Multiple Regression_AirbnbIstanbul_Dataset

By_Kalyani Pandit

Loading Libraries

```
library(data.table)
library(fpp)

## Loading required package: forecast

## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo

## Loading required package: fma

## Loading required package: expsmooth

## Loading required package: lmtest

## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

## Loading required package: tseries

library(fpp2)

## Loading required package: ggplot2

##
## Attaching package: 'fpp2'

## The following objects are masked from 'package:fpp':
##
##     ausair, ausbeer, austa, austourists, debitcards, departures,
##     elecequip, euretail, guinearice, oil, sunspotarea, usmelec

library(cowplot)

##
## ********************************************************
## Note: As of version 1.0.0, cowplot does not change the

##   default ggplot2 theme anymore. To recover the previous

##   behavior, execute:
##   theme_set(theme_cowplot())

## ********************************************************

library(tidyverse)

## -- Attaching packages ---------------------------- tidyverse 1.3.0 --

## v tibble  2.1.3     v dplyr   0.8.4
## v tidyr   1.0.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0
## v purrr   0.3.3

## -- Conflicts ------------------------------- tidyverse_conflicts() --
## x dplyr::between()   masks data.table::between()
## x dplyr::filter()    masks stats::filter()
## x dplyr::first()     masks data.table::first()
## x dplyr::lag()       masks stats::lag()
## x dplyr::last()      masks data.table::last()
## x purrr::transpose() masks data.table::transpose()

library(psych)

##
## Attaching package: 'psych'

## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha

library(e1071)
library(dplyr)
library(corrplot)

## corrplot 0.84 loaded

library(GGally)

## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2

##
## Attaching package: 'GGally'

## The following object is masked from 'package:dplyr':
##
##     nasa
## The following object is masked from 'package:fma':
## 
##     pigs
```

```
library(reshape2)
```

```
## 
## Attaching package: 'reshape2'
```

```
## The following object is masked from 'package:tidyr':
## 
##     smiths
```

```
## The following objects are masked from 'package:data.table':
## 
##     dcast, melt
```

```
#AirbnbIstanbul <- read.csv("C:/Pritesh/Rutgers/Courses/Projects/MVA/Dataset/
AirbnbIstanbul.csv", stringsAsFactors=FALSE)
#AirbnbIstanbul<-read.csv("C:/Alok/OneDrive/Rutgers_MITA/Semester2/MVA/R/Airb
nbIstanbul.csv",stringsAsFactors = FALSE)
AirbnbIstanbul <- read.csv("C:/Users/prach/Desktop/MVA/Copy_of_AirbnbIstanbul
.csv", stringsAsFactors = FALSE)

Istanbul <- copy(AirbnbIstanbul)
class(Istanbul)
```

```
## [1] "data.frame"
```

```
setDT(Istanbul)
```

# data exploration and cleansing

```
str(Istanbul) ## to check data type of each var.
```

```
## Classes 'data.table' and 'data.frame':   16251 obs. of  16 variables:
##  $ id                            : int  4826 20815 25436 27271 28277 28308 
28318 29241 30697 33368 ...
##  $ name                          : chr  "The Place" "The Bosphorus from Th
e Comfy Hill" "House for vacation rental furnutare" "LOVELY APT. IN PERFECT L
OCATION" ...
##  $ host_id                       : int  6603 78838 105823 117026 121607 12
1695 121721 125742 132137 135136 ...
##  $ host_name                     : chr  "Kaan" "GÃ¼lder" "Yesim" "Mutlu" .
..
##  $ neighbourhood_group           : logi  NA NA NA NA NA NA ...
##  $ neighbourhood                 : chr  "Uskudar" "Besiktas" "Besiktas" "B
eyoglu" ...
##  $ latitude                      : num  41.1 41.1 41.1 41 41 ...
##  $ longitude                     : num  29.1 29 29 29 29 ...
##  $ room_type                     : chr  "Entire home/apt" "Entire home/apt
" "Entire home/apt" "Entire home/apt" ...
##  $ price                         : int  554 100 211 237 591 237 633 264 59
6 295 ...
##  $ minimum_nights                : int  1 30 21 5 3 1 3 3 1 2 ...
##  $ number_of_reviews             : int  1 41 0 2 0 0 0 0 1 1 ...
##  $ last_review                   : chr  "6/1/2009" "11/7/2018" "" "5/4/201
8" ...
##  $ reviews_per_month             : num  0.01 0.38 NA 0.04 NA NA NA NA 0.01
0.02 ...
