## 1. **Synopsis**

The Data was collected from [**Kaggle**](https://www.kaggle.com/). In this project we present to you exploratory data analysis, visualizations of New York Airbnb data. We focus on New York City’s data as we wish to perform an in-depth analysis on one of the most densely populated cities in the world.

In this project, we also try to predict the factors that affect the pricing of the airbnbs around New York. This includes creating different kind of models, model specification, transformation, variable selection and many more.

We carried out the project in the following steps:

* Data Cleaning and Preparation
* Data Visualization
* Modelling and Model Checking
* Finalising the Model
* Prediction using the Final Model.

The original Data Set can be found here —> [**New York AirBnB Data 2019**](https://www.kaggle.com/dgomonov/new-york-city-airbnb-open-data)



## 2. **Packages Required**

#install DAAG from archived source  
if(!is.element("DAAG", installed.packages()[,1])){  
 packageurl <- "https://cran.r-project.org/src/contrib/Archive/DAAG/DAAG\_1.22.tar.gz"  
 install.packages("latticeExtra")  
 install.packages(packageurl, repos=NULL, type="source")  
}  
  
  
library(tidyr)  
library(DT)  
library(ggplot2)  
library(dplyr)  
library(tidyverse)  
library(kableExtra)  
library(lubridate)  
library(readxl)  
library(highcharter)  
library(lubridate)  
library(scales)  
library(RColorBrewer)  
library(wesanderson)  
library(plotly)  
library(shiny)  
library(readxl)  
library(readr)  
library(choroplethr)  
library(choroplethrMaps)  
library(GGally)  
library(zoo)  
library(scales)  
library(ggmap)  
library(stringr)  
library(gridExtra)  
library(caret)  
library(treemap)  
library(psych)  
library(DAAG)  
library(leaps)  
library(corrplot)  
library(glmnet)

Package

Description

library(tidyr)

For changing the layout of your data sets, to convert data into the tidy format

library(DT)

For HTML display of data

library(ggplot2)

For customizable graphical representation

library(dplyr)

For data manipulation

library(tidyverse)

Collection of R packages designed for data science that works harmoniously with other packages

library(kableExtra)

To display table in a fancy way

library(lubridate)

Lubridate makes it easier to do the things R does with date-times and possible to do the things R does not

library(readxl)

The readxl package makes it easy to get data out of Excel and into R

library(highcharter)

Highcharter is a R wrapper for Highcharts javascript libray and its modules

library(scales)

The idea of the scales package is to implement scales in a way that is graphics system agnostic

library(RColorBrewer)

RColorBrewer is an R package that allows users to create colourful graphs with pre-made color palettes that visualize data in a clear and distinguishable manner

library(wesanderson)

A Wes Anderson is color palette for R

library(plotly)

Plotly’s R graphing library makes interactive, publication-quality graphs

library(shiny)

Shiny is an R package that makes it easy to build interactive web apps straight from R

## 3. **Data Preparation**

### 3.1 Loading and Reading the Data

#### Summary and Glimpse of the Data

summary(airbnb\_data)

