Problem Set 1

Sachin Badole

9/17/2020

Economics 725: Machine Learning for Economists, University of Wisconsin-Madison.

Question (1)
0) I have download Rstudio.
1) I have download the DB1BMarket data table as per given instrucation in problem set 1.
a. Take only data from the first quarter of 2015.- Done
b. Take the following variables: ItinID, MktID, OriginAirportID, DestAirportID, TkCarrierChange, TicketCarrier, Passengers, MarketFare, and MarketDistance. - Done
c. Download the data and bring it into R. - Done

```
# Set the working directory.
setwd("G:/My Documents/Sem III/Econ 725 Machine Learning for Econmist/Problem Sets/Problem set 1")

# Load dataset using the following command.
Airline_ticktes_data <- read.csv("65799243_T_DB1B_MARKET.csv")

# 1) b) Here, I am checking the name of the columns.
names(Airline_ticktes_data)  # Name of columns
```

```
##  [1] "ITIN_ID"           "MKT_ID"            "ORIGIN_AIRPORT_ID"
##  [4] "DEST_AIRPORT_ID"   "TK_CARRIER_CHANGE" "TICKET_CARRIER"
##  [7] "PASSENGERS"        "MARKET_FARE"       "MARKET_DISTANCE"
## [10] "X"
```

```
# I found there is one extra column in the above dataset, so
# I have dropped it.
Airline_ticktes_data <- subset(Airline_ticktes_data, select = -c(X))
```

Initial number of observations.

```
length(Airline_ticktes_data$ITIN_ID)  # Initial number of observations.
```

```
## [1] 5582629
```

Question (2)
Remove tickets that can't be assigned to a unique carrier, remove markets (a unidirectional origin-destination pair) with less than 20 passengers per day, and remove tickets with extreme prices.

```
# 2.a) Removing any tickets taht have a ticket carrier
# change. Ticketing Carrier Change Indicator (1=Yes)
Airline_ticktes_data_v1 <- Airline_ticktes_data[Airline_ticktes_data$TK_CARRIER_CHANGE ==
    0, ]
```

The number of observation after Removing any tickets taht have a ticket carrier change.

```r
# shows number of observatins after the removing the
# ticketing carrier.
length(Airline_ticktes_data_v1$TK_CARRIER_CHANGE)
```

```
## [1] 5345056
```

```r
# 2.c) Remove tickets with prices less than $25 or more than
# $2,500.
Airline_ticktes_data_v1 <- Airline_ticktes_data_v1[Airline_ticktes_data_v1$MARKET_FARE >
    25 & Airline_ticktes_data_v1$MARKET_FARE < 2500, ]
```

The number of observation after remove tickets with prices less than $25 or more then $2500.

```r
# The number of observation after remove tickets with prices
# less than $25 or more then $2500.
length(Airline_ticktes_data_v1$MARKET_FARE)
```

```
## [1] 5078406
```

```r
# 2.b ) Create new Variable called Total_No_Passengers which
# defind that passengers number multiply by 10 for each
# ticket.

# Find the total numebr of passengers in each market.
Airline_ticktes_data_v1 <- Airline_ticktes_data_v1 %>% group_by(ORIGIN_AIRPORT_ID,
    DEST_AIRPORT_ID) %>% mutate(TOTAL_PASSENGERR = sum(PASSENGERS) <
    (365/4) * 20/10)

# Drop some observations those are duplicates.
Airline_ticktes_data_v1 <- Airline_ticktes_data_v1[!(Airline_ticktes_data_v1$TOTAL_PASSENGERR ==
    TRUE), ]
```

The number of observation after find the total number of passengers in each market.

```r
# The number of observation after find the total number of
# passengers in each market.
length(Airline_ticktes_data_v1$TOTAL_PASSENGERR)
```

```
## [1] 4039709
```

Question (3)- You will create two datasets: one at the market-carrier level and another at the market level.

