W203 Lab01 Broadband Exploration

Chandra Sekar, Bhuvnesh Sharma, Eugene Tang

Introduction

Research Question

With the recent rise of the Internet and its increasing importance in our lives, the availablility of access to the Internet has also become an increasingly large question and necessity in some societies. In this analysis, we look at broadband markets and in particular, three aspects of these markets and their relationships with each other:

- Price: how much does it cost to access the Internet
- Penetration: what fraction of customers have access to network service
- Speed: what rate can customers upload or download bits of data

We in particular consider this data in the context of open access policies. Much of the developed world has developed aggressive regulatory structures to compel network owners to increase penetration while there are some nations that do not. It is still an open debate on whether such policies are beneficial or harmful in price, penetration, and speed.

In this analysis we seek to tackle two main questions:

- Does a trade-off exist between network price, penetration, and speed?
- Is there evidence for beneifical effects of open access policies?

Dataset Setup (code)

Here is the code we used to prepare our dataset. Please see the below sections to see why certain decisions were made in the preparation of the dataset.

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(car)
```

```
##
## Attaching package: 'car'
```

```
## The following object is masked from 'package:dplyr':
##
##     recode
```

```
df_price = read.table("Price.csv", header = TRUE, sep = ",")
df_penetration = read.table("Penetration_Modified.csv", header = TRUE, sep = ",")
df_speed = read.table("Speed_Modified.csv", header = TRUE, sep = ",")
```

```r
# some dataset cleaning (see "Data Processing" for more details on why we did this)
drops_penetration <- c("X") # extra column to remove
df_penetration = df_penetration[ , !(names(df_penetration) %in% drops_penetration)]
colnames(df_speed)[2] <- "Country.Code" # rename country code column

# convert numeric fields to the numeric type
convert_to_numeric = function(col) {
  return(as.numeric(sub("[\\$%,]","", col)))
}
NON_ID_DATA_START <- 3 # columns 1 and 2 are the country and country code data
df_price[NON_ID_DATA_START:length(df_price)] = lapply(df_price[NON_ID_DATA_START:length(df_price)], conv
df_penetration[NON_ID_DATA_START:length(df_penetration)] =
  lapply(df_penetration[NON_ID_DATA_START:length(df_penetration)], convert_to_numeric)
df_speed[NON_ID_DATA_START:length(df_speed)] =
  lapply(df_speed[NON_ID_DATA_START:length(df_speed)], convert_to_numeric)
```

```
## Warning in FUN(X[[i]], ...): NAs introduced by coercion
```

```r
# Join the data
df_partial = full_join(df_penetration, df_price, by = c("Country", "Country.Code"))
df_full = full_join(df_partial, df_speed, by = c("Country", "Country.Code"))
```

Dataset Description

Our dataset comes in three csv files. One for price, penetration, and speed respectively. Each dataset contains observations on 30 countries, with one row for each country. Each dataset contains a variety of variables. Below we include tables of each variable, its type, and its description.

Each of the three datasets contains string columns to represent country and country code field. We use the two fields to join the datasets together since they are unique across each row (though either field individually would also have sufficed. For conciseness, we exclude these two columns in the tables below.