##  $ calculated_host_listings_count: int  1 2 1 1 13 1 1 1 1 2 ...
##  $ availability_365              : int  365 49 83 228 356 365 365 365 365
232 ...
##  - attr(*, ".internal.selfref")=<externalptr>
```

```
grep('NA',Istanbul) ## indicates NA values are there in 2nd, 5th and 14th col
umn
```

```
## [1]  2  5 14
```

```
# i.e. name, neighbourhood_group and reviews_per_month have NA values
head(Istanbul,10)
```

```
##        id                                name host_id host_name
##  1:  4826                           The Place    6603      Kaan
##  2: 20815   The Bosphorus from The Comfy Hill   78838   GÃ¼lder
##  3: 25436 House for vacation rental furnutare  105823     Yesim
##  4: 27271      LOVELY APT. IN PERFECT LOCATION  117026     Mutlu
##  5: 28277      Duplex Apartment with Terrace  121607      Alen
##  6: 28308      Great apartment in Cihangir...  121695   Mustafa
##  7: 28318      Cosy home overlooking Bosphorus  121721     Aydin
##  8: 29241    âtª Istanbul, Your second house  125742    Åževki
##  9: 30697            nice home in popular area  132137       Nan
## 10: 33368   Deluxe double bedroom @ Nisantasi  135136     Ozlem
##     neighbourhood_group neighbourhood latitude longitude       room_type p
rice
##  1:                  NA       Uskudar 41.05650  29.05367 Entire home/apt
554
##  2:                  NA      Besiktas 41.06984  29.04545 Entire home/apt
100
##  3:                  NA      Besiktas 41.07731  29.03891 Entire home/apt
211
##  4:                  NA       Beyoglu 41.03220  28.98216 Entire home/apt
237
##  5:                  NA         Sisli 41.04471  28.98567 Entire home/apt
591
##  6:                  NA       Beyoglu 41.03105  28.98297 Entire home/apt
237
##  7:                  NA       Sariyer 41.09048  29.05559 Entire home/apt
633
```

```
##  8:                  NA       Beyoglu 41.04844  28.95254    Private room
264
##  9:                  NA       Beyoglu 41.03350  28.97626    Private room
596
## 10:                  NA         Sisli 41.05382  28.99739    Private room
295
##     minimum_nights number_of_reviews last_review reviews_per_month
##  1:              1                 1    6/1/2009              0.01
##  2:             30                41   11/7/2018              0.38
##  3:             21                 0                            NA
##  4:              5                 2    5/4/2018              0.04
##  5:              3                 0                            NA
##  6:              1                 0                            NA
##  7:              3                 0                            NA
##  8:              3                 0                            NA
##  9:              1                 1   6/14/2010              0.01
## 10:              2                 1  10/21/2014              0.02
##     calculated_host_listings_count availability_365
##  1:                              1              365
##  2:                              2               49
##  3:                              1               83
##  4:                              1              228
##  5:                             13              356
##  6:                              1              365
##  7:                              1              365
##  8:                              1              365
##  9:                              1              365
## 10:                              2              232
```

```r
dim(Istanbul) # 16251 obs. and 16 vars
```

```
## [1] 16251    16
```

```r
summary(Istanbul) ## summarized view of all the feature/vars
```

```
##        id                 name                host_id             host_name
##  Min.   :    4826   Length:16251       Min.   :     6603   Length:16251
##  1st Qu.: 8500978   Class :character   1st Qu.: 17882300   Class :characte
r
##  Median :21619750   Mode  :character   Median : 52107399   Mode  :characte
r