a) For each market-airline. (Calculate the average price, Calculate the total number of passengers, and Calculate the average distance.)

```r
# a) For each market-airline.

data_market_airline <- Airline_ticktes_data_v1 %>% group_by(ORIGIN_AIRPORT_ID,
    DEST_AIRPORT_ID, TICKET_CARRIER) %>% mutate(TOTAL_NO_PASSENGERS = sum(PASSENGERS) *
    10, AVERAGE_PRICE_AIRLINE = weighted.mean(MARKET_FARE, PASSENGERS),
    AVERAGE_DISTANCE_AIRLINE = weighted.mean(MARKET_DISTANCE,
        PASSENGERS))

data_market_airline <- data_market_airline %>% group_by(ORIGIN_AIRPORT_ID,
    DEST_AIRPORT_ID, TICKET_CARRIER) %>% distinct(TOTAL_NO_PASSENGERS,
    AVERAGE_PRICE_AIRLINE, AVERAGE_DISTANCE_AIRLINE)
```

The number of observation in the Market-airline dataset.

```r
# The number of observation in the Market-airline dataset.
length(data_market_airline$TOTAL_NO_PASSENGERS)
```

```
## [1] 28286
```

```r
# b)For each market

data_market <- data_market_airline %>% group_by(ORIGIN_AIRPORT_ID,
    DEST_AIRPORT_ID) %>% mutate(AVERAGE_PRICE_MARKET = weighted.mean(AVERAGE_PRICE_AIRLINE,
    TOTAL_NO_PASSENGERS), AVERAGE_DISTANCE_MARKET = weighted.mean(AVERAGE_DISTANCE_AIRLINE,
    TOTAL_NO_PASSENGERS), HHI = sum(((TOTAL_NO_PASSENGERS * 100)/sum(TOTAL_NO_PASSENGERS))^2))

data_market <- data_market %>% count(DEST_AIRPORT_ID, AVERAGE_PRICE_MARKET,
    AVERAGE_DISTANCE_MARKET, HHI, sort = TRUE, name = "TOTAL_NO_FIRMS")

colnames(data_market) = tolower(colnames(data_market))
```

The number of observation in the Market-level dataset.

```r
# The number of observation in the Market-level dataset.
length(data_market$hhi)
```

```
## [1] 6299
```

Load the given populations data and merge with the market-level dataset.

```r
# load given populations data for merge.
load(file = "populations.R")

# data_market <-
# merge(data_market,populations,by.x='origin_airport_id',
# by.y='dest_airport_id')
data_market <- merge(data_market, populations, by = c("origin_airport_id",
    "dest_airport_id"))
```

The number of observation in the Market-level dataset after merging with population data.

```r
# The number of observation in the Market-level dataset after
# merging with population data.
length(data_market$hhi)
```

```
## [1] 6137
```

4) Generate tables with summary statistics for each of your datasets and generate plots characterizing the distributions of market level prices and HHI as well as the relationship between them.

a) Report summary statistics for your tables (hint: use the kable function in the knitr package).

Summary Statistics for each market-airline.

```r
kable(summary(data_market_airline[, c("TOTAL_NO_PASSENGERS",
    "AVERAGE_PRICE_AIRLINE", "AVERAGE_DISTANCE_AIRLINE")]))
```

TOTAL_NO_PASSENGERS	AVERAGE_PRICE_AIRLINE	AVERAGE_DISTANCE_AIRLINE
Min. : 10	Min. : 40.39	Min. : 68.0
1st Qu.: 210	1st Qu.: 201.16	1st Qu.: 871.6
Median : 690	Median : 250.96	Median :1294.5
Mean : 3039	Mean : 260.28	Mean :1490.0
3rd Qu.: 2010	3rd Qu.: 303.42	3rd Qu.:2013.1
Max. :136120	Max. :2403.00	Max. :7437.8

Summary Statistics for each market.

