Mobile Phones Selling Price Report

Proposal for final project (MDSA Winter 2023)

Yongpeng Fu (10182778), Rudy Brown (10057171), Jose Palacios (30190988), Stuart Finley(30191070), Andrii Voitkiv(30199373)

2023-02-07

Contents

Introduction

Mobile phones are everywhere, so are the prices. Despite still having the word "phone" in the name, a typical modern smartphone has much more features than just to make and receive calls. They are boasting a staggering range of applications like brand, memory, storage, camera, resolution, just to name a few. All these cutting edge technology and features packed in one little device does not come without a cost. A 2020 review of premium mobile phones shows a staggering *490%* rise in the last two decades.

With so many mobile phones on the market, it can be difficult to decide which one you want to buy. As a customer, we are particularly interested in finding some relation between all these features and its selling price. To this purpose, we collected the MobilePhone's dataset from Kaggle and apply a set of statistical analysis hoping to answer some guiding questions:

1. Can we estimate the average price for mobile phones?
2. What is the impact of each mobile phone's feature on the selling price?
3. Can a classification model to distinguish the selling price range?
4. Can we build a decent model to predict the selling price for a mobile phone?

Dataset and Cleanup

The initial dataset consists of 8 columns and 28,036 rows and no missing values. These 8 columns are:

- **Model**: categorical variables with sub-classes. These names include the color of the unit and its storage capacity. The latter being also listed as a separate column. - **Independent Variable**
- **Company**: categorical variable. Name of the phone's manufacturer. - **Independent Variable**
- **Price**: continuous variable. Units in Indian Rupees. - **Dependent Variable**
- **Rating**: continuous variable. Units in Indian Rupees. - **Independent Variable**
- **Number of ratings**: discrete variable: a simple count. - **Independent Variable**
- **Total reviews**: discrete variable: a simple count. - **Independent Variable**
- **RAM size**: categorical variable. RAM specification of the phone. - **Independent Variable**
- **ROM size**: categorical variable. Storage (non-volatile memory) capacity of the phone. - **Independent Variable**

Some initial steps can be completed to clean the dataset and create new variables which can be used in our analysis. The initial steps for cleaning the dataset are as follows:

1. Remove any duplicates in the dataset;
2. Because **Model** column contains sub-class of a mobile phone, we decide to further break it down to *Model* and *Color*;
3. Convert all units from **RAM size** and **ROM size** measure to GB and then remove unit suffix;
4. Add additional column to segment the **Price** into 4 different levels;
5. Add additional column to determine if a phone has %G feature or not based on **Model** information.

```r
mobile_dataset <- read.csv("./Updated_Mobile_Dataset.csv")
#Step 0: convert Model and Company letter to uppercase.
mobile_dataset$Model <- toupper(mobile_dataset$Model)
mobile_dataset$Company <- toupper(mobile_dataset$Company)
#Break down Model column into Model only and Color columns
#Step 1: remove the Company name from Model
Model_no_Company <- stringr::str_remove(mobile_dataset$Model,mobile_dataset$Company)
↪  %>% trimws(., which = c("both"))
#Step 2: remove anything after parentheses to get Model only info
mobile_dataset$Model_Only <- stringr::str_replace(Model_no_Company, "
↪  \\s*\\([^\\)]+\\)", "")
```

```r
#Step 3: Get the color information from inside the last ()
#Step 3.1: Get the parenthesis and what is inside from the last ()
Model_no_Company_parenthesis <- stringr::str_extract(Model_no_Company,
↪  "(?<=\\()([^()]*?)(?=\\)[^()]*$)")
#step 3.2: Get the color by just retaining the info before ,
mobile_dataset$Color <- gsub(",.*$", "", Model_no_Company_parenthesis)
#Step 4: Remove duplicated rows in the dataset
mobile_dataset <- mobile_dataset[!duplicated(mobile_dataset),]
#Step 5: cut the price based on the percentile into 4 different levels
mobile_dataset <- mobile_dataset %>% mutate(Price_Level = ntile(Price, n = 4))
#Step 5.1: map each number level to the character
from <- c(1,2,3,4)
to <- c("Low", "Medium", "High", "Very High")
mobile_dataset$Price_Level <- mapvalues(mobile_dataset$Price_Level, from = from, to
↪  = to)


#Step 6: Get the numeric part of RomSize (remove GB and MB, but convert MB to GB),
↪  discard any record that no numeric in RomSize
#Step 6.1: there are some data input errors for RamSize and RomSize. In the records
↪  where RomSize is "Not Known" are swapped with RamSize, so we need to correct
↪  that.
RamSize_temp <- ifelse(mobile_dataset$RomSize == "Not Known", "0 GB",
↪  mobile_dataset$RamSize)
mobile_dataset$RomSize <- ifelse(mobile_dataset$RomSize == "Not Known",
↪  mobile_dataset$RamSize, mobile_dataset$RomSize)
mobile_dataset$RamSize <- RamSize_temp
#Step 6.2: split RomSize into two columns with size number and unit, and convert MB
↪  to 1/1000GB, KB to 1/1000000GB
mobile_dataset$RamSize_Ori <- mobile_dataset$RamSize
mobile_dataset$RomSize_Ori <- mobile_dataset$RomSize
mobile_dataset <- mobile_dataset %>% separate(RomSize, c("RomSize_num",
↪  "RomSize_Unit")) %>% mutate(RomSize_Unit= mapvalues(.$RomSize_Unit, from =
↪  c("GB", "MB", "KB"), to = c(1, 1/1000, 1/1000000)))
```

```
## Warning: Expected 2 pieces. Additional pieces discarded in 1 rows [573].
```

```r
#step 6.3: remove any rows that are not numeric value for RomSize
mobile_dataset <- mobile_dataset[!is.na(as.numeric(mobile_dataset$RomSize_num)),]
```

```
## Warning in `[.data.frame`(mobile_dataset, !
## is.na(as.numeric(mobile_dataset$RomSize_num)), : NAs introduced by coercion
```

```r
mobile_dataset$RomSize_num <- as.numeric(mobile_dataset$RomSize_num)
mobile_dataset$RomSize_Unit <-
↪  ifelse(is.na(as.numeric(mobile_dataset$RomSize_Unit)), 0,
↪  as.numeric(mobile_dataset$RomSize_Unit))
#Step 6.4: generate the final column RomSize_inGB
mobile_dataset$RomSize_inGB <- mobile_dataset$RomSize_num *
↪  mobile_dataset$RomSize_Unit
```

```
#Step 7: Get the numeric part of RamSize (remove GB and MB, but convert MB to GB),
↪  discard any record that no numeric in RamSize
#Step 7.1: split RamSize into two columns with size number and unit, and convert MB
↪  to 1/1000GB
mobile_dataset <- mobile_dataset %>% separate(RamSize, c("RamSize_num",
↪  "RamSize_Unit")) %>% mutate(RamSize_Unit= mapvalues(.$RamSize_Unit, from =
↪  c("GB", "MB"), to = c(1, 1/1000)))
#step 7.2: remove any rows that are not numeric value for RamSize
mobile_dataset <- mobile_dataset[!is.na(as.numeric(mobile_dataset$RamSize_num)),]
mobile_dataset$RamSize_num<- as.numeric(mobile_dataset$RamSize_num)
mobile_dataset$RamSize_Unit <- as.numeric(mobile_dataset$RamSize_Unit)
#Step 7.3: generate the final column RamSize_inGB
mobile_dataset$RamSize_inGB <- mobile_dataset$RamSize_num *
↪  mobile_dataset$RamSize_Unit


#Step 8: Create a new column to determine if the phone is 5G or not
mobile_dataset$Is_5G <- ifelse(str_detect(mobile_dataset$Model_Only, "5G"), "Yes",
↪  "No")
#Step 9: only keep the columns we need
column_names <- c("Model", "Company", "Price", "Rating", "No_of_ratings",
↪  "TotalRevivews", "Model_Only", "Color", "Price_Level", "RamSize_inGB",
↪  "RomSize_inGB", "RamSize_Ori", "RomSize_Ori", "Is_5G" )
mobile_dataset <- mobile_dataset[column_names]

# #Step 9: final check up. Convert all company name to uppercase and then do a final
↪  duplicated removal
# mobile_dataset$Company <- toupper(mobile_dataset$Company)
# mobile_dataset <- mobile_dataset[!duplicated(mobile_dataset),]
write.csv(mobile_dataset, file = './Cleaned_Mobile_Dataset.csv', row.names = F)
```

