Capital One Challenge

Ajinkya Tope

November 25, 2018

1. Introduction

In this data challenge, I have created a data product which determines the most profitable zipcodes in New York City for short term rentals. This data product is scalable and will work fine for new market as well.

Assumptions

1. Occupancy rate is assumed to be 75%
2. Each year comprises of 365 days
3. The investor will pay for the property in cash (i.e. no mortgage/interest rate will need to be accounted for).
4. The time value of money discount rate is 0% (i.e. $1 today is worth the same 100 years from now).
5. All properties and all square feet within each locale can be assumed to be homogeneous (i.e. a 1000 square foot property in a locale such as Bronx or Manhattan generates twice the revenue and costs twice as much as any other 500 square foot property within that same locale.)

Approach

1. Match the two datasets for latest scrapped date, so that we are referring to costs and prices of the properties for same time period.
2. Calculate average Cost of properties for each zipcode.
3. Calculate average prices of properties for each zipcode.
4. Calculate Return on Investment on yearly basis for each zipcode.
5. Determine the most profitable zipcodes.

2. Packages required for analysis

```
#install.packages("ggplot2")
#install.packages("kernlab")
#install.packages("VIM")
#install.packages("magrittr")
#install.packages("dplyr")
#install.packages("stringr")
#install.packages("knitr")

library("ggplot2")
library("kernlab")
library("VIM")
library("magrittr")
library("dplyr")
library("stringr")
library("knitr")
```

3. Reading Input Files

airbnb : we read the given revenue data from airbnb into this variable zillow : we read the cost data, which provides an estimate of value for 2 bedroom properties into this variable

```
airbnb <- read.csv("C:/Study/Competition/Capital One/listings.csv", header=TRUE)
zillow <- read.csv("C:/Study/Competition/Capital One/Zip_Zhvi_2bedroom.csv", header=TRUE)
```

4. Data Quality Check of the datasets

Missing Values in Airbnb dataset

1. There are many missing values across the columns in Airbnb dataset.
2. The columns important for our analysis i.e. price, zipcode, city , state and bedrooms does not have or have very less missing values.

```
sapply(airbnb, function(x) sum(length(which(is.na(x)))))
```

Missing values in Zillow dataset

1. Columns from April, 1996 to July, 2013 have a lot of missing values.
2. There are no missing values after August, 2013 except for Feb,March,April and May months of year 2016.

```
sapply(zillow, function(x) sum(length(which(is.na(x)))))
```

Zipcodes

As the two datasets will be merged on the basis of zipcodes, it is important to have a look at the zipcodes given in both the datasets.

Zipcodes in Airbnb dataset

1. There are 95 columns and contains 40753 observations in airbnb dataset
2. It can be noted that airbnb data is specific to New York only.

```
str(airbnb)
head(airbnb)
```

1. There are about 205 unique zipcodes in airbnb dataset
2. All zipcodes are not in standard format, some of the zipcodes have area code attached to it.

```
length(unique(airbnb$zipcode))
```

```
## [1] 205
```

```
airbnb$zipcode
```

After converting the zipcodes in standard format, airbnb data contains 200 unique zipcodes

```
length(unique(str_replace(airbnb$zipcode, "\\-.*$", "")))
```

```
## [1] 200
```

Zipcodes in zillow dataset

1. There are 262 columns and contains 8946 observations in airbnb dataset
2. zipcodes in zillow dataset are given under the column name of “RegionName”

```
str(zillow)
head(zillow)
```

1. Zillow data set is not specific to New York, and hence we will have to filter the data for New York.

```
temp <- zillow[zillow$City == 'New York',]
length(unique(temp$RegionName))
```

```
## [1] 25
```

1. After filtering the zillow data for New York, we observe that it has only 25 unique zipcodes.
2. All the zipcodes are in standard format
3. The number of zipcodes in zillow data is very less as compared to the number of zipcodes given in Airbnb data.

5. Preprocessing of Data

1. In order to correctly calculate profit, it is important to make sure that we compare data of same timeline.

Function for extracting latest scrapped data from airbnb data

1. This function will return the latest scrapped date for the zipcodes from Airbnb data.

2. This date then can be used to match with the years column in zillow data to keep only the relevant column for cost.

```
zipcodes_last_scrapped <- function(){
  zipcodes_list <- zillow$RegionName
  airbnb <- airbnb
  airbnb$calendar_last_scraped <- as.Date(airbnb$calendar_last_scraped)
  pincodes_last_scrapped <- airbnb %>%
    filter(airbnb$zipcode %in% zipcodes_list) %>% #Filtering NYC zipcodes
    group_by(zipcode) %>%
    summarise(last_scraped = max(format(as.Date(airbnb$calendar_last_scraped, "%m/%d/%Y"),"X%Y.%m"))) %>% # changing format of date a
ccording to Time format in Zillow data
    select(last_scraped, zipcode)
  last_scraped <- max(pincodes_last_scrapped$last_scraped) #taking the latest last scraped
  return(last_scraped)
}
```