## id name   
## Min. : 2539 Hillside Hotel : 18   
## 1st Qu.: 9471945 Home away from home : 17   
## Median :19677284 : 16   
## Mean :19017143 New york Multi-unit building : 16   
## 3rd Qu.:29152178 Brooklyn Apartment : 12   
## Max. :36487245 Loft Suite @ The Box House Hotel: 11   
## (Other) :48805   
## host\_id host\_name neighbourhood\_group  
## Min. : 2438 Michael : 417 Bronx : 1091   
## 1st Qu.: 7822033 David : 403 Brooklyn :20104   
## Median : 30793816 Sonder (NYC): 327 Manhattan :21661   
## Mean : 67620011 John : 294 Queens : 5666   
## 3rd Qu.:107434423 Alex : 279 Staten Island: 373   
## Max. :274321313 Blueground : 232   
## (Other) :46943   
## neighbourhood latitude longitude   
## Williamsburg : 3920 Min. :40.50 Min. :-74.24   
## Bedford-Stuyvesant: 3714 1st Qu.:40.69 1st Qu.:-73.98   
## Harlem : 2658 Median :40.72 Median :-73.96   
## Bushwick : 2465 Mean :40.73 Mean :-73.95   
## Upper West Side : 1971 3rd Qu.:40.76 3rd Qu.:-73.94   
## Hell's Kitchen : 1958 Max. :40.91 Max. :-73.71   
## (Other) :32209   
## room\_type price minimum\_nights   
## Entire home/apt:25409 Min. : 0.0 Min. : 1.00   
## Private room :22326 1st Qu.: 69.0 1st Qu.: 1.00   
## Shared room : 1160 Median : 106.0 Median : 3.00   
## Mean : 152.7 Mean : 7.03   
## 3rd Qu.: 175.0 3rd Qu.: 5.00   
## Max. :10000.0 Max. :1250.00   
##   
## number\_of\_reviews last\_review reviews\_per\_month  
## Min. : 0.00 :10052 Min. : 0.010   
## 1st Qu.: 1.00 2019-06-23: 1413 1st Qu.: 0.190   
## Median : 5.00 2019-07-01: 1359 Median : 0.720   
## Mean : 23.27 2019-06-30: 1341 Mean : 1.373   
## 3rd Qu.: 24.00 2019-06-24: 875 3rd Qu.: 2.020   
## Max. :629.00 2019-07-07: 718 Max. :58.500   
## (Other) :33137 NA's :10052   
## calculated\_host\_listings\_count availability\_365  
## Min. : 1.000 Min. : 0.0   
## 1st Qu.: 1.000 1st Qu.: 0.0   
## Median : 1.000 Median : 45.0   
## Mean : 7.144 Mean :112.8   
## 3rd Qu.: 2.000 3rd Qu.:227.0   
## Max. :327.000 Max. :365.0   
##

glimpse(airbnb\_data)

## Observations: 48,895  
## Variables: 16  
## $ id <int> 2539, 2595, 3647, 3831, 5022, 5...  
## $ name <fct> "Clean & quiet apt home by the ...  
## $ host\_id <int> 2787, 2845, 4632, 4869, 7192, 7...  
## $ host\_name <fct> John, Jennifer, Elisabeth, Lisa...  
## $ neighbourhood\_group <fct> Brooklyn, Manhattan, Manhattan,...  
## $ neighbourhood <fct> Kensington, Midtown, Harlem, Cl...  
## $ latitude <dbl> 40.64749, 40.75362, 40.80902, 4...  
## $ longitude <dbl> -73.97237, -73.98377, -73.94190...  
## $ room\_type <fct> Private room, Entire home/apt, ...  
## $ price <int> 149, 225, 150, 89, 80, 200, 60,...  
## $ minimum\_nights <int> 1, 1, 3, 1, 10, 3, 45, 2, 2, 1,...  
## $ number\_of\_reviews <int> 9, 45, 0, 270, 9, 74, 49, 430, ...  
## $ last\_review <fct> 2018-10-19, 2019-05-21, , 2019-...  
## $ reviews\_per\_month <dbl> 0.21, 0.38, NA, 4.64, 0.10, 0.5...  
## $ calculated\_host\_listings\_count <int> 6, 2, 1, 1, 1, 1, 1, 1, 1, 4, 1...  
## $ availability\_365 <int> 365, 355, 365, 194, 0, 129, 0, ...

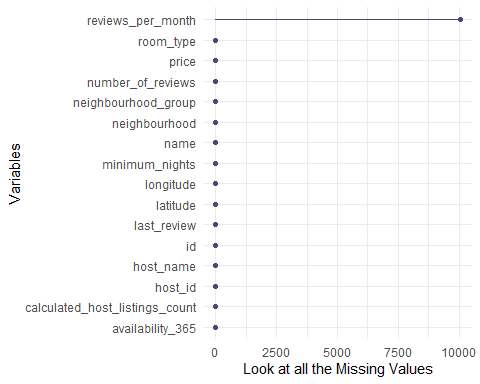
#### Checking for NA

summary(is.na(airbnb\_data))