After cleaning and breaking down columns, the dataset now consists of 14 columns and 725 rows and no missing values. These 14 columns are:

- **Model**: categorical variables with sub-classes. These names include the color of the unit and its storage capacity. The latter being also listed as a separate column. - **Independent Variable**
- **Company**: categorical variable. Name of the phone's manufacturer. - **Independent Variable**
- **Price**: continuous variable. Units in Indian Rupees. - **Dependent Variable**
- **Rating**: continuous variable. Units in Indian Rupees. - **Independent Variable**
- **Number of ratings**: discrete variable: a simple count. - **Independent Variable**
- **Total reviews**: discrete variable: a simple count. - **Independent Variable**
- **Model_Only**: categorical variable: only contains the model information of a mobile phone. - **Independent Variable**
- **Color**: categorical variable: color of a mobile phone. - **Independent Variable**
- **Price_Level**: The price level of a mobile phone, with levels of "Low", "Medium", "High", "Very High". - **Independent Variable**
- **RamSize_inGB**: continuous variable. RAM specification of the phone in GB. - **Independent Variable**
- **RomSize_inGB**: continuous variable. Storage (non-volatile memory) capacity of the phone in GB. - **Independent Variable**
- **RamSize_Ori**: categorical variable. RAM specification of the phone, original information. - **Independent Variable**

- **RomSize_Ori**: categorical variable. Storage (non-volatile memory) capacity of the phone, original information. - **Independent Variable**
- **Is_5G**: categorical variable. If the phone has 5G service or not. - **Independent Variable**

```
mobile_dataset <- as_tibble(read.csv("./Cleaned_Mobile_Dataset.csv"))
#Specify the level for Price_Level column
mobile_dataset$Price_Level <- factor(mobile_dataset$Price_Level, levels = c("Low",
↪  "Medium", "High", "Very High"))
mobile_dataset %>% head(4)
```