##  Mean   :18856396                      Mean   : 88887056
##  3rd Qu.:28702192                      3rd Qu.:168134520
##  Max.   :32457561                      Max.   :243734065
##
##  neighbourhood_group neighbourhood        latitude       longitude
##  Mode:logical        Length:16251       Min.   :40.81   Min.   :28.03
##  NA's:16251          Class :character   1st Qu.:41.00   1st Qu.:28.97
##                      Mode  :character   Median :41.03   Median :28.98
##                                         Mean   :41.03   Mean   :28.99
##                                         3rd Qu.:41.05   3rd Qu.:29.02
##                                         Max.   :41.41   Max.   :29.91  
##                                                                        
##    room_type             price          minimum_nights    number_of_reviews
##   Length:16251      Min.   :    0.0   Min.   :   1.000   Min.   :  0.000  
##   Class :character  1st Qu.:  105.0   1st Qu.:   1.000   1st Qu.:  0.000  
##   Mode  :character  Median :  190.0   Median :   1.000   Median :  0.000  
##                     Mean   :  354.7   Mean   :   4.693   Mean   :  7.187  
##                     3rd Qu.:  327.0   3rd Qu.:   2.000   3rd Qu.:  4.000  
##                     Max.   :59561.0   Max.   :1125.000   Max.   :307.000  
##                                                                          
##   last_review        reviews_per_month calculated_host_listings_count
##   Length:16251       Min.   : 0.010    Min.   : 1.000                
##   Class :character   1st Qu.: 0.180    1st Qu.: 1.000                
##   Mode  :character   Median : 0.520    Median : 1.000                
##                      Mean   : 0.915    Mean   : 4.104                
##                      3rd Qu.: 1.190    3rd Qu.: 4.000                
##                      Max.   :12.000    Max.   :77.000                
##                      NA's   :8484                                    
##   availability_365
##   Min.   :  0.0   
##   1st Qu.:101.0   
##   Median :340.0   
##   Mean   :249.5   
##   3rd Qu.:365.0   
##   Max.   :365.0   
## 
```

```
unique(Istanbul$room_type) ## 3 unique room types
```

```
## [1] "Entire home/apt" "Private room"    "Shared room"
```

```
unique(Istanbul$neighbourhood) ## 39 unique neighbourhoods
```

```
##  [1] "Uskudar"       "Besiktas"      "Beyoglu"       "Sisli"        
##  [5] "Sariyer"       "Beykoz"        "Atasehir"      "Fatih"        
##  [9] "Adalar"        "Kadikoy"       "Kagithane"     "Maltepe"      
## [13] "Bakirkoy"      "Esenyurt"      "Basaksehir"    "Kartal"       
## [17] "Gaziosmanpasa" "Bahcelievler"  "Bagcilar"      "Buyukcekmece" 
## [21] "Silivri"       "Beylikduzu"    "Umraniye"      "Sile"         
## [25] "Cekmekoy"      "Sancaktepe"    "Tuzla"         "Pendik"       
## [29] "Sultangazi"    "Eyup"          "Zeytinburnu"   "Kucukcekmece" 
## [33] "Avcilar"       "Gungoren"      "Catalca"       "Bayrampasa"   
## [37] "Esenler"       "Sultanbeyli"   "Arnavutkoy"
```

```
## since, I used stringsAsFactors=FALSE while importing the dataset, few of t
he columns
## like name, host_name, neighbourhood and room_type belongs to character dat
a type
## hence, will factor neighbourhood and room_type for now. name and host_name
doesn't seem
```

```
## to be much interest for now, hence will leave those.
str(Istanbul)
```

```
## Classes 'data.table' and 'data.frame':   16251 obs. of  16 variables:
##  $ id                            : int  4826 20815 25436 27271 28277 28308
28318 29241 30697 33368 ...
##  $ name                          : chr  "The Place" "The Bosphorus from Th
e Comfy Hill" "House for vacation rental furnutare" "LOVELY APT. IN PERFECT L
OCATION" ...
##  $ host_id                       : int  6603 78838 105823 117026 121607 12
1695 121721 125742 132137 135136 ...
##  $ host_name                     : chr  "Kaan" "GÃ¼lder" "Yesim" "Mutlu" .
..
##  $ neighbourhood_group           : logi  NA NA NA NA NA NA ...
##  $ neighbourhood                 : chr  "Uskudar" "Besiktas" "Besiktas" "B
eyoglu" ...
##  $ latitude                      : num  41.1 41.1 41.1 41 41 ...
##  $ longitude                     : num  29.1 29 29 29 29 ...
##  $ room_type                     : chr  "Entire home/apt" "Entire home/apt
" "Entire home/apt" "Entire home/apt" ...
##  $ price                         : int  554 100 211 237 591 237 633 264 59
6 295 ...
##  $ minimum_nights                : int  1 30 21 5 3 1 3 3 1 2 ...
##  $ number_of_reviews             : int  1 41 0 2 0 0 0 0 1 1 ...
##  $ last_review                   : chr  "6/1/2009" "11/7/2018" "" "5/4/201
8" ...
##  $ reviews_per_month             : num  0.01 0.38 NA 0.04 NA NA NA NA 0.01
0.02 ...
##  $ calculated_host_listings_count: int  1 2 1 1 13 1 1 1 1 2 ...
##  $ availability_365              : int  365 49 83 228 356 365 365 365 365
232 ...