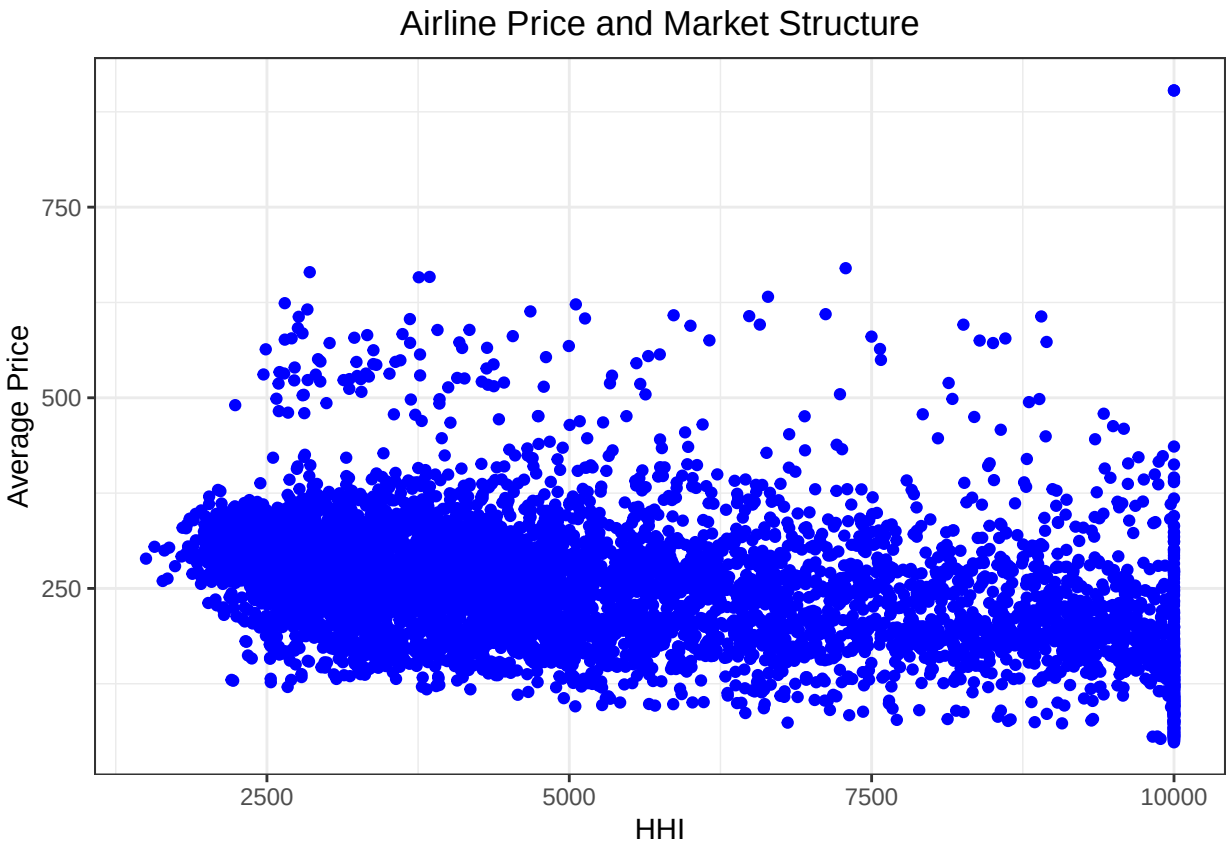
```
View(data_market)
kable(summary(data_market[, c("average_price_market", "average_distance_market",
    "hhi", "market_size", "total_no_firms")]))
```

average_price_market	average_distance_market	hhi	market_size	total_no_firms
Min. : 48.22	Min. : 84	Min. : 1500	Min. : 99639	Min. : 1.000
1st Qu.:188.98	1st Qu.: 665	1st Qu.: 3383	1st Qu.: 1414240	1st Qu.: 4.000
Median :247.41	Median :1034	Median : 4788	Median : 2160635	Median : 5.000
Mean :247.03	Mean :1233	Mean : 5474	Mean : 2780314	Mean : 4.483
3rd Qu.:297.71	3rd Qu.:1639	3rd Qu.: 7416	3rd Qu.: 3516122	3rd Qu.: 5.000
Max. :903.18	Max. :5210	Max. :10000	Max. :16338394	Max. :11.000