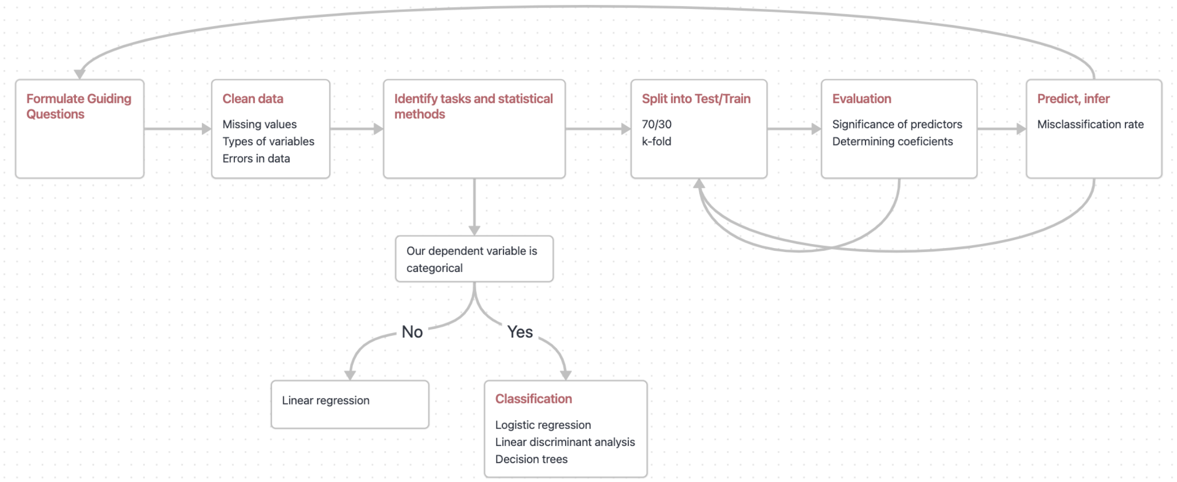
```
## # A tibble: 4 x 14
##   Model          Company Price Rating No_of_ratings TotalReviwes Model_Only Color
##   <chr>          <chr>   <int>  <dbl>         <int>        <int> <chr>      <chr>
## 1 INFINIX HOT~ INFINIX  8199    4.3           505           52 HOT 20 PL~ LUNA ~
## 2 MOTOROLA E4~ MOTORO~  7999    4.1         56085         5600 E40        CARBO~
## 3 MOTOROLA E4~ MOTORO~  7999    4.1         56085         5600 E40        PINK ~
## 4 POCO C31 (S~ POCO     7499    4.3        183688        11185 C31        SHADO~
## # ... with 6 more variables: Price_Level <fct>, RamSize_inGB <dbl>,
## #   RomSize_inGB <dbl>, RamSize_Ori <chr>, RomSize_Ori <chr>, Is_5G <chr>
```

Table 1: The cleaned-up dataset for Mobile Phone

The dataset and detailed analysis can be found at this repository.

Scope of Analysis

Our team is finalizing what the full analysis of the dataset will look like, but a preliminary template and breakdown of work by team member has been included below. The different colors represent which components of the project different team members would take on. It is anticipated that all members will assist in the finalization of the report.



Chapter 4: Exploratory Data Analysis

4.1: A summary of the dataset

The dimension of the cleaned dataset is 726 rows and 14 columns. A summary of the data is as follows:

```r
mobile_dataset <- as_tibble(read.csv("./Cleaned_Mobile_Dataset.csv"))
#Specify the level for Price_Level column
mobile_dataset$Price_Level <- factor(mobile_dataset$Price_Level, levels = c("Low",
↪  "Medium", "High", "Very High"))
mobile_dataset %>% summary()
```

```
##     Model              Company              Price            Rating
##  Length:725         Length:725         Min.   :   698   Min.   :2.700
##  Class :character   Class :character   1st Qu.:  7014   1st Qu.:4.100
##  Mode  :character   Mode  :character   Median : 12889   Median :4.200
##                                        Mean   : 15903   Mean   :4.227
##                                        3rd Qu.: 16999   3rd Qu.:4.300
##                                        Max.   :149900   Max.   :4.800
##  No_of_ratings     TotalReviwes     Model_Only          Color
##  Min.   :     3   Min.   :    0   Length:725         Length:725
##  1st Qu.:   874   1st Qu.:   80   Class :character   Class :character
##  Median :  7325   Median :  670   Mode  :character   Mode  :character
##  Mean   : 34522   Mean   : 2496
##  3rd Qu.: 37211   3rd Qu.: 2972
##  Max.   :575907   Max.   :33942
##     Price_Level   RamSize_inGB     RomSize_inGB    RamSize_Ori
##  Low      :171   Min.   : 0.000   Min.   :  0.00   Length:725
##  Medium   :185   1st Qu.: 0.064   1st Qu.: 32.00   Class :character
##  High     :185   Median : 4.000   Median : 64.00   Mode  :character
##  Very High:184   Mean   : 3.882   Mean   : 81.78
##                  3rd Qu.: 6.000   3rd Qu.:128.00
##                  Max.   :12.000   Max.   :512.00
##  RomSize_Ori           Is_5G
##  Length:725         Length:725
##  Class :character   Class :character
##  Mode  :character   Mode  :character
##
##
##
```