## id name host\_id host\_name   
## Mode :logical Mode :logical Mode :logical Mode :logical   
## FALSE:48895 FALSE:48895 FALSE:48895 FALSE:48895   
##   
## neighbourhood\_group neighbourhood latitude longitude   
## Mode :logical Mode :logical Mode :logical Mode :logical   
## FALSE:48895 FALSE:48895 FALSE:48895 FALSE:48895   
##   
## room\_type price minimum\_nights number\_of\_reviews  
## Mode :logical Mode :logical Mode :logical Mode :logical   
## FALSE:48895 FALSE:48895 FALSE:48895 FALSE:48895   
##   
## last\_review reviews\_per\_month calculated\_host\_listings\_count  
## Mode :logical Mode :logical Mode :logical   
## FALSE:48895 FALSE:38843 FALSE:48895   
## TRUE :10052   
## availability\_365  
## Mode :logical   
## FALSE:48895   
##

We visualise the number of missings in each variable using naniar gg\_miss\_var

naniar::gg\_miss\_var(airbnb\_data) +  
 theme\_minimal()+  
 labs(y = "Look at all the Missing Values")



### 3.2 Data Cleaning

### 3.3 Cleaned Dataset

The final cleaned dataset can be found below in an interactive table.

datatable(head(airbnb\_data, 20), class = 'cell-border stripe')

### 3.4 Summary of Variables

#Reading the variable summary excel File  
var\_sum <- read\_excel("variable\_summary.xlsx")  
  
kable(var\_sum) %>%  
 kable\_styling(bootstrap\_options = c("striped", "hover", "condensed", "responsive"), full\_width = F, fixed\_thead = T, )

Variable

Class

Description

anime\_id

int

anime ID (as in <https://myanimelist.net/anime/animeID>)

anime\_name

character

anime title - extracted from the site.

anime\_type

factor

anime type (e.g. TV, Movie, OVA)

source

character

source of anime (i.e original, manga, game, music, visual novel etc.)

producers

character

producers

genre

factor

genre

studio

character

studio

no\_of\_episodes

int

number of episodes

airing\_status

character

True/False- is still airing?

start\_date

date

start date

episode\_duration

character

per episode duration or entire duration, text string

MPAA\_rating

factor

age rating

viwers\_rating

num

score (higher = better)

rated\_by\_no\_of\_viewers

int

number of users that scored

rankings

int

rank - weight according to MyAnimeList formula

popularity\_index

int

based on how many members/users have the respective anime in their list

wishlisted\_members

int

number members that added this anime in their list

favorites

int

number members that favorites these in their list

premiered\_season

factor

the season the show first premiered

premiered\_year

num

the year the show first premiered

broadcast\_day

factor

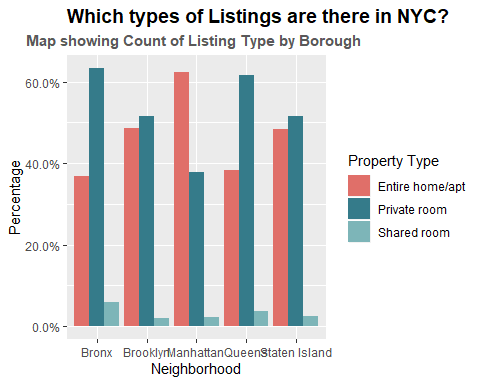
the day of the week it gets broadcasted

broadcast\_time

character

the time of the day it gets broadcasted

## 4. **Exploratory Data Analysis**



## 5. **Modelling**

### 5.1 Data Splitting

Training set will be 70% percent of the original data. Objects with price equal to 0 will be ommited since price can’t be 0 (faulty records). They would make predictive models significantly weaker.

airbnb\_data <- airbnb\_data %>% mutate(id = row\_number())  
  
airbnb\_train <- airbnb\_data %>% sample\_frac(.7) %>% filter(price > 0)  
  
airbnb\_test <- anti\_join(airbnb\_data, airbnb\_train, by = 'id') %>% filter(price > 0)