##  - attr(*, ".internal.selfref")=<externalptr>
```

```
Istanbul[,room_type:=factor(room_type)]
Istanbul[,neighbourhood:=factor(neighbourhood)]
Istanbul[,last_review:=as.Date(last_review,'%Y-%m-%d')] ## converting last_re
view to date datatype

# datatypes looks better now. hence will see again for NA values
grep ('NA',Istanbul) # 2, 5, 13 and 14 column have NA values
```

```
## [1]  2  5 13 14
```

```
Istanbul[is.na(neighbourhood_group),NROW(neighbourhood_group)] # entire obs.
is blank, will drop this var
```

```
## [1] 16251
```

```
Istanbul[is.na(last_review),NROW(last_review)] ## there are 8484 NA values
```

```
## [1] 16251
```

```
Istanbul[is.na(reviews_per_month),NROW(reviews_per_month)] ## there are 8484
NA values
```

```
## [1] 8484
```

```
Istanbul$neighbourhood_group <- NULL ## removing neighbourhood_group column
Istanbul[is.na(reviews_per_month),reviews_per_month:=0] ## nearly 50% of the
dataset is filled with NA.
# hence we can't simply remove these many rows. Hence imputing with 0 values.
```

# removing outliers

```
Istanbul.1 <- Istanbul[price < 1000]
summary(Istanbul.1)
```

```
##        id               name              host_id           host_name
##  Min.   :    4826   Length:15638       Min.   :     6603   Length:15638
##  1st Qu.: 8454342   Class :character   1st Qu.: 17953407   Class :characte
r
##  Median :21640688   Mode  :character   Median : 51932114   Mode  :characte
r
##  Mean   :18853866                      Mean   : 88865840
##  3rd Qu.:28713844                      3rd Qu.:167997481
##  Max.   :32457561                      Max.   :243734065
##
##   neighbourhood     latitude       longitude               room_type
##  Beyoglu :4101   Min.   :40.81   Min.   :28.03   Entire home/apt:6794
##  Sisli   :2255   1st Qu.:41.00   1st Qu.:28.97   Private room   :8353
##  Fatih   :2025   Median :41.03   Median :28.98   Shared room    : 491
##  Kadikoy :1699   Mean   :41.03   Mean   :28.99
##  Besiktas:1320   3rd Qu.:41.05   3rd Qu.:29.02
##  Uskudar : 582   Max.   :41.41   Max.   :29.91
##  (Other) :3656
##      price       minimum_nights     number_of_reviews  last_review
##  Min.   :  0.0   Min.   :   1.000   Min.   :  0.000   Min.   :NA
##  1st Qu.:100.0   1st Qu.:   1.000   1st Qu.:  0.000   1st Qu.:NA
##  Median :179.0   Median :   1.000   Median :  0.000   Median :NA
##  Mean   :234.3   Mean   :   4.569   Mean   :  7.356   Mean   :NA
##  3rd Qu.:301.0   3rd Qu.:   2.000   3rd Qu.:  4.000   3rd Qu.:NA
##  Max.   :997.0   Max.   :1125.000   Max.   :307.000   Max.   :NA
##                                                       NA's   :15638
##  reviews_per_month calculated_host_listings_count availability_365
##  Min.   : 0.000    Min.   : 1.000                 Min.   :  0.0
##  1st Qu.: 0.000    1st Qu.: 1.000                 1st Qu.: 90.0
##  Median : 0.000    Median : 1.000                 Median :337.0
##  Mean   : 0.448    Mean   : 4.027                 Mean   :247.6
##  3rd Qu.: 0.490    3rd Qu.: 4.000                 3rd Qu.:365.0
##  Max.   :12.000    Max.   :77.000                 Max.   :365.0
##
```

```
# including all the categorical and numerical columns
Istanbul_Reg <- Istanbul.1[,c("neighbourhood","latitude","longitude","room_ty
pe","price","minimum_nights","number_of_reviews","reviews_per_month","calcula
ted_host_listings_count","availability_365")]
```

# Splitting Data into training and testing dataset into 80 /20 ratio

#Splitting done based on random values of Price .