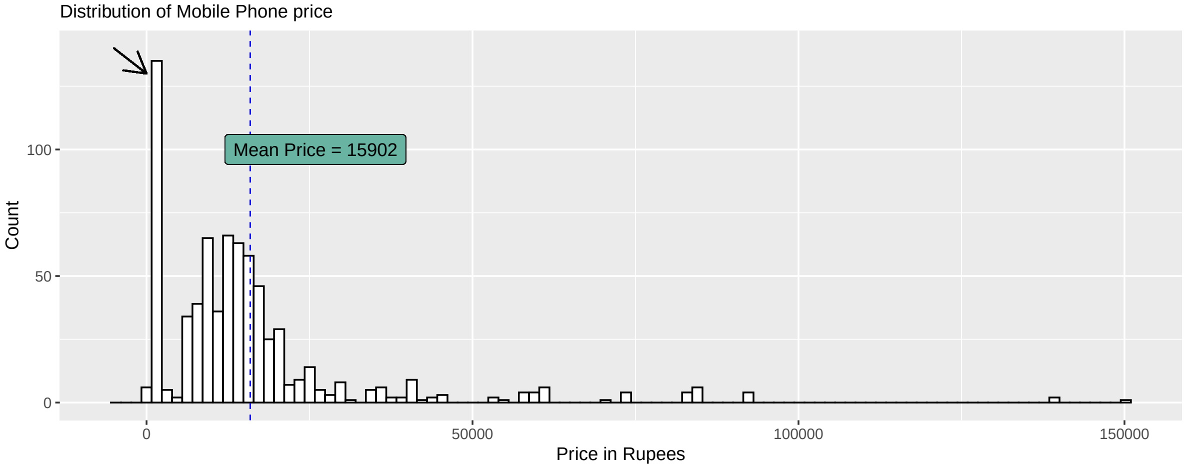
Table 2: A summary for Mobile Phone dataset

4.2: Exploratory Data Analysis

This section is focused on the exploration of relation between variables using various visualization techniques.

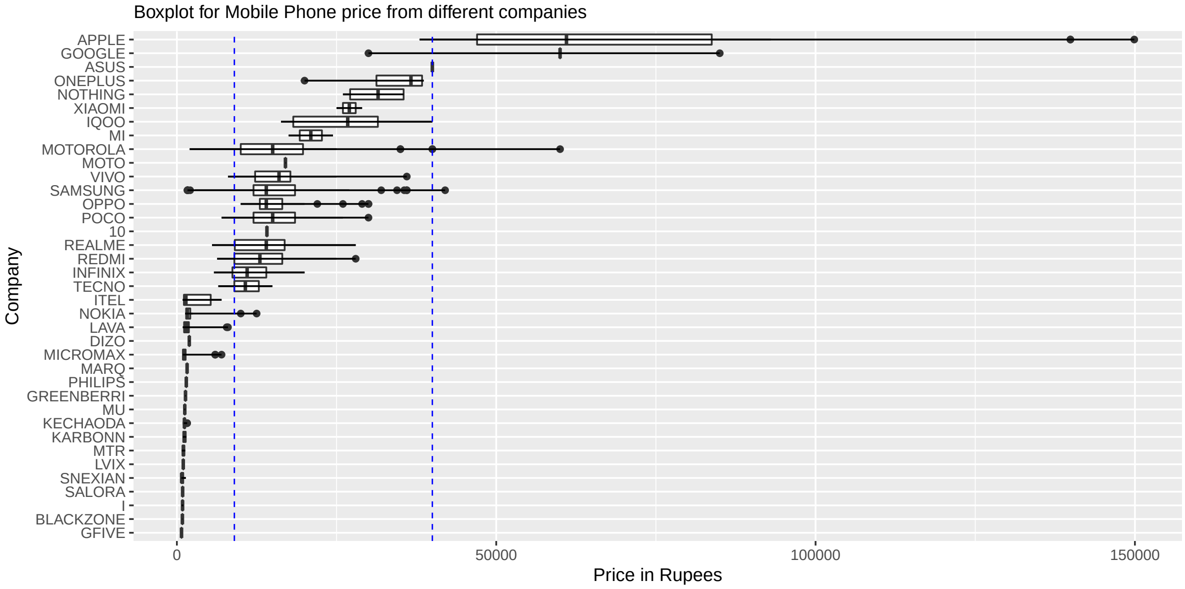
The main target is Price in the dataset. In the following visualization it shows the distribution of Mobile Phone price with long tail towards to the high end. The average price is 15902 Rupees. Most price falls in the range of 50000 Rupees. One really stands out price range (arrow indicated) is 1000-2000 Rupees with the highest frequency.

```
ggplot(data = mobile_dataset, mapping = aes(x=Price)) +
↪   geom_histogram(color="black", fill="white", bins = 100)+
  geom_vline(aes(xintercept=mean(Price)),
            color="blue", linetype="dashed", size=0.4) + geom_label(
    label="Mean Price = 15902",
    x= mean(mobile_dataset$Price) + 10000,
    y=100,
    label.padding = unit(0.35, "lines"),
    label.size = 0.25,
    color = "black",
    fill="#69b3a2"
  ) +  geom_segment(aes(x = -5000, y = 140, xend = 0, yend = 130),lineend =
↪   "round",linejoin = "round",
                  arrow = arrow(length = unit(0.5, "cm"))) + labs(y="Count",
                  ↪   x="Price in Rupees",
       subtitle="Distribution of Mobile Phone price")
```