# sanity check  
nrow(airbnb\_train) + nrow(airbnb\_test) == nrow(airbnb\_data %>% filter(price > 0))

## [1] TRUE

### 5.2 Model Building Process

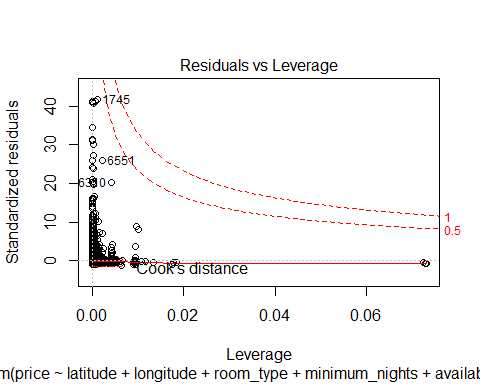
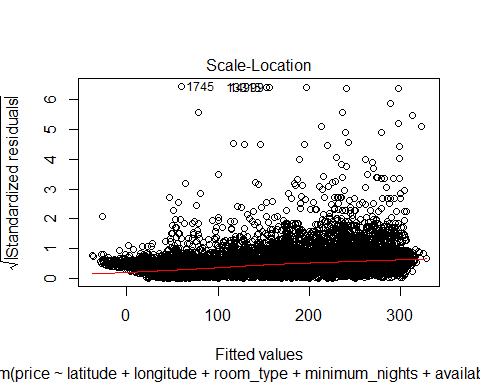
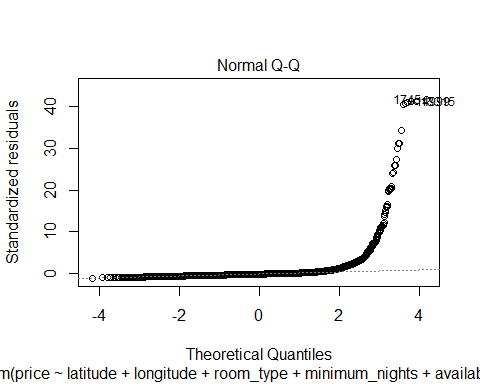
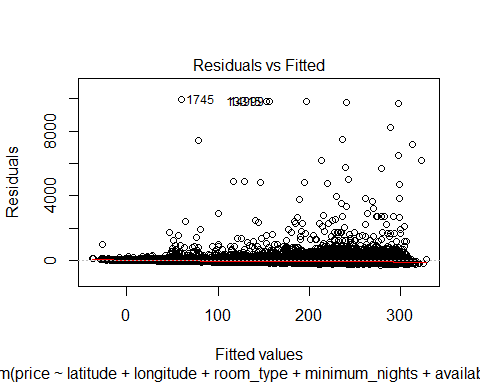
#### 5.2.1 1st Linear Regression Model

airbnb\_model\_1 <- lm (price ~ latitude + longitude + room\_type + minimum\_nights + availability\_365 + neighbourhood\_group, data = airbnb\_train)  
  
summary(airbnb\_model\_1)

##   
## Call:  
## lm(formula = price ~ latitude + longitude + room\_type + minimum\_nights +   
## availability\_365 + neighbourhood\_group, data = airbnb\_train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -265.4 -62.7 -24.7 15.0 9939.8   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) -2.851e+04 4.003e+03 -7.122 1.08e-12  
## latitude -2.182e+02 3.909e+01 -5.581 2.41e-08  
## longitude -5.083e+02 4.487e+01 -11.329 < 2e-16  
## room\_typePrivate room -1.057e+02 2.700e+00 -39.160 < 2e-16  
## room\_typeShared room -1.342e+02 8.622e+00 -15.561 < 2e-16  
## minimum\_nights 7.554e-02 6.483e-02 1.165 0.243944  
## availability\_365 1.608e-01 1.014e-02 15.863 < 2e-16  
## neighbourhood\_groupBrooklyn -3.978e+01 1.096e+01 -3.630 0.000284  
## neighbourhood\_groupManhattan 2.336e+01 9.949e+00 2.348 0.018855  
## neighbourhood\_groupQueens -8.954e+00 1.060e+01 -0.845 0.398136  
## neighbourhood\_groupStaten Island -1.476e+02 2.108e+01 -7.004 2.54e-12  
##   
## (Intercept) \*\*\*  
## latitude \*\*\*  
## longitude \*\*\*  
## room\_typePrivate room \*\*\*  
## room\_typeShared room \*\*\*  
## minimum\_nights   
## availability\_365 \*\*\*  
## neighbourhood\_groupBrooklyn \*\*\*  
## neighbourhood\_groupManhattan \*   
## neighbourhood\_groupQueens   
## neighbourhood\_groupStaten Island \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 238.8 on 34207 degrees of freedom  
## Multiple R-squared: 0.08855, Adjusted R-squared: 0.08829   
## F-statistic: 332.3 on 10 and 34207 DF, p-value: < 2.2e-16