```
library(caTools)

## Warning: package 'caTools' was built under R version 3.6.3

set.seed(123)
split = sample.split(Istanbul_Reg$price, SplitRatio = 0.8)
training_Istanbul = subset(Istanbul_Reg, split == TRUE)
test_Istanbul = subset(Istanbul_Reg, split == FALSE)

summary(training_Istanbul)

##   neighbourhood     latitude       longitude                 room_type
##  Beyoglu :3238   Min.   :40.81   Min.   :28.03   Entire home/apt:5422
##  Sisli   :1844   1st Qu.:41.00   1st Qu.:28.97   Private room   :6695
##  Fatih   :1610   Median :41.03   Median :28.98   Shared room    : 384
##  Kadikoy :1373   Mean   :41.03   Mean   :28.99
##  Besiktas:1059   3rd Qu.:41.05   3rd Qu.:29.02
##  Uskudar : 447   Max.   :41.41   Max.   :29.91
##  (Other) :2930
##      price       minimum_nights     number_of_reviews reviews_per_month
##  Min.   :  0.0   Min.   :   1.000   Min.   :  0.000   Min.   : 0.0000
##  1st Qu.:100.0   1st Qu.:   1.000   1st Qu.:  0.000   1st Qu.: 0.0000
##  Median :179.0   Median :   1.000   Median :  0.000   Median : 0.0000
##  Mean   :234.3   Mean   :   4.683   Mean   :  7.272   Mean   : 0.4419
##  3rd Qu.:301.0   3rd Qu.:   2.000   3rd Qu.:  4.000   3rd Qu.: 0.4900
##  Max.   :997.0   Max.   :1125.000   Max.   :307.000   Max.   :12.0000
##
##  calculated_host_listings_count availability_365
##  Min.   : 1.000                 Min.   :  0.0
##  1st Qu.: 1.000                 1st Qu.: 90.0
##  Median : 1.000                 Median :335.0
##  Mean   : 4.001                 Mean   :246.8
##  3rd Qu.: 4.000                 3rd Qu.:365.0
##  Max.   :77.000                 Max.   :365.0
##

summary(test_Istanbul)
```

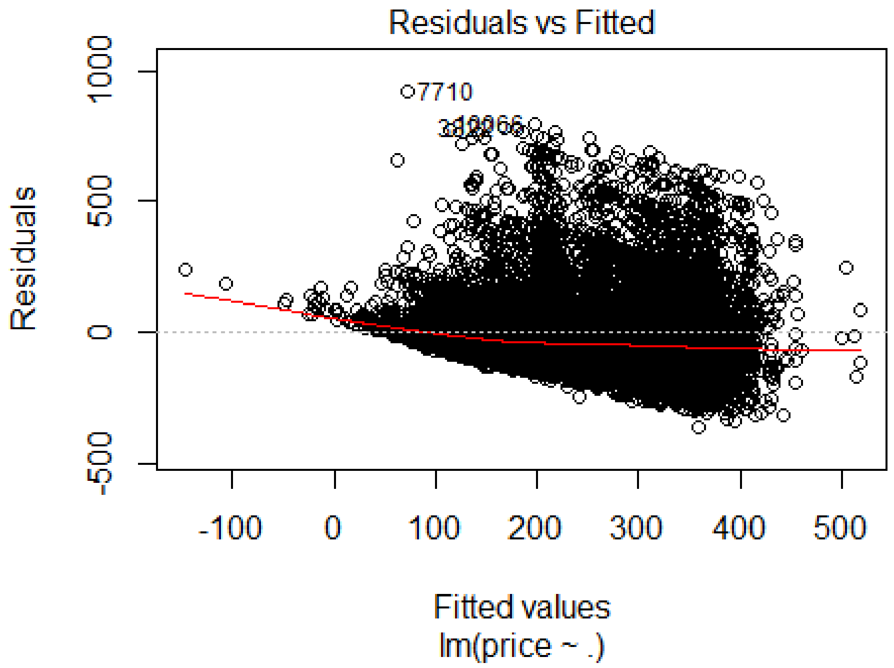
```
##   neighbourhood     latitude        longitude                room_type   
##  Beyoglu :863   Min.   :40.82   Min.   :28.17   Entire home/apt:1372  
##  Fatih   :415   1st Qu.:41.01   1st Qu.:28.97   Private room   :1658  
##  Sisli   :411   Median :41.03   Median :28.98   Shared room    : 107  
##  Kadikoy :326   Mean   :41.03   Mean   :28.98                         
##  Besiktas:261   3rd Qu.:41.05   3rd Qu.:29.02                         
##  Uskudar :135   Max.   :41.32   Max.   :29.86                         
##  (Other) :726                                                         
##      price        minimum_nights    number_of_reviews reviews_per_month
##  Min.   :  0.0   Min.   :  1.000   Min.   :  0.00    Min.   :0.0000   
##  1st Qu.:100.0   1st Qu.:  1.000   1st Qu.:  0.00    1st Qu.:0.0000   
##  Median :179.0   Median :  1.000   Median :  0.00    Median :0.0000   
##  Mean   :234.4   Mean   :  4.117   Mean   :  7.69    Mean   :0.4724   
##  3rd Qu.:301.0   3rd Qu.:  2.000   3rd Qu.:  4.00    3rd Qu.:0.5200   
##  Max.   :997.0   Max.   :900.000   Max.   :251.00    Max.   :7.5000   
##                                                                        
##  calculated_host_listings_count availability_365
##  Min.   : 1.000                 Min.   :  0.0   
##  1st Qu.: 1.000                 1st Qu.:110.0   
##  Median : 1.000                 Median :341.0   
##  Mean   : 4.129                 Mean   :250.8   
##  3rd Qu.: 4.000                 3rd Qu.:365.0   
##  Max.   :77.000                 Max.   :365.0   
## 
```

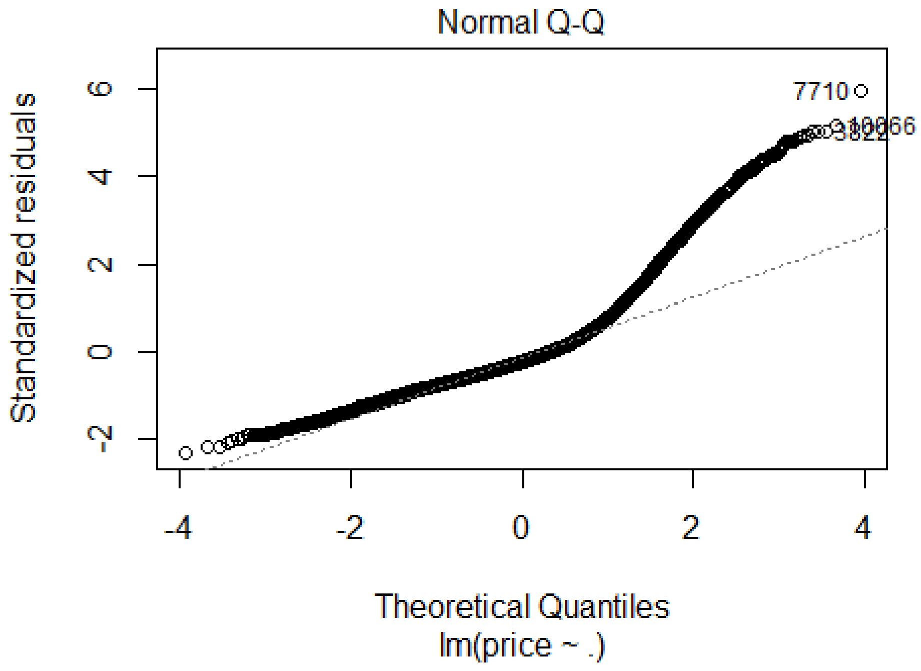
# Fitting Multiple Linear Regression to the Training set with all the independent vars.

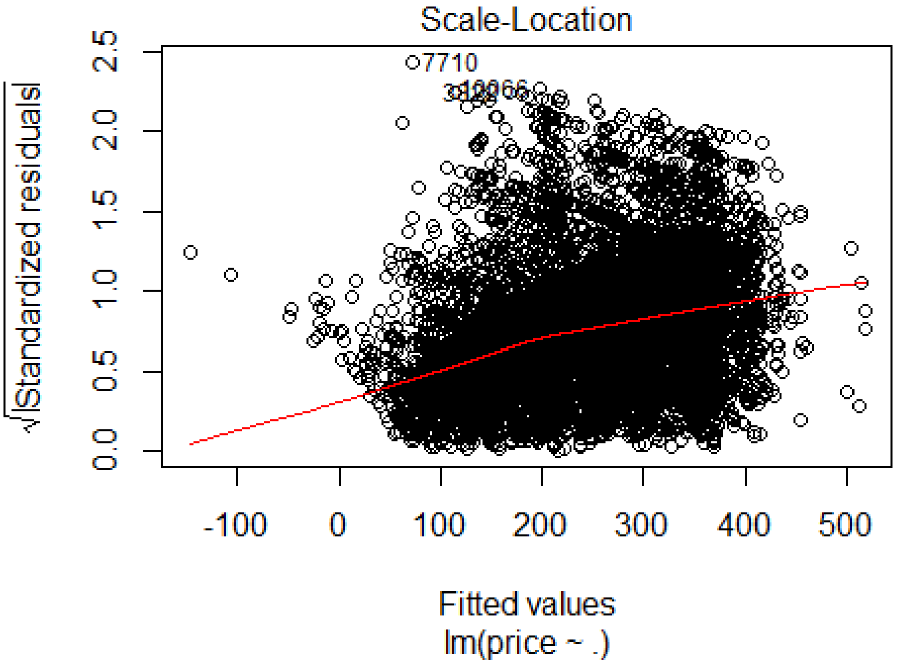
```
Istanbul_m1 = lm(formula = price ~ .,data = training_Istanbul)
summary(Istanbul_m1)#Adjusted R-squared:  0.2425 F-statistic: 86.14
```

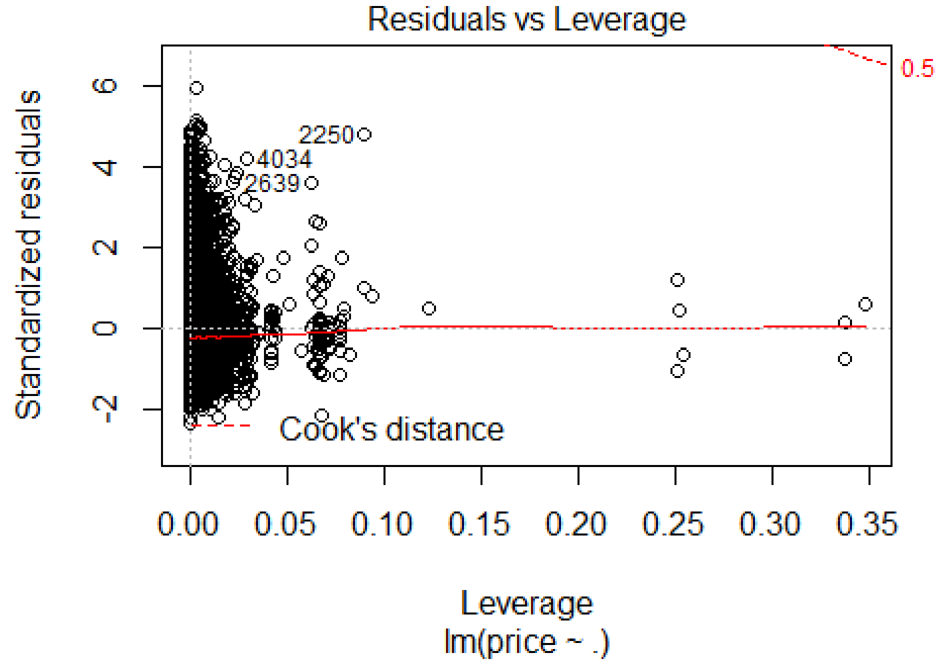
```
## 
## Call:
## lm(formula = price ~ ., data = training_Istanbul)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -359.35  -95.57  -37.65   49.90  918.85 