As we can see May, 2017 is the latest scrapped date in airbnb dataset. So, now we need to consider the column for only May, 2017 in zillow dataset. This will make sure that we are looking at Cost and Price for the same time period.

```
zipcodes_last_scrapped()
```

```
## [1] "X2017.05"
```

Function for filtering zillow data by last scrapped date

1. This function will take last scrapped date as input and filter the zillow data to keep Cost information only for May,2017
2. It will also filter the zillow dataset by city, so that the output has the data only for New York.

The output of this function will return a dataframe which will have only the important columns, that are required for our further analysis.

```
get_ny_zillow <- function(city){
  zillow <- zillow
  airbnb <- airbnb
  last_scraped <- zipcodes_last_scrapped()
  zillow_ny <- zillow %>%
    distinct() %>%
    filter(City == city) %>% #filtering for NYC
    select(RegionID,RegionName,City,State,
           Metro,CountyName,last_scraped) # selecting appropriate columns
  colnames(zillow_ny)<- c("RegionID","RegionName","City","State","Metro","CountyName","Cost")
  return(zillow_ny)
}
```

1. The dataframe returned by the function has only 25 observations.

```
zillow <- get_ny_zillow("New York")
head(zillow)
```

```
##   RegionID RegionName     City State     Metro CountyName     Cost
## 1    61639      10025 New York    NY New York   New York 1390000
## 2    61637      10023 New York    NY New York   New York 2095000
## 3    61703      10128 New York    NY New York   New York 1720500
## 4    61625      10011 New York    NY New York   New York 2419700
## 5    61617      10003 New York    NY New York   New York 2109100
## 6    62012      11201 New York    NY New York      Kings 1407300
```

Function for cleaning and merging the two datasets

1. The function takes airbnb and zillow dataset as input and returns the final data, that will be used for the analysis.
2. The function also filters the airbnb data for 2 bedrooms and keeps only the columns which are important for our further analysis.
3. It performs cleaning for price column and removes special charcters.Columns are also converted to appropriate data types

```
data_prep <- function(){
  zillow <- zillow
  airbnb_2 <- airbnb[airbnb$bedrooms == 2,]
  airbnb_2 <- airbnb_2[,c("price","zipcode")]
  finaldata <- merge(zillow,airbnb_2,by.x = 'RegionName',by.y ='zipcode') #joining data on zipcodes
  finaldata$price <- str_replace(finaldata$price, "[$]", " ") #Removing $
  finaldata$price <- str_replace(finaldata$price, ",", "") #Removing ','
  finaldata$price <- as.numeric(finaldata$price) #changing format of price to numeric
  finaldata$RegionName <- as.factor(finaldata$RegionName) #changing data type to factor
  finaldata$RegionID <- as.factor(finaldata$RegionID) #changing data type to factor
  finaldata$City <- as.character(finaldata$City) #changing data type to character
  finaldata$State <- as.character(finaldata$State) #changing data type to character
  finaldata$Metro <- as.character(finaldata$Metro) #changing data type to character
  finaldata$CountyName <- as.character(finaldata$CountyName) #changing data type to character
  return(finaldata)
}
```

1. The final data has only 1238 observations and 10 columns
2. The final table has information of cost and price, which can be further used to calculate Return on Investment(ROI)

```
data <- data_prep()
head(data)
```

```
##   RegionName RegionID     City State    Metro CountyName    Cost price
## 1      10003    61617 New York    NY New York   New York 2109100   275
## 2      10003    61617 New York    NY New York   New York 2109100    99
## 3      10003    61617 New York    NY New York   New York 2109100   195
## 4      10003    61617 New York    NY New York   New York 2109100   165
## 5      10003    61617 New York    NY New York   New York 2109100   496
## 6      10003    61617 New York    NY New York   New York 2109100   300
```

6. Exploratory Analysis

From preliminary analysis, we can note that: Cost: The mean Cost of the properties is 1803035. The mean Cost is greater than median, this indicates that the data is right skewed. Price: The range of price is very huge.It varies from 28 to 4700. The mean is greater than median, suggesting a right skewness. RegionName: Zipcodes with highest number of properties are 11215, 10003, 10025, 10036 and 10011.

```
(summary(data))
```