b) Plots

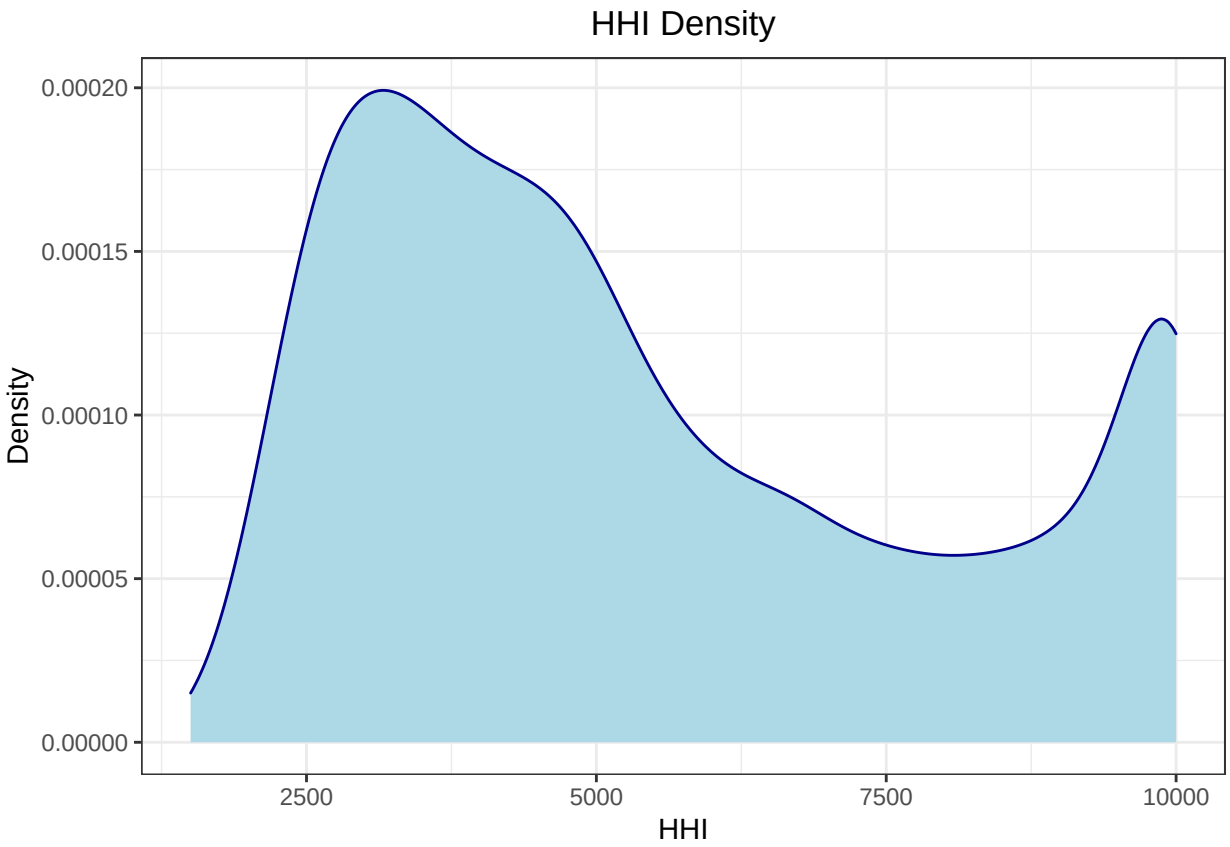
1. The Scatter plot of HHI versus prices at the market level.

```
ggplot(data_market, aes(x = hhi, y = average_price_market)) +
    geom_point(color = "blue") + theme_bw() + labs(title = "Airline Price and Market Structure",
    x = "HHI", y = "Average Price") + theme(plot.title = element_text(hjust = 0.5))
```