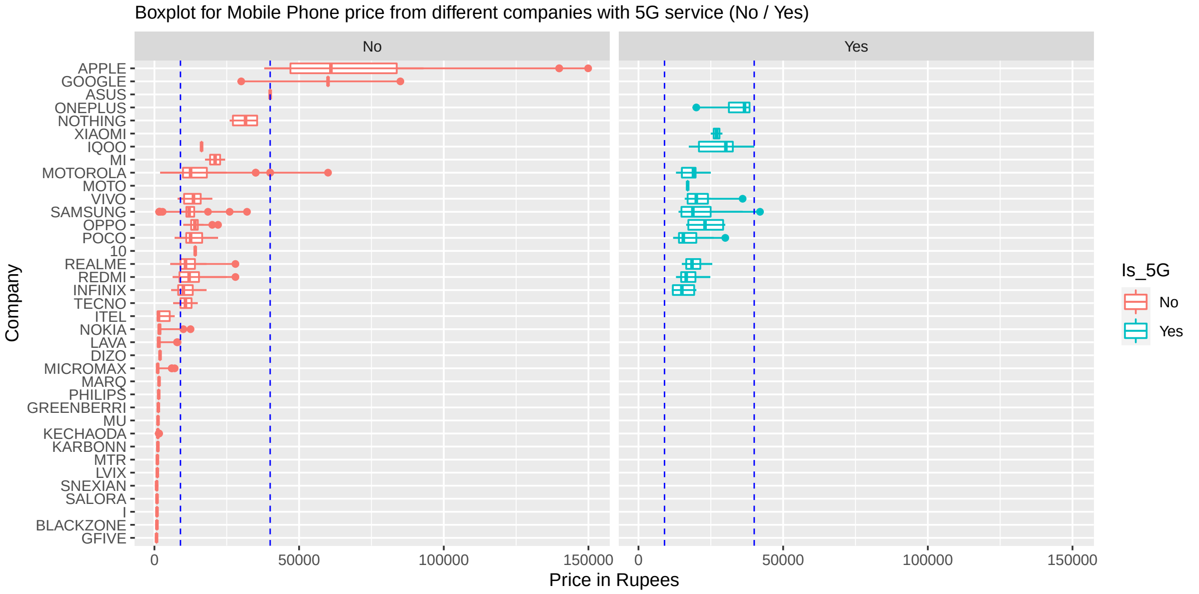
The following figure is to break down the Price by different phone-making companies. There are a total of 37 companies in the dataset. Iin consistent with previous figure about Price distribution, we see a big portion of Price falls between 9000 (left dash line) and 40000 (right dash line) Rupees. *Apple* alone contributes the most of high priced Mobile Phones, while companies like from *NOKIA* to *GFIVE* contribute to the low-priced ones.

```
ggplot(data = mobile_dataset, mapping = aes(y = reorder(Company, Price), x = Price))
↪   +geom_boxplot()+ geom_line() +
    sapply(c(9000, 40000), function(xint) geom_vline(aes(xintercept = xint),
↪   color="blue", linetype="dashed", size=0.4)) + labs(y="Company", x="Price in
↪   Rupees",
       subtitle="Boxplot for Mobile Phone price from different companies")
```



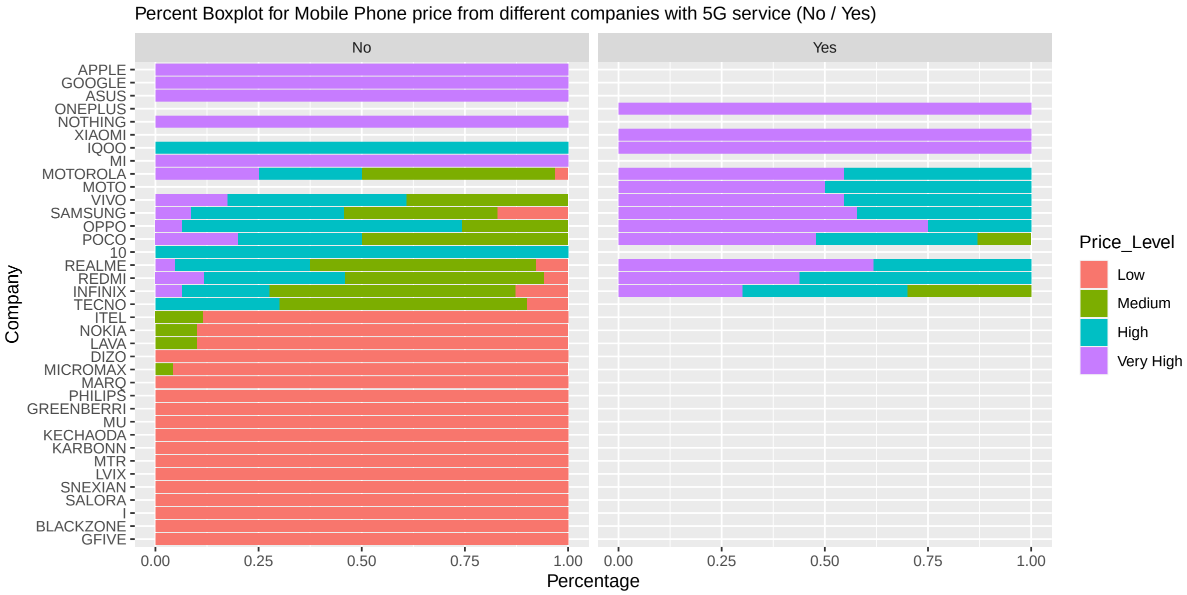
The Price is further broken down by their 5G services. Interestingly, neither high-priced nor low-priced Mobile Phones have equipped 5G service. In contrast, almost all middle-priced Mobile Phones have 5G service. Price for those with 5G service are slightly more expensive than those without. NOTE: the dashed line indicates the price between 9000 (left dash line) and 40000 (right dash line) Rupees.

```
ggplot(data = mobile_dataset, mapping = aes(y = reorder(Company, Price), x = Price,
↪  color = Is_5G)) +geom_boxplot(position=position_dodge(width = 0.8))+ geom_line()
↪  +
   sapply(c(9000, 40000), function(xint) geom_vline(aes(xintercept = xint),
↪  color="blue", linetype="dashed", size=0.4)) + labs(y="Company", x="Price in
↪  Rupees",
      subtitle="Boxplot for Mobile Phone price from different companies with 5G
      ↪  service (No / Yes)") + facet_wrap(~Is_5G)
```