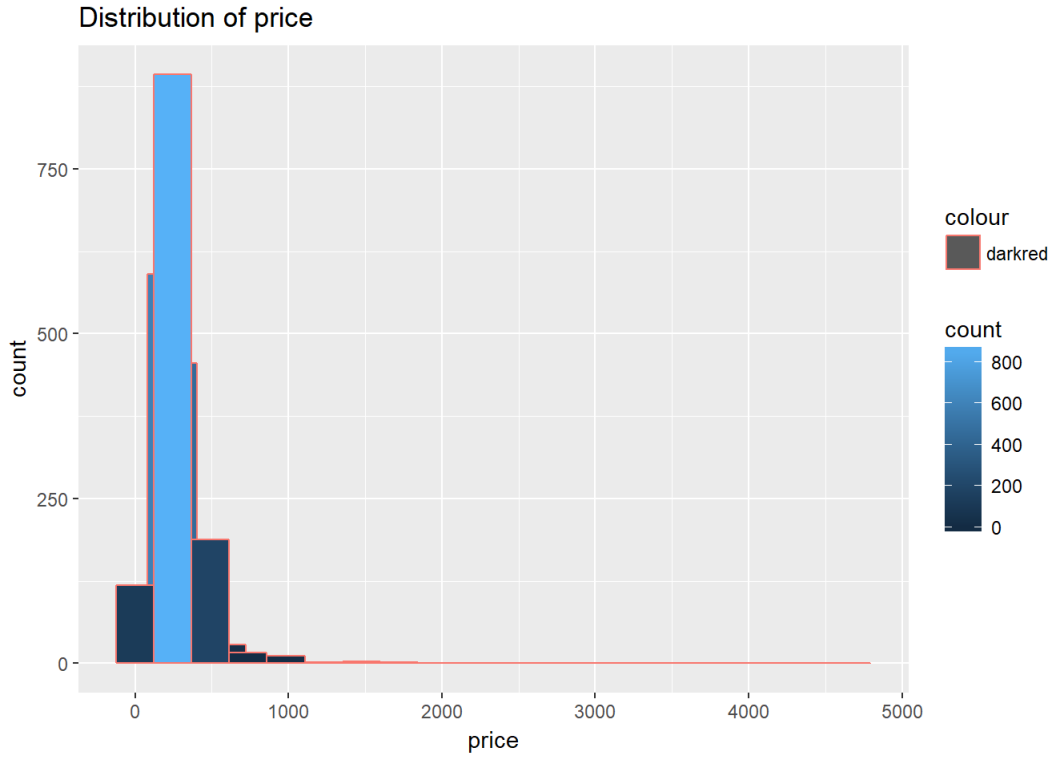
```
##    RegionName     RegionID       City              State
##  11215  :141   62026  :141   Length:1238        Length:1238
##  10003  :133   61617  :133   Class :character   Class :character
##  10025  :112   61639  :112   Mode  :character   Mode  :character
##  10036  :108   61650  :108
##  10011  :102   61625  :102
##  10014  : 95   61628  : 95
##  (Other):547   (Other):547
##     Metro           CountyName            Cost             price
##  Length:1238        Length:1238        Min.   : 321300   Min.   :  28.0
##  Class :character   Class :character   1st Qu.:1276400   1st Qu.: 165.0
##  Mode  :character   Mode  :character   Median :1720500   Median : 240.0
##                                        Mean   :1803035   Mean   : 278.7
##                                        3rd Qu.:2109100   3rd Qu.: 325.0
##                                        Max.   :3262200   Max.   :4700.0
##
```

1. As there is high variation in Price we need to remove outliers before we proceed with the analysis.
2. It can be noted that there are many outliers on the upper end for Price. So we need to write a function to remove outliers from the dataset.

```
boxplot(data$price, main="Distribution of Price", col=c("orange"), xlab = "Price" ,ylim=c(0,1000))
```


```
qplot(x = data$price, color="darkred", fill=..count.., geom="histogram") +
  ggtitle("Distribution of price") +xlab("price") + stat_bin(bins=20)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Function for Outlier Removal

This function will remove the outliers and return the final dataset

```
outliers_m <- function(column)
{
  finaldata <- data
  lowerq <- as.vector(quantile(column)[2]) # returns 1st quartile
  upperq <- as.vector(quantile(column)[4]) # returns 1st quartile
  iqr <- upperq-lowerq

  # Moderate outliers

  mod.outliers.upper <- (iqr * 1.5) + upperq
  mod.outliers.lower <- lowerq - (iqr * 1.5)
  mod.outliers <- which(column > mod.outliers.upper |
                        column < mod.outliers.lower)

  finaldata <- finaldata[-mod.outliers,]
  return(finaldata)
}
```

1. After removing the outliers, 1181 observations are left in the final data