#### Plot of the 1st Linear Regression Model

plot(airbnb\_model\_1)



#### 5.2.2 **Transformation** - 2nd Linear Regression Model -

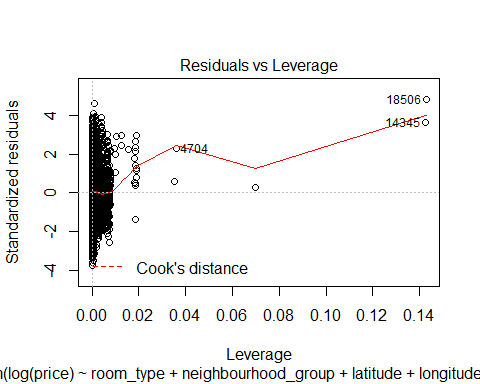
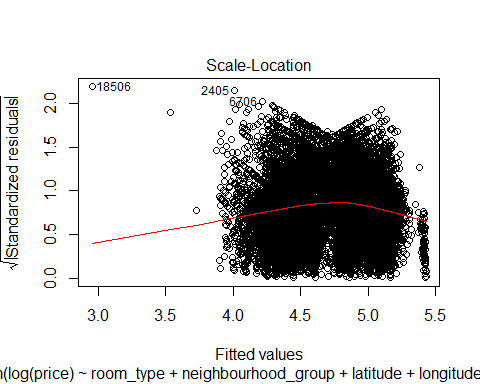
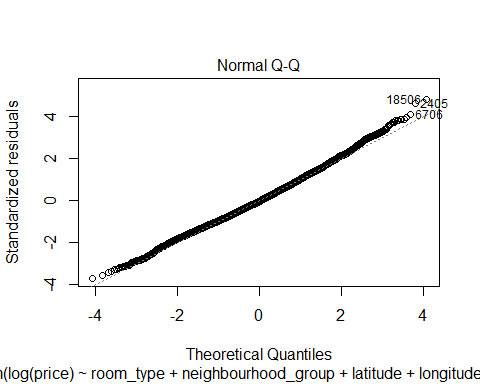
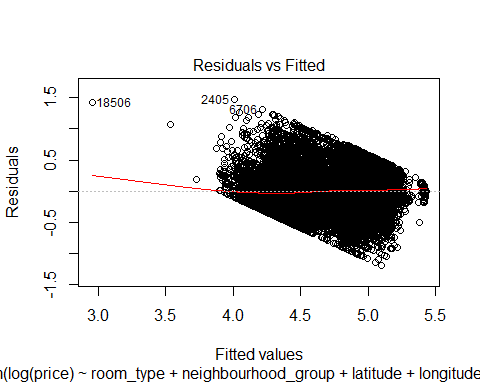
airbnb\_data\_2 <- airbnb\_train %>%   
 filter(price < quantile(airbnb\_train$price, 0.9) & price > quantile(airbnb\_train$price, 0.1)) %>%   
 drop\_na()

airbnb\_model\_2 <- lm(log(price) ~ room\_type + neighbourhood\_group + latitude + longitude + number\_of\_reviews + availability\_365   
 + reviews\_per\_month + calculated\_host\_listings\_count + minimum\_nights, data = airbnb\_data\_2)  
  