```r
writeLines(paste(length(df_full$Country), "countries"))
```

```
## 30 countries
```

```r
writeLines(paste(df_full$Country, collapse = ", ", sep = ""))
```

```
## Australia, Austria, Belgium, Canada, Czech Republic, Denmark, Finland, France, Germany, Greece, Hunga
```

Price Dataset

Our price dataset contained information on the cost to access different levels of Internet

Column	Interpretation
Price for low speeds, combined	Price (in USD$) for low-speed Internet
Price for med speeds, combined	Price (in USD$) for medium-speed Internet
Price for high speeds, combined	Price (in USD$) for high-speed Internet
Price for very high speeds, combined	Price (in USD$) for very high-speed Internet

Penetration Dataset

Our penetration dataset contained information on how widespread network availability is in a country.

Column	Interpretation
Penetration per 100 OECD, 2008	"Penetration" per 100 people in 2008 as measured by the OECD
Penetration per 100 OECD, 2007	"Penetration" per 100 people in 2007 as measured by the OECD
Household penetration, OECD	Household penetration as measured by the OECD
2G and 3G penetration per 100, OECD	2G and 3G network penetration per 100 people for OECD
Penetration per 100 GC	Penetration per 100 people as measured by GC
3G penetration per 100	3G network penetration per 100 people
Growth in 3G penetration	Growth in 3G network penetration
Wi-Fi hotspots, JiWire	Number of Wi-Fi hotspots
Wi-Fi hotspots per 100,000, JiWire	Number of Wi-Fi hotspots per 100,000 people
Percent of population in urban areas	Percent of population in urban areas with access to the Internet

Speed Dataset

The speed dataset has various metrics on the speed of the network available in a country.

Column	Interpretation
Maximum advertised speed OECD (kbps)	(same as column name)
Average advertised speed OECD (kbps)	(same as column name)
Average actual speed, Akamai (kbps)	(same as column name)
Average download speedtest.net (kbps)	(same as column name)
Standard deviation download, speedtest.net	(same as column name)
Average upload speedtest.net (kbps)	(same as column name)
Standard deviation upload, speedtest.net	(same as column name)
Average latency speedtest.net	Average latency in ms (measured by speedtest.net)
Standard deviation latency, speedtest.net	Standard deviation of latency in ms (measured by speedtest.net)
Median download, speedtest.net (kbps)	(same as column name)
Median upload, speedtest.net (kbps)	(same as column name)
Median latency, speedtest.net	Median latency in ms (measured by speedtest.net)
90p. Download, speedtest.net (kbps)	Top 90% of download speed (measured by speedtest.net)
90p. Upload, speedtest.net (kbps)	Top 90% of upload speed (measured by speedtest.net)
10p. Latency, speedtest.net	Lowest 10% of latency in ms (measured by speedtest.net)

Data Processing / Preparation

When first starting with the dataset, we noticed that two of the files (Speed.csv and Penetration.csv) had extra lines at the bottom. To help work with this, we manually removed the two lines and saved the results in Speed_Modified.csv and Penetration_Modified.csv (also attached).

The next thing we noticed was that the penetration dataset had an extra column called X with no values. We chose to removed this column since it has no values. In the speed dataset, we noticed that the country code column was named differently ("Country code") from that in the penetration and price datasets ("Country.Code"). To adjust for this, we renamed the country code column to "Country.Code" to help

facilitate joins later on.

```
# some dataset cleaning
drops_penetration <- c("X") # extra column
df_penetration = df_penetration[ , !(names(df_penetration) %in% drops_penetration)]
colnames(df_speed)[2] <- "Country.Code"
```

After performing these adjustments, we found that a lot of data were read as factors because some columns had percentage signs or commas in the numbers which had the read.tables function interpret the input as Strings instead of numbers. Because we would want to interact with all of the values as numbers (short of the country and country code columns), we next converted all of the columns to numeric type.

```
# convert numeric fields to the numeric type
convert_to_numeric = function(col) {
  return(as.numeric(sub("[\\$%,]","", col)))
}
NON_ID_DATA_START <- 3 # columns 1 and 2 are the country and country code data
df_price[NON_ID_DATA_START:length(df_price)] =
  lapply(df_price[NON_ID_DATA_START:length(df_price)], convert_to_numeric)
df_penetration[NON_ID_DATA_START:length(df_penetration)] =
  lapply(df_penetration[NON_ID_DATA_START:length(df_penetration)], convert_to_numeric)
df_speed[NON_ID_DATA_START:length(df_speed)] = lapply(df_speed[NON_ID_DATA_START:length(df_speed)], con
```

Finally, we joined the three tables to make it easier to work with.