```
finaldata <- outliers_m(data$price)
str(finaldata)
```

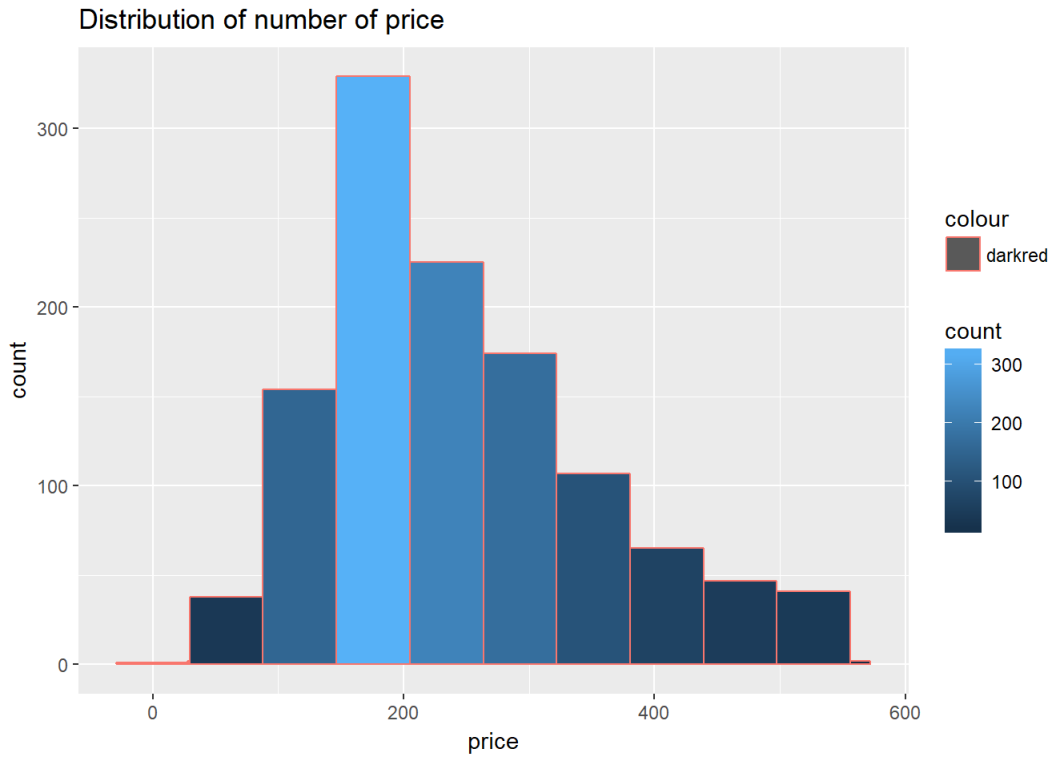
```
## 'data.frame':    1181 obs. of  8 variables:
##  $ RegionName: Factor w/ 22 levels "10003","10011",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ RegionID  : Factor w/ 22 levels "61617","61625",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ City      : chr  "New York" "New York" "New York" "New York" ...
##  $ State     : chr  "NY" "NY" "NY" "NY" ...
##  $ Metro     : chr  "New York" "New York" "New York" "New York" ...
##  $ CountyName: chr  "New York" "New York" "New York" "New York" ...
##  $ Cost      : int  2109100 2109100 2109100 2109100 2109100 2109100 2109100 2109100 2109100 2109100 ...
##  $ price     : num  275 99 195 165 496 300 280 199 450 199 ...
```

Distribution of Number of Price

1. After the removal of outliers, the distribution of price looks more normal now.
2. Most of the properties have prices around 200 - 300 dollars.

```
qplot(x = finaldata$price, color="darkred", fill=..count.., geom="histogram") +
  ggtitle("Distribution of number of price") +xlab("price") + stat_bin(bins=10)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Number of properties per zipcode