The pattern is more obvious when we provide color for 4 different price level (“Low”, “Medium”, “High”, “Very High”). Although mobile phones from companies like APPLE, GOOGLE, ASUS and NOTHING have no 5G service, their price are all very high. It is worth investigating if some other attributes like brand effect, Storage capacity (Rom Size), and calculation power (Ram Size) are playing roles. Most Mobile Phones (except MI) with 5G service, again, have a higher proportion falling in a Very High price level, compared to the counterparts without 5G service. Almost all the rest mobile phones without 5G service fall in a Low price level.

```
ggplot(data = mobile_dataset,mapping = aes(y =reorder(Company, Price), fill =
↪  Price_Level)) + geom_bar(stat = "count", position="fill") + labs(y="Company",
↪  x="Percentage",
       subtitle="Percent Boxplot for Mobile Phone price from different companies
        ↪  with 5G service (No / Yes)") + facet_wrap(~Is_5G)
```



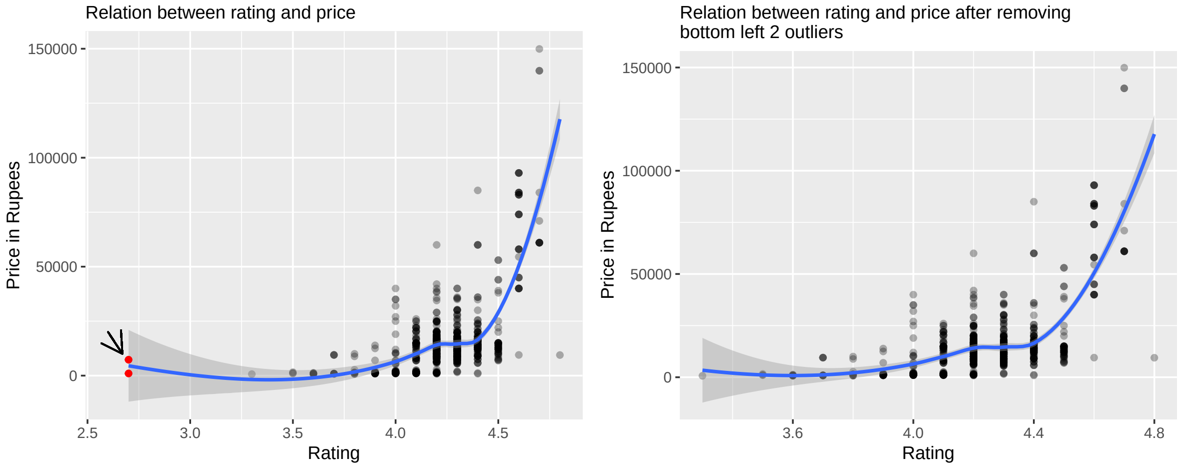
Next, we explored some quantitative features with respect to the Mobile Phone price. The first scatter plot shows an obvious positive relation between Rating and Price as the smooth line indicates. The higher the Rating, the higher the Price. And this holds when we remove those bottom left 2 outlier data points.

```
#The first plot is one with outlier
plot1 = ggplot(data = mobile_dataset,mapping = aes(x = Rating, y = Price)) +
↪  geom_point(alpha=0.3) + geom_smooth(se=T) +
  geom_segment(aes(x = 2.6, y = 20000, xend = 2.67, yend = 10000),lineend =
↪  "round",linejoin = "round",arrow = arrow(length = unit(0.5, "cm"))) +
  geom_point(data=mobile_dataset[mobile_dataset$Rating<3.0,], aes(x = Rating, y =
↪  Price), color='red') +
  labs(y="Price in Rupees", x="Rating", subtitle="Relation between rating and
↪  price")

#The first plot is a one with outliers removed
plot2 = ggplot(data = mobile_dataset[mobile_dataset$Rating>3.0, ],mapping = aes(x =
↪  Rating, y = Price)) + geom_point(alpha=0.3) + geom_smooth(se=T) +
  labs(y="Price in Rupees", x="Rating", subtitle="Relation between rating and price
↪  after removing \nbottom left 2 outliers")
```

```
grid.arrange(plot1, plot2, ncol=2)
```