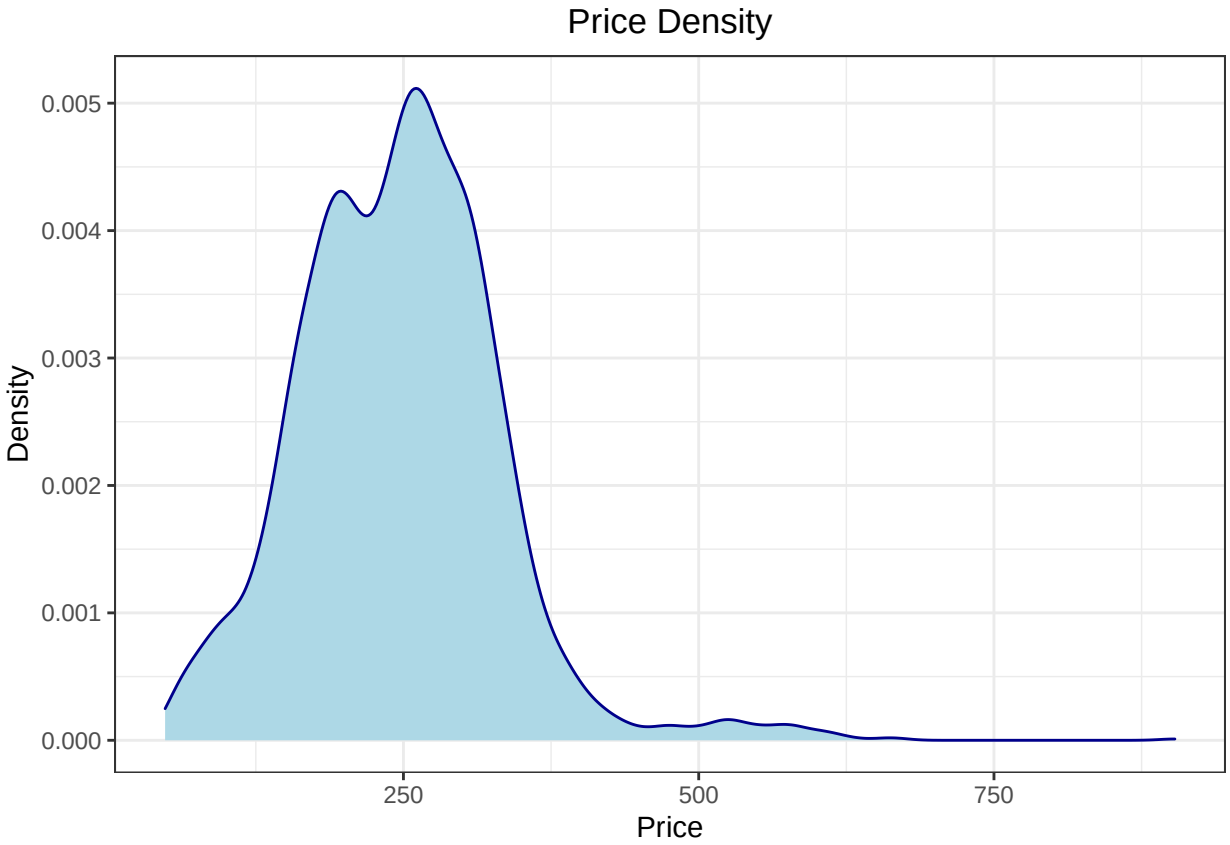
2. The market level HHI density plot.

```
ggplot(data_market, aes(x = hhi)) + labs(title = "HHI Density",
    x = "HHI", y = "Density") + geom_density(color = "darkblue",
    fill = "lightblue") + theme_bw() + theme(plot.title = element_text(hjust = 0.5))
```



3. The market level Average price density plot.

```
ggplot(data_market, aes(x = average_price_market)) + labs(title = "Price Density",
    x = "Price", y = "Density") + geom_density(color = "darkblue",
    fill = "lightblue") + theme_bw() + theme(plot.title = element_text(hjust = 0.5))
```


Price Density
Density
0.005
0.004
0.003
0.002
0.001
0.000
250
500
750
Price