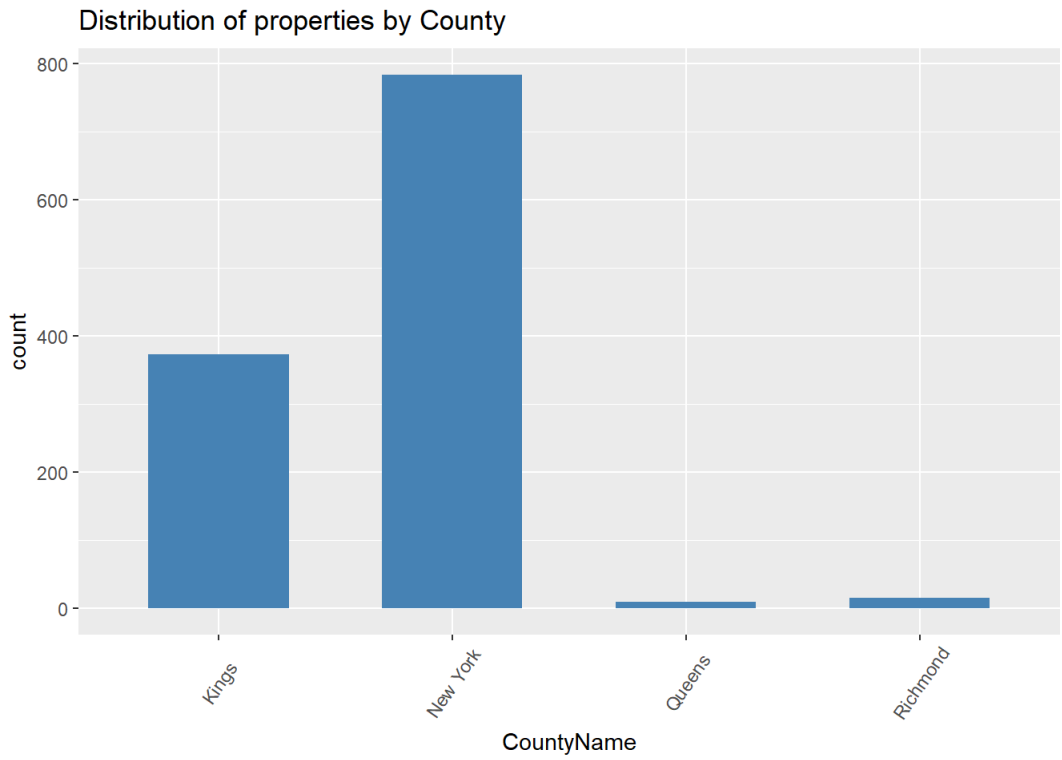
1. Zipcodes 11215,10003,10025 and 10036 have highest number of properties
2. Zipcodes 10304, 10305, 10306, 10308 and 103212 have the least number of properties

```
ggplot(finaldata, aes(RegionName,fill=RegionName))+ geom_bar(width=.6) + labs(title="Distribution of properties by zipcodes") +
  theme(axis.text.x = element_text(angle=55, vjust=0.5))
```

Number of properties per County

1. ‘New York’ county has the highest number of properties.
2. ‘Queens’ and ‘Richmond’ counties has the least number of properties.

```
g <- ggplot(finaldata, aes(CountyName))
g + geom_bar(width=.6,fill="steel blue") + labs(title="Distribution of properties by County") +
  theme(axis.text.x = element_text(angle=55, vjust=0.5))
```



Function for calculating Average Price by zipcode

1. This function groups the data by zipcode and calculates the average price for each zipcode.

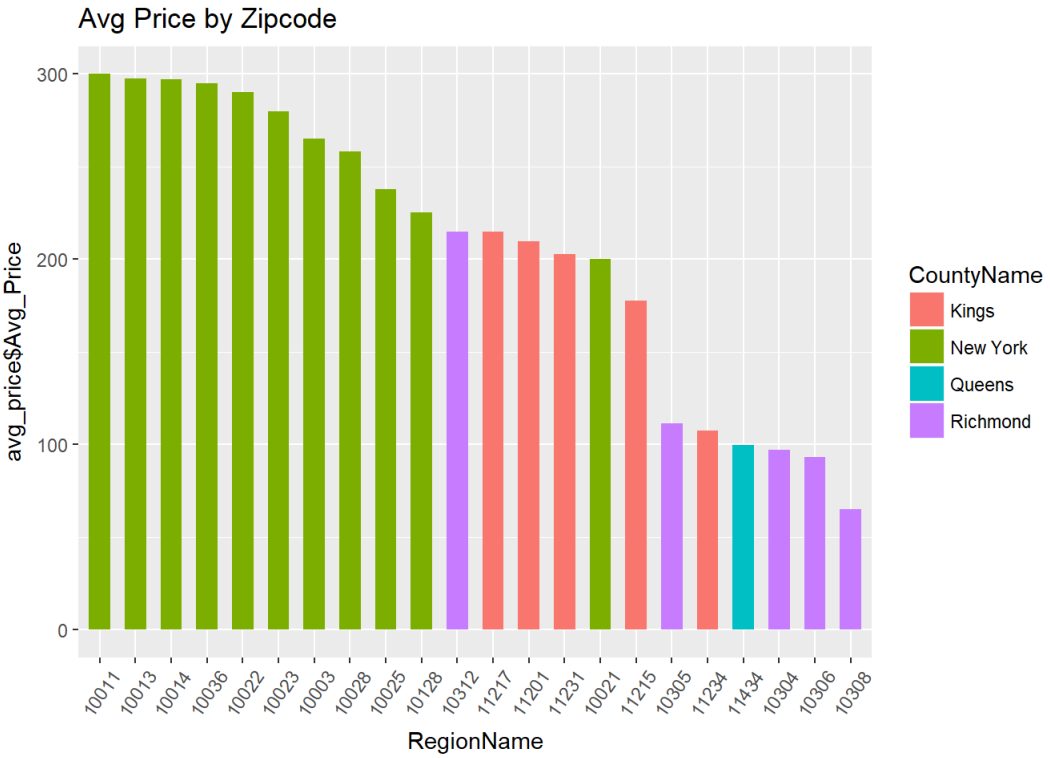
```
calc_avg_price <- function()
{
  f <- finaldata
  f <- f %>%  group_by(RegionName) %>%
    summarise(Avg_Price = mean(price))%>%
    arrange(desc(Avg_Price))
  f$RegionName <- factor(f$RegionName,levels = f$RegionName)

  return(f)
}
```

```
avg_price <- calc_avg_price()
head(avg_price)
```

```
## # A tibble: 6 x 2
##   RegionName Avg_Price
##   <fct>          <dbl>
## 1 10011            300
## 2 10013            298
## 3 10014            297
## 4 10036            295
## 5 10022            290
## 6 10023            280
```