```
# Join the data
df_partial = full_join(df_penetration, df_price, by = c("Country", "Country.Code"))
df_full = full_join(df_partial, df_speed, by = c("Country", "Country.Code"))
```

Data Quality Evaluation

There were a few columns that had fewer than 30 observations.

```
cols_with_nas <- colnames(df_full)[colSums(is.na(df_full)) > 0]
```

Of these columns, the measurement of the price of very high Internet speeds by far was the most incomplete (had the most NAs). One possible explanation for this is that not all countries have very high speed Internet and thus would not have a value for this column. For columns with NAs, we decided to omit the countries with the NAs from our analyses since we do not have a good default to put in.

```
print_nas = function(colname) {
  writeLines(paste("Number of NAs in ", colname, ": ", sum(is.na(df_full[colname])), sep = ""))
}
invisible(lapply(cols_with_nas, print_nas))
```

```
## Number of NAs in Price.for.low.speeds..combined: 1
## Number of NAs in Price.for.high.speeds..combined: 2
## Number of NAs in Price.for.very.high.speeds..combined: 11
## Number of NAs in Maximum.advertised.speed.OECD..kbps.: 1
## Number of NAs in Average.actual.speed..Akamai..kbps.: 4
```

Outside of this, our data was fairly complete. However, looking through our data, we did notice some irregularities. The first we found is that Poland's value for the percent of population in urban areas field is 162%. We do not have a full understanding of what that means, but it seems that this field measures a percentage of the population of people in urban areas, so since $162\% > 100\%$, we are a little dubious of that data point and feel that it could be likely due to reporting error. For this reason, we decided to remove Poland from analyses containing this column.

Another irregularity was that Turkey's 3G penetration per 100 is 0 despite having 20.20% growth in 3G penetration. It is possible that Turkey had a really really low 3G penetration such that a 20.20% growth would still result in an average of 0 per 100. However, given some of the other values in the other fields, (e.g. 2G and 3G penetration per 100, OECD having a value of 84.93), we are a little dubious of this value and decided to remove Turkey from analyses containing this column.

Finally, we noticed that in the measurement of the standard deviation of latency for Iceland, Iceland has an unusually high value of 1,199 (where all the other standard deviations are around 300). TODO: what did we do about this?

Univariate Analysis of Key Variables

Now we have the combined data in the data frame (df_full) , we can use the invividual dataframe to perform the Univariate analysis of Key variables. First we renamed the column names for clarity while displaying summary and the plots.

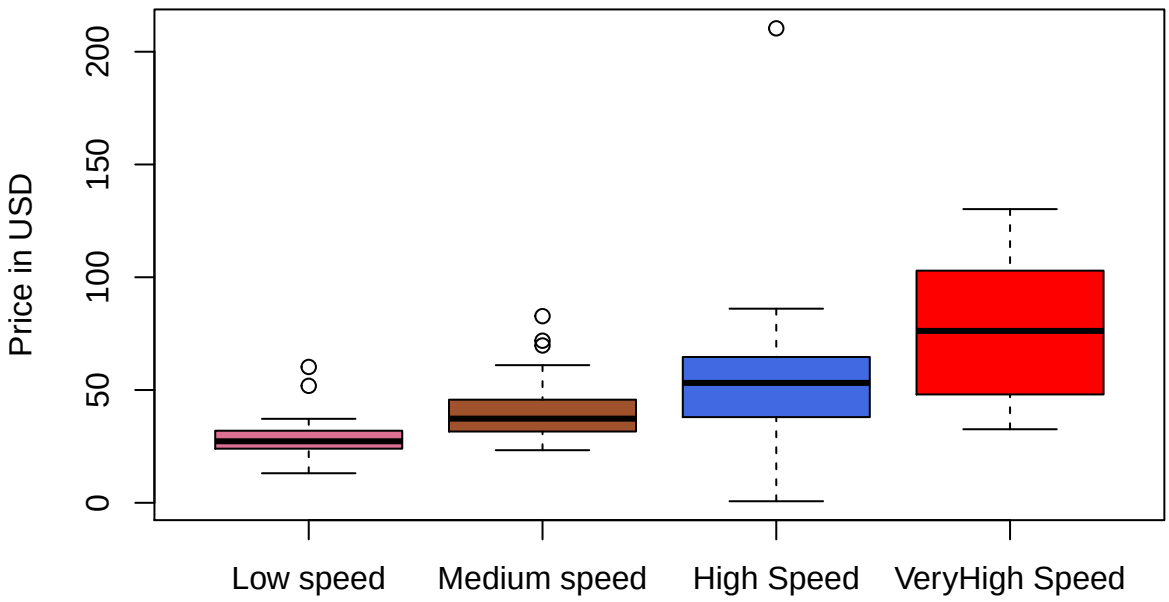
Here is the analysis on the Price dataset.