# Summarize the results  
summary(airbnb\_model\_2)

##   
## Call:  
## lm(formula = log(price) ~ room\_type + neighbourhood\_group + latitude +   
## longitude + number\_of\_reviews + availability\_365 + reviews\_per\_month +   
## calculated\_host\_listings\_count + minimum\_nights, data = airbnb\_data\_2)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.18715 -0.22352 -0.01742 0.20643 1.46944   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) -1.306e+02 6.656e+00 -19.623 < 2e-16  
## room\_typePrivate room -5.404e-01 4.410e-03 -122.536 < 2e-16  
## room\_typeShared room -6.253e-01 2.034e-02 -30.742 < 2e-16  
## neighbourhood\_groupBrooklyn -4.522e-02 1.942e-02 -2.329 0.0199  
## neighbourhood\_groupManhattan 1.495e-01 1.783e-02 8.384 < 2e-16  
## neighbourhood\_groupQueens 3.709e-02 1.885e-02 1.968 0.0491  
## neighbourhood\_groupStaten Island -6.033e-01 3.591e-02 -16.802 < 2e-16  
## latitude -5.728e-01 6.495e-02 -8.820 < 2e-16  
## longitude -2.147e+00 7.481e-02 -28.705 < 2e-16  
## number\_of\_reviews -1.106e-04 5.260e-05 -2.103 0.0355  
## availability\_365 3.491e-04 1.809e-05 19.294 < 2e-16  
## reviews\_per\_month -1.714e-03 1.531e-03 -1.120 0.2629  
## calculated\_host\_listings\_count 4.694e-04 8.835e-05 5.313 1.09e-07  
## minimum\_nights -1.416e-03 1.212e-04 -11.686 < 2e-16  
##   
## (Intercept) \*\*\*  
## room\_typePrivate room \*\*\*  
## room\_typeShared room \*\*\*  
## neighbourhood\_groupBrooklyn \*   
## neighbourhood\_groupManhattan \*\*\*  
## neighbourhood\_groupQueens \*   
## neighbourhood\_groupStaten Island \*\*\*  
## latitude \*\*\*  
## longitude \*\*\*  
## number\_of\_reviews \*   
## availability\_365 \*\*\*  
## reviews\_per\_month   
## calculated\_host\_listings\_count \*\*\*  
## minimum\_nights \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.3183 on 22140 degrees of freedom  
## Multiple R-squared: 0.497, Adjusted R-squared: 0.4967   
## F-statistic: 1683 on 13 and 22140 DF, p-value: < 2.2e-16

#### Plot of the Transformed Linear Regression Model

plot(airbnb\_model\_2)

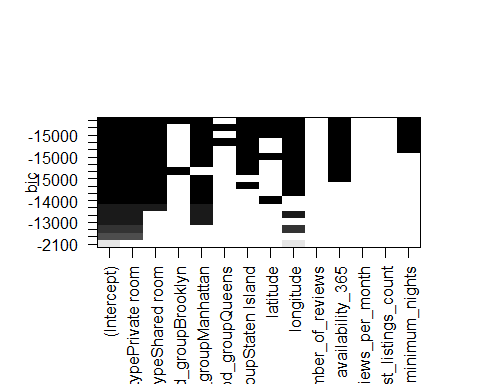


#### 5.2.3 Model Building by **Variable Selection** Method

#### **Best Subset Regression Method**

best\_fit\_model <- regsubsets (log(price) ~ room\_type + neighbourhood\_group + latitude + longitude + number\_of\_reviews + availability\_365   
 + reviews\_per\_month + calculated\_host\_listings\_count + minimum\_nights, data = airbnb\_data\_2, nbest = 2, nvmax = 9)  
  
summary(best\_fit\_model)

plot(best\_fit\_model, scale="bic")



#### Model Building by **Stepwise Regression with AIC/BIC** (direction = forward/backward/both)

#### **stepwise selection using AIC** (Direction = “both”)

null <- lm(log(price)~1, data = airbnb\_train)  
full <- lm(log(price) ~ room\_type + neighbourhood\_group + latitude + longitude + number\_of\_reviews + availability\_365   
 + reviews\_per\_month + calculated\_host\_listings\_count + minimum\_nights, data = airbnb\_train)  
  
step(null, scope =list(lower=null, upper= full), direction = "both")

## Start: AIC=-24719.36  
## log(price) ~ 1

## Warning in add1.lm(fit, scope$add, scale = scale, trace = trace, k = k, :  
## using the 27284/34218 rows from a combined fit