Visualisation of Avg Price by zipcode

```
zip_county<- zillow[,c("RegionName","CountyName")]
avg_price <- merge(avg_price,zip_county,by="RegionName",all = F)

ggplot(avg_price, aes(x=RegionName, y=avg_price$Avg_Price, fill=CountyName)) +
  geom_bar(stat="identity", width=.6) +
  labs(title="Avg Price by Zipcode") +
  theme(axis.text.x = element_text(angle=55, vjust=0.5))
```



1. Almost all the zipcodes of 'New York' county have higher prices as compared to zipcodes. The prices vary from $200 - 300
2. 'Richmond' county has the lowest prices.The prices are below $100.
3. 'Kings' county has moderate price range

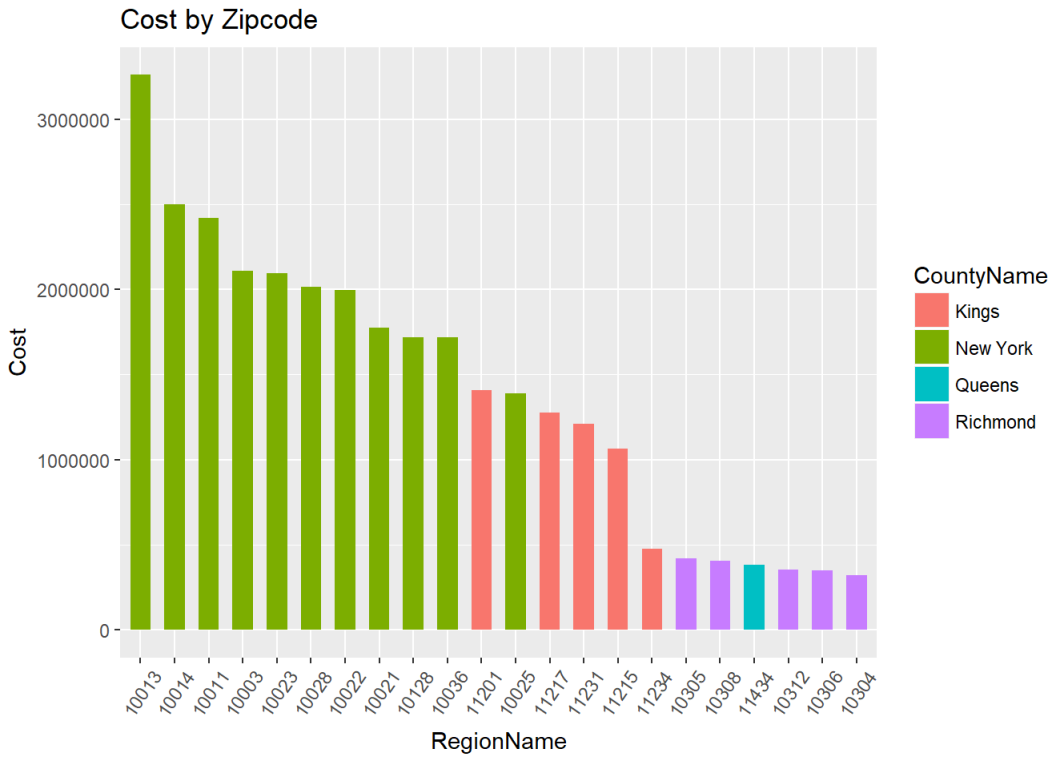
Function for calculating Average Cost by zipcode

1. This function groups the data by zipcode and calculates the average Cost for each zipcode.

```
calc_cost_by_zip <- function()
{
  cost_by_zip <- finaldata %>%
    group_by(RegionName) %>%
    summarise(Cost = mean(Cost))%>%
    arrange(desc(Cost))
  cost_by_zip$RegionName <- factor(cost_by_zip$RegionName,levels = cost_by_zip$RegionName)

  return(cost_by_zip)
}
```