```
colnames(df_price)[3:6] <- c("Low speed", "Medium speed", "High Speed", "VeryHigh Speed")
summary(df_price)
```

```
##            Country    Country.Code   Low speed      Medium speed
##  Australia     : 1   AT     : 1    Min.   :13.10   Min.   :23.32
##  Austria       : 1   AU     : 1    1st Qu.:24.01   1st Qu.:31.62
##  Belgium       : 1   BE     : 1    Median :27.28   Median :37.32
##  Canada        : 1   CA     : 1    Mean   :29.11   Mean   :41.45
##  Czech Republic: 1   CH     : 1    3rd Qu.:31.96   3rd Qu.:45.71
##  Denmark       : 1   CZ     : 1    Max.   :60.23   Max.   :82.76
##  (Other)       :24   (Other):24    NA's   :1
##    High Speed       VeryHigh Speed
##  Min.   :  0.6931   Min.   : 32.61
##  1st Qu.: 38.1225   1st Qu.: 48.04
##  Median : 53.1600   Median : 76.22
##  Mean   : 55.6437   Mean   : 77.07
##  3rd Qu.: 64.2075   3rd Qu.:102.92
##  Max.   :210.3600   Max.   :130.21
##  NA's   :2          NA's   :11
```

By looking at the min, max and the different Quartiles we can see there are some outliers. Using a boxplot we can clearly see which variable has outliers and how much the deviation is.