## Df Sum of Sq RSS AIC  
## + room\_type 2 4883.6 7125.5 -36626  
## + neighbourhood\_group 4 1529.7 10479.4 -26098  
## + longitude 1 1359.4 10649.7 -25664  
## + calculated\_host\_listings\_count 1 127.4 11881.7 -22677  
## + availability\_365 1 81.2 11927.9 -22571  
## + latitude 1 43.2 11965.9 -22485  
## + reviews\_per\_month 1 18.2 11990.9 -22428  
## + minimum\_nights 1 6.0 12003.1 -22400  
## + number\_of\_reviews 1 5.6 12003.5 -22399  
## <none> 12009.1 -22388  
##   
## Step: AIC=-41338.6  
## log(price) ~ room\_type

## Warning in add1.lm(fit, scope$add, scale = scale, trace = trace, k = k, :  
## using the 27284/34218 rows from a combined fit

## Df Sum of Sq RSS AIC  
## <none> 10221.4 -41339  
## + neighbourhood\_group 4 845.1 6280.3 -40063  
## + longitude 1 590.3 6535.1 -38984  
## + availability\_365 1 123.6 7001.9 -37101  
## + latitude 1 79.0 7046.5 -36928  
## + calculated\_host\_listings\_count 1 37.9 7087.6 -36770  
## + minimum\_nights 1 7.2 7118.3 -36652  
## + number\_of\_reviews 1 3.6 7121.8 -36638  
## + reviews\_per\_month 1 2.5 7123.0 -36634  
## - room\_type 2 6393.3 16614.7 -24719

##   
## Call:  
## lm(formula = log(price) ~ room\_type, data = airbnb\_train)  
##   
## Coefficients:  
## (Intercept) room\_typePrivate room room\_typeShared room   
## 5.1409 -0.8441 -1.1643

#### **stepwise selection using BIC** (Direction = “both”)

null <- lm(log(price)~1, data = airbnb\_train)  
full <- lm(log(price) ~ room\_type + neighbourhood\_group + latitude + longitude + number\_of\_reviews + availability\_365   
 + reviews\_per\_month + calculated\_host\_listings\_count + minimum\_nights, data = airbnb\_train)  
  
n=dim(airbnb\_train[1])  
step(null, scope =list(lower=null, upper= full), k=log(n), direction = "both")

## Start: AIC=-24710.92  
## log(price) ~ 1

## Warning in add1.lm(fit, scope$add, scale = scale, trace = trace, k = k, :  
## using the 27284/34218 rows from a combined fit

## Df Sum of Sq RSS AIC  
## + room\_type 2 4883.6 7125.5 -36632  
## + neighbourhood\_group 4 1529.7 10479.4 -26056  
## + longitude 1 1359.4 10649.7 -25647  
## + calculated\_host\_listings\_count 1 127.4 11881.7 -22660  
## + availability\_365 1 81.2 11927.9 -22554  
## + latitude 1 43.2 11965.9 -22489  
## + reviews\_per\_month 1 18.2 11990.9 -22432  
## + minimum\_nights 1 6.0 12003.1 -22404  
## + number\_of\_reviews 1 5.6 12003.5 -22403  
## <none> 12009.1 -22380  
##   
## Step: AIC=-41313.27  
## log(price) ~ room\_type

## Warning in add1.lm(fit, scope$add, scale = scale, trace = trace, k = k, :  
## using the 27284/34218 rows from a combined fit

## Warning in k \* dfs: longer object length is not a multiple of shorter  
## object length

## Df Sum of Sq RSS AIC  
## <none> 10221.4 -41313  
## + neighbourhood\_group 4 845.1 6280.3 -40077  
## + longitude 1 590.3 6535.1 -38992  
## + availability\_365 1 123.6 7001.9 -37109  
## + latitude 1 79.0 7046.5 -36895  
## + calculated\_host\_listings\_count 1 37.9 7087.6 -36778  
## + minimum\_nights 1 7.2 7118.3 -36618  
## + number\_of\_reviews 1 3.6 7121.8 -36604  
## + reviews\_per\_month 1 2.5 7123.0 -36600  
## - room\_type 2 6393.3 16614.7 -24721

##   
## Call:  
## lm(formula = log(price) ~ room\_type, data = airbnb\_train)  
##   
## Coefficients:  
## (Intercept) room\_typePrivate room room\_typeShared room   
## 5.1409 -0.8441 -1.1643

## 6. **Conclusion**