Cost by Zipcode

```
cost_by_zip <- calc_cost_by_zip()
cost_by_zip <- merge(cost_by_zip,zip_county,by="RegionName",all = F)

options("scipen"=999)

ggplot(cost_by_zip, aes(x=RegionName, y=Cost, fill=CountyName)) +
  geom_bar(stat="identity", width=.6) +
  labs(title="Cost by Zipcode") +
  theme(axis.text.x = element_text(angle=55, vjust=0.5))
```



1. 'New York' county has the zipcodes with costlieat properties.
2. 'Richmond' county has the cheapest properties in New York City.

Function for calculating ROI

1. This function calculates the yearly prices and Return on Investment for each zipcodes.
2. The fomula used for calculating ROI is ROI = (yearly_price_of_the_property)*(Occupany_rate)*(100)/(Cost_of_the_property)

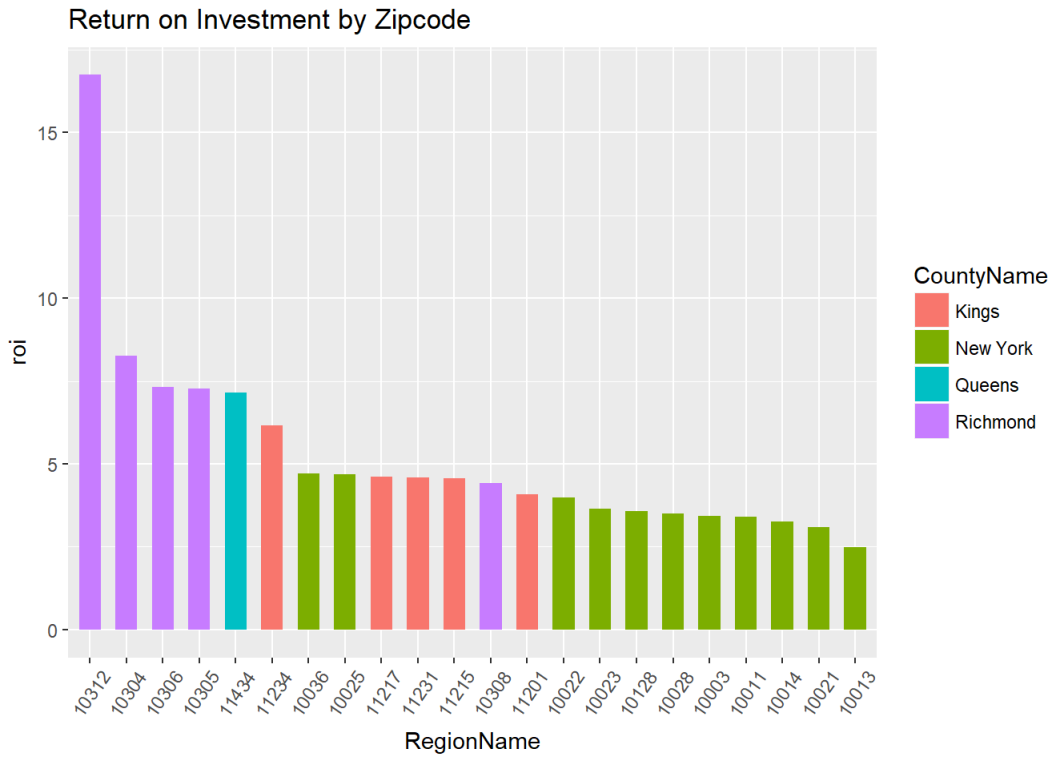
```
calc_roi <- function()
{
finaldata$yearly_price <- (finaldata$price)*365
finaldata$roi <- (finaldata$yearly_price/finaldata$Cost)*0.75*100
result <- finaldata %>%  group_by(RegionName) %>%summarise(roi = mean(roi))%>% arrange(desc(roi))
result <- as.data.frame(result)
result$RegionName <- factor(result$RegionName,levels = result$RegionName)
  return(result)
}
```

```
final <- calc_roi()
head(final)
```

```
##   RegionName       roi
## 1      10312 16.734788
## 2      10304  8.264472
## 3      10306  7.319940
## 4      10305  7.268342
## 5      11434  7.160157
## 6      11234  6.158562
```

ROI by zipcode

```
final <- merge(final,zip_county,by="RegionName",all = F)

ggplot(final, aes(x=RegionName, y=roi, fill=CountyName)) +
  geom_bar(stat="identity", width=.6) +
  labs(title="Return on Investment by Zipcode") +
  theme(axis.text.x = element_text(angle=55, vjust=0.5))
```