```
boxplot(df_price[3:6],ylab = "Price in USD" , col = c("palevioletred","sienna","royalblue","red1"))
```



The max value in the "High Speed" variable is 210.36 for the country Poland, while the mean is 55 and the 3rd Quartile is 64. The min value for the "High speed" vaiable is 0.69 for the country Luxembourg which could be wrong since that country's Low speed and Medium speed prices are 26 and 36 dollars respectively.

Looking at the NA's, there are 11 NAs for the "VeryHigh Speed" which could mean that those 11 countries do not have veryhigh speed broadband services. But the NA in Lowspeed variable for the country Belgium is something we need to be aware of when doing further analysis. Similarly, Mexico and Turkey have NAs in the "High speed" variable.

Looking at our next dataset df_speed:

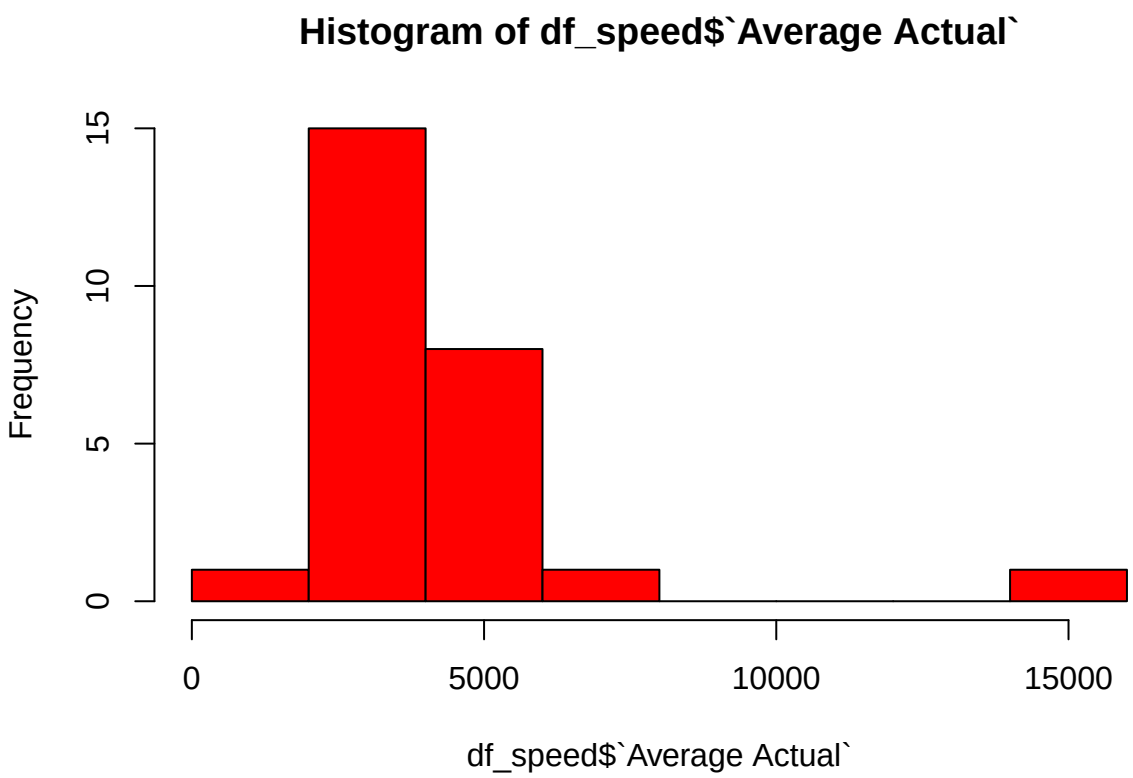
```
colnames(df_speed)[3:17] <- c("Max Advertised", "Average Advertised", "Average Actual", "Average Downloa
summary(df_speed)
```

```
##            Country    Country.Code Max Advertised   Average Advertised
##  Australia     : 1   AT     : 1    Min.   :  4000   Min.   : 1514
##  Austria       : 1   AU     : 1    1st Qu.: 20480   1st Qu.: 7514
##  Belgium       : 1   BE     : 1    Median : 25600   Median :10570
##  Canada        : 1   CA     : 1    Mean   : 43762   Mean   :16765
##  Czech Republic: 1   CH     : 1    3rd Qu.: 50000   3rd Qu.:14500
##  Denmark       : 1   CZ     : 1    Max.   :110000   Max.   :92846
##  (Other)       :24   (Other):24    NA's   :1
##  Average Actual  Average Download  SD download    Average upload
##  Min.   :  948   Min.   : 1377    Min.   : 2473   Min.   : 380.0
##  1st Qu.: 3032   1st Qu.: 4135    1st Qu.: 6469   1st Qu.: 712.5
##  Median : 3780   Median : 5730    Median : 8698   Median : 972.0
##  Mean   : 4205   Mean   : 6729    Mean   :10416   Mean   :1486.8
```

