0 | 0.10 | [-0.19, 0.19]
##
## Adjusted for:
## * review_count =  7.58
## *    StateRate = -0.00
## *   CountyRate =  0.00
## *     CityRate =  0.00
## *  SpecialRate = -0.00
```

```r
signi_improve <- mydf[mydf$conf.low >0, ]
signi_improve[c(1,nrow(signi_improve)), ]
```

```
##
## #
## # x = Delta Price
##
##    x | Predicted |   SE |       95% CI
## --------------------------------------
## 0.26 |      0.07 | 0.03 | [0.00, 0.13]
## 0.64 |      0.07 | 0.03 | [0.00, 0.13]
##
## Adjusted for:
## * review_count =  7.58
## *    StateRate = -0.00
## *   CountyRate =  0.00
## *     CityRate =  0.00
## *  SpecialRate = -0.00
```

```r
signi_improve[which.max(signi_improve$predicted), ]
```

```
##
## #
## # x = Delta Price
##
##    x | Predicted |   SE |       95% CI
## --------------------------------------
## 0.45 |      0.08 | 0.02 | [0.04, 0.12]
##
## Adjusted for:
## * review_count =  7.58
## *    StateRate = -0.00
## *   CountyRate =  0.00
## *     CityRate =  0.00
## *  SpecialRate = -0.00
```