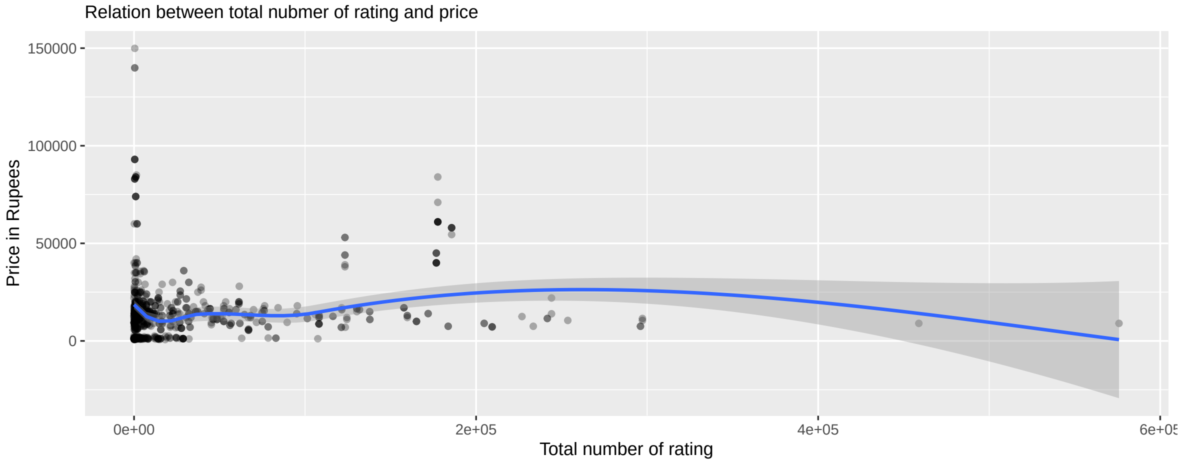
```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



The second scatter plot shows the relation between Total number of rating and Price. As the smooth line indicates, there is no obvious correlation between them. The result holds even when we remove some seemingly outliers on the far right end.

```
ggplot(data = mobile_dataset,mapping = aes(x = No_of_ratings, y = Price)) +
↪   geom_point(alpha=0.3) + geom_smooth()+ labs(y="Price in Rupees", x="Total number
↪   of rating", subtitle="Relation between total nubmer of rating and price")
```

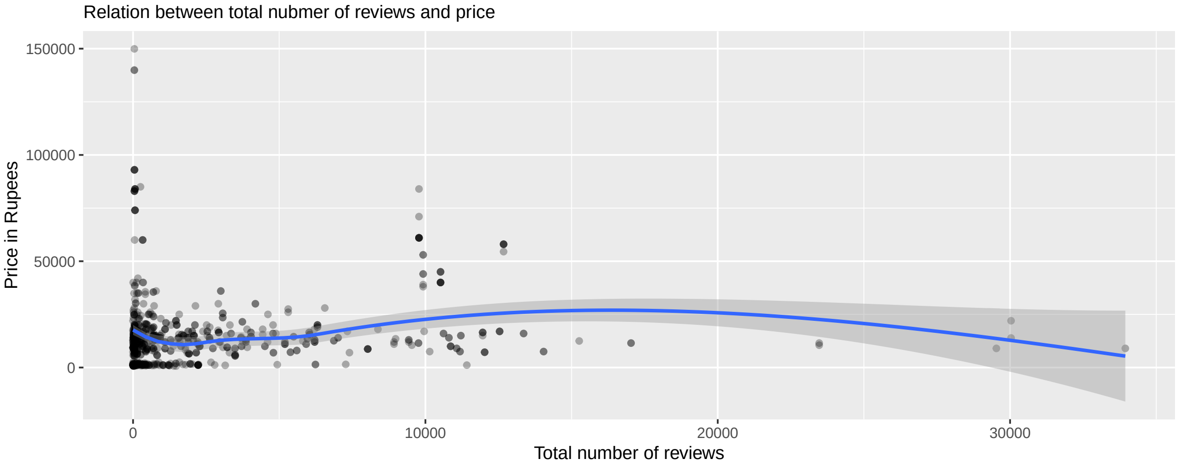
```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



The same pattern is also observed in the third scatter plot where relation between Total number of reviews and Price is plotted. As the smooth line indicates, there is no obvious correlation between them. The result holds even when we remove some seemingly outliers on the far right end.

```
ggplot(data = mobile_dataset,mapping = aes(x = TotalReviwes, y = Price)) +
↪   geom_point(alpha=0.3) + geom_smooth()+ labs(y="Price in Rupees", x="Total number
↪   of reviews", subtitle="Relation between total nubmer of reviews and price")
```

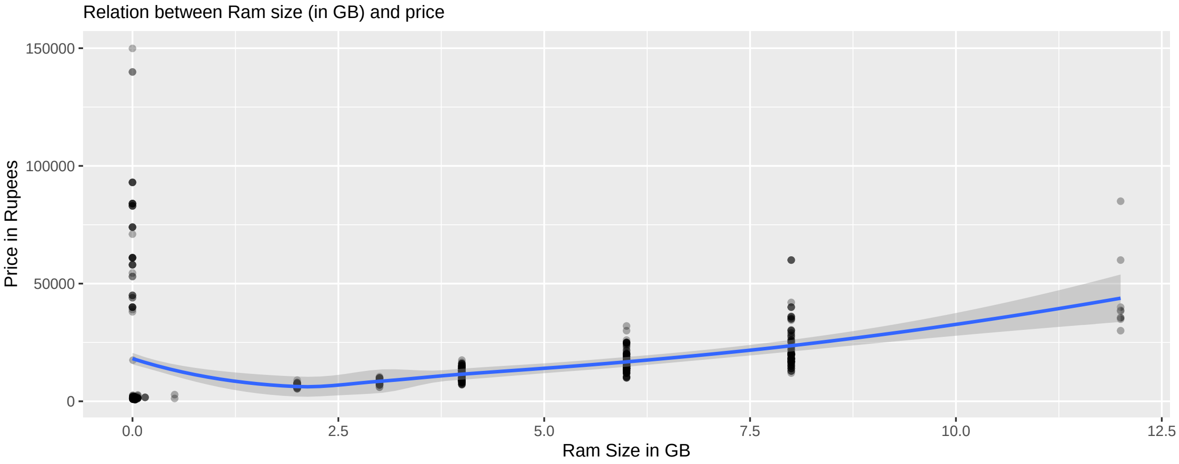
```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



In the fourth scatter plot, below, there seems a slight positive relation between Ram size (in GB) and the Price. Ram is normally associated with speed and performance of an operating system. The higher ram is, the better it is in speed and performance. It makes sense that a mobile phone with a larger Ram will charge more. However, because nowadays most mobile phones are very fast and stable, people wont tell too much difference, making the added on value from ram is only weekly associated with price.

```
ggplot(data = mobile_dataset,mapping = aes(x = RamSize_inGB, y = Price)) +
↪  geom_point(alpha=0.3) + geom_smooth(se=T)+ labs(y="Price in Rupees", x="Ram Size
↪  in GB", subtitle="Relation between Ram size (in GB) and price")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



In the final scatter plot, again, we have two charts, one the original dataset, the other with outlier removed. Both cases tells the same story. In contrast to the previous scatter plot (Ram size (in GB) and the Price), there is a very obvious positive linear relation between Rom size (in GB) and the Price. This makes sense because often times a mobile phone is more limited by its non-volatile memory space than its speed & performance.

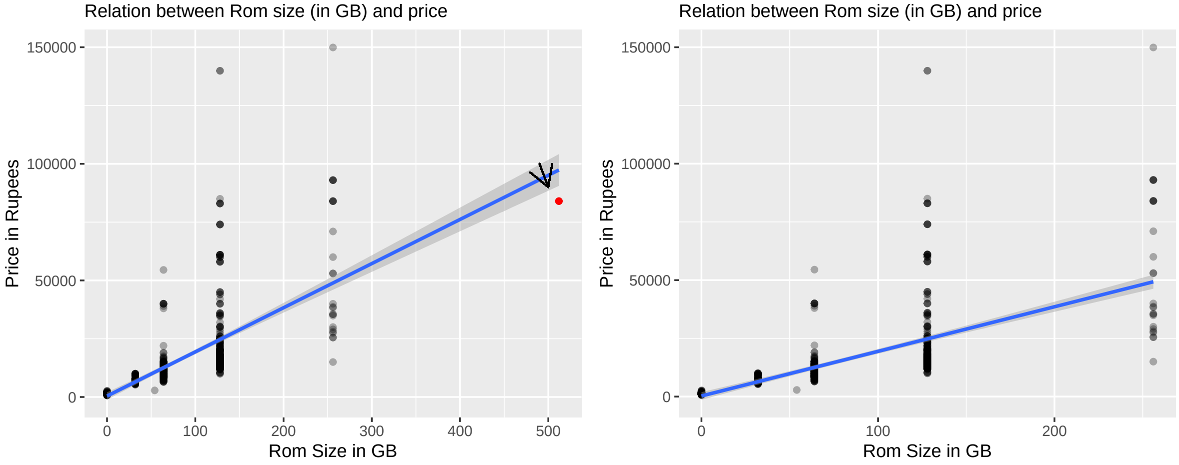
```
#The first plot is a one with outliers
plot1 = ggplot(data = mobile_dataset,mapping = aes(x = RomSize_inGB, y = Price)) +
↪  geom_point(alpha=0.3) + geom_smooth(se=T, method = lm)+
  geom_segment(aes(x = 490, y = 100000, xend = 500, yend = 90000),lineend =
↪  "round",linejoin = "round",arrow = arrow(length = unit(0.5, "cm"))) +
```

```r
  geom_point(data=mobile_dataset[mobile_dataset$RomSize_inGB>400,], aes(x =
↪   RomSize_inGB, y = Price), color='red') +
  labs(y="Price in Rupees", x="Rom Size in GB", subtitle="Relation between Rom size
↪   (in GB) and price")

#The second plot is a one with outliers removed
plot2 = ggplot(data = mobile_dataset[mobile_dataset$RomSize_inGB<400, ],mapping =
↪   aes(x = RomSize_inGB, y = Price)) + geom_point(alpha=0.3) +
↪   geom_smooth(se=T,method = lm) +
  labs(y="Price in Rupees", x="Rom Size in GB", subtitle="Relation between Rom size
↪   (in GB) and price")

grid.arrange(plot1, plot2, ncol=2)
```

```
## `geom_smooth()` using formula 'y ~ x'
## `geom_smooth()` using formula 'y ~ x'
```



References

2020 review. “High-End Mobile Phones Price Have Soared 490% in 20 Years | This Is Money.” This Is Money, This Is Money, 23 July 2020, https://www.thisismoney.co.uk/money/bills/article-8548235/High-end-mobile-phones-price-soared-490-20-years.html.

MobilePhone’s dataset. “MobilePhone’s Dataset | Kaggle.” Kaggle: Your Machine Learning and Data Science Community, Kaggle, 20 Dec. 2022, https://www.kaggle.com/datasets/sudhanshuy17/mobilephone.