## 
## Coefficients:
##                                  Estimate Std. Error t value Pr(>|t|)    
## (Intercept)                    -1.058e+04  5.066e+03  -2.088 0.036809 *  
## neighbourhoodArnavutkoy         1.858e+02  8.920e+01   2.082 0.037326 *  
## neighbourhoodAtasehir          -9.150e+01  2.114e+01  -4.328 1.52e-05 ***
## neighbourhoodAvcilar            9.063e+01  3.696e+01   2.452 0.014210 *  
## neighbourhoodBagcilar           8.991e+01  3.432e+01   2.620 0.008811 ** 
## neighbourhoodBahcelievler       5.567e+01  2.941e+01   1.893 0.058383 .  
## neighbourhoodBakirkoy           9.694e+01 2.788e+01   3.477 0.000510 ***
## neighbourhoodBasaksehir         2.184e+02 3.889e+01   5.615 2.01e-08 ***
## neighbourhoodBayrampasa        -3.234e+01 4.933e+01  -0.656 0.512152
## neighbourhoodBesiktas          -1.291e+00 2.446e+01  -0.053 0.957900
## neighbourhoodBeykoz             9.300e-02 3.465e+01   0.003 0.997858
## neighbourhoodBeylikduzu         1.902e+02 4.394e+01   4.329 1.51e-05 ***
## neighbourhoodBeyoglu            3.616e+01 2.314e+01   1.562 0.118198
## neighbourhoodBuyukcekmece       2.571e+02 4.811e+01   5.345 9.21e-08 ***
## neighbourhoodCatalca            4.037e+02 1.124e+02   3.591 0.000331 ***
## neighbourhoodCekmekoy          -1.314e+02 3.576e+01  -3.675 0.000239 ***
## neighbourhoodEsenler            6.267e+01 4.872e+01   1.287 0.198289
## neighbourhoodEsenyurt           1.327e+02 3.936e+01   3.371 0.000752 ***
## neighbourhoodEyup               3.014e+01 3.335e+01   0.904 0.366176
## neighbourhoodFatih              7.707e+01 2.210e+01   3.487 0.000490 ***
## neighbourhoodGaziosmanpasa      6.412e+01 3.609e+01   1.777 0.075636 .
## neighbourhoodGungoren           4.613e+01 3.334e+01   1.383 0.166548
## neighbourhoodKadikoy           -6.462e+01 1.885e+01  -3.428 0.000610 ***
## neighbourhoodKagithane         -2.805e+00 2.829e+01  -0.099 0.921029
## neighbourhoodKartal            -1.620e+02 2.201e+01  -7.358 1.98e-13 ***
## neighbourhoodKucukcekmece       7.170e+01 3.470e+01   2.067 0.038791 *
## neighbourhoodMaltepe           -1.175e+02 1.893e+01  -6.208 5.53e-10 ***
## neighbourhoodPendik            -2.447e+02 2.485e+01  -9.849  < 2e-16 ***
## neighbourhoodSancaktepe        -1.960e+02 3.794e+01  -5.166 2.42e-07 ***
## neighbourhoodSariyer            1.895e-01 3.174e+01   0.006 0.995236
## neighbourhoodSile              -2.897e+02 6.475e+01  -4.475 7.72e-06 ***
## neighbourhoodSilivri            5.127e+02 7.974e+01   6.430 1.33e-10 ***
## neighbourhoodSisli              2.146e+01 2.489e+01   0.862 0.388615
## neighbourhoodSultanbeyli       -2.181e+02 4.536e+01  -4.807 1.55e-06 ***
## neighbourhoodSultangazi         3.530e+01 5.313e+01   0.664 0.506465
## neighbourhoodTuzla             -2.260e+02 3.133e+01  -7.214 5.75e-13 ***
## neighbourhoodUmraniye          -1.114e+02 2.494e+01  -4.469 7.93e-06 ***
## neighbourhoodUskudar           -5.361e+01 2.229e+01  -2.405 0.016181 *
## neighbourhoodZeytinburnu        5.556e+01 3.135e+01   1.772 0.076376 .
## latitude                       -1.251e+02 1.065e+02  -1.174 0.240344
## longitude                       5.515e+02 7.520e+01   7.334 2.37e-13 ***
## room_typePrivate room          -1.514e+02 2.962e+00 -51.105  < 2e-16 ***
## room_typeShared room           -1.824e+02 8.289e+00 -22.006  < 2e-16 ***
## minimum_nights                 -9.873e-02 4.663e-02  -2.117 0.034272 *
## number_of_reviews              -1.159e-01 8.037e-02  -1.443 0.149158
## reviews_per_month              -2.703e+01 1.969e+00 -13.725  < 2e-16 ***
## calculated_host_listings_count -1.858e-01 1.944e-01  -0.956 0.339232
## availability_365                1.746e-01 1.041e-02  16.768  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 154.8 on 12453 degrees of freedom
## Multiple R-squared:  0.2453, Adjusted R-squared:  0.2425
## F-statistic: 86.14 on 47 and 12453 DF,  p-value: < 2.2e-16
```

```
plot(Istanbul_m1)
```


Residuals vs Fitted
Residuals
1000
500
0
-500
7710
3822
10066
-100
0
100
200
300
400
500
Fitted values
lm(price ~ .)


Normal Q-Q
Standardized residuals
6
4
2
0
-2
7710
10066
3822
-4
-2
0
2
4
Theoretical Quantiles
lm(price ~ .)

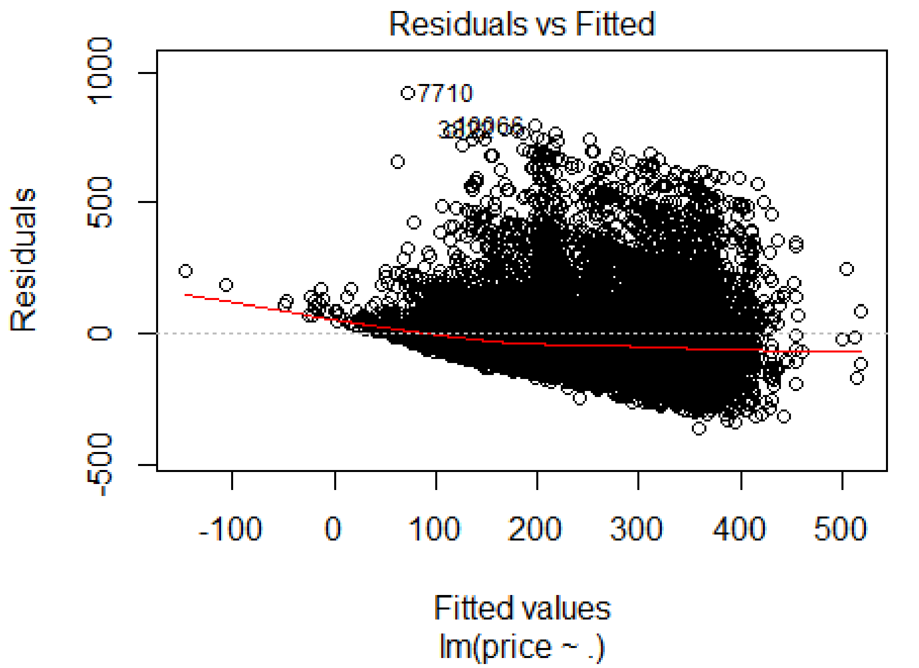