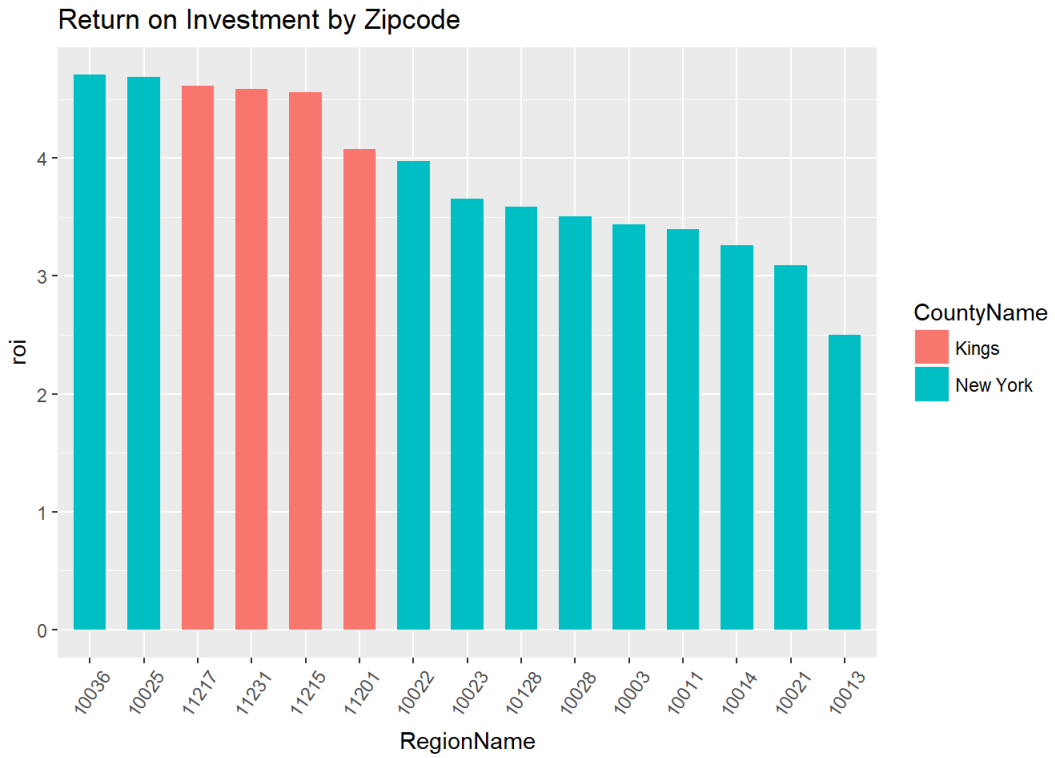


1. 'Richmond' county properties are shown to have high ROI. Zipcode 10312 has about 16% ROI.
2. 'New York' county has the least ROI.
3. 'Queens' county has ROI of about 9%.
4. 'Kings' county have ROI between 4-5%.
5. Top 5 most profitable zipcodes are 10312, 10304, 10306, 10305, 11434

However, earlier we saw that 'Richmond' and 'Queens' counties have very less number of properties. 1. This might be the reason the ROI for these 2 counties is high, as the average of prices remain high. 2. We can not confidently say that zipcodes of 'Richmond' and 'Queens' counties are the most profitable as they have very limited data.

Considering Zipcodes having atleast 10 properties

```
zip_count <- finaldata %>% group_by(RegionName) %>% summarise(count = length(RegionName))
final <- merge(final,zip_count,by="RegionName")

result <- final[final$count>=10,]
ggplot(result, aes(x=RegionName, y=roi, fill=CountyName)) +
  geom_bar(stat="identity", width=.6) +
  labs(title="Return on Investment by Zipcode") +
  theme(axis.text.x = element_text(angle=55, vjust=0.5))
```



1. The most profitable zipcodes, if we consider areas with atleast 10 properties are 10036, 10025, 11217, 11231 and 11215.

8. Summary

Insights:

1. 'Richmond' and 'Queens' have least number of properties. But, these properties have high ROI. 10312, 10304, 10306, 10305 and 11434 are the most profitable zipcodes in New York city

2. Zipcodes in 'New York' county have ROI of around 2.5-5%

3. Zipcodes in 'Kings' county have ROI of around 4-5%

4. If we exclude zipcodes having less than 10 properties, zipcodes 10036, 10025, 11217, 11231 and 11201 comes out to be top 5 profitable zipcodes in New York City.

Future Scope:

1. ARIMA can be used to predict Cost for current months using the historical time series data.

2. Regression techniques can be used to predict Prices for zipcodes for current months.

3. Predictive models can be used to predict Costs and Prices for missing zipcodes, so that we have enough data for each zipcode to perform analysis.