```
##  3rd Qu.: 4474   3rd Qu.: 8415    3rd Qu.:13547   3rd Qu.:1601.5
##  Max.   :15239   Max.   :20493    Max.   :26235   Max.   :5784.0
##  NA's   :4
##    SD upload     Average Latency   SD Latency     Median Download
##  Min.   :  677   Min.   : 69.0   Min.   : 191.0   Min.   : 876
##  1st Qu.: 2664   1st Qu.:107.5   1st Qu.: 254.5   1st Qu.:2521
##  Median : 3140   Median :128.5   Median : 288.0   Median :3330
##  Mean   : 4093   Mean   :145.4   Mean   : 329.2   Mean   :3766
##  3rd Qu.: 4839   3rd Qu.:156.2   3rd Qu.: 324.8   3rd Qu.:4856
##  Max.   :12098   Max.   :464.0   Max.   :1199.0   Max.   :9362
##
##  Median upload    Median Latency   Top 90% Download Top 90% Upload
##  Min.   : 211.0   Min.   : 28.00   Min.   : 2591    Min.   :  521.0
##  1st Qu.: 350.8   1st Qu.: 43.50   1st Qu.: 9218    1st Qu.:  884.5
##  Median : 477.0   Median : 63.00   Median :11916    Median : 1461.0
##  Mean   : 510.2   Mean   : 68.40   Mean   :15221    Mean   : 3048.1
##  3rd Qu.: 631.0   3rd Qu.: 74.75   3rd Qu.:17912    3rd Qu.: 3125.2
##  Max.   :1123.0   Max.   :182.00   Max.   :62301    Max.   :20174.0
##
##  Bottom 10% Latency
##  Min.   : 9.00
##  1st Qu.:12.25
##  Median :19.00
##  Mean   :24.17
##  3rd Qu.:30.50
##  Max.   :79.00
##
```

Let's look at the distribution of the key variable using an histogram:

```
hist(df_speed$`Average Actual`, breaks=10, col="red")
```



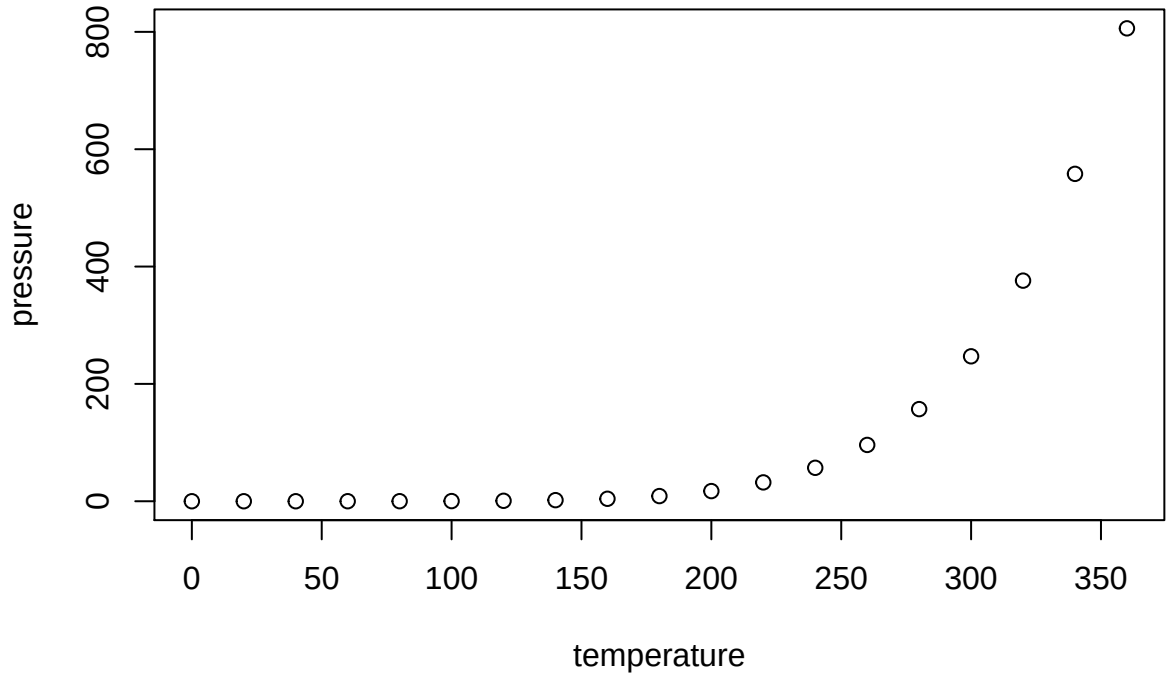
Analysis of Key Relationships

Analysis of Secondary Effects

Conclusion

Including Plots

You can also embed plots, for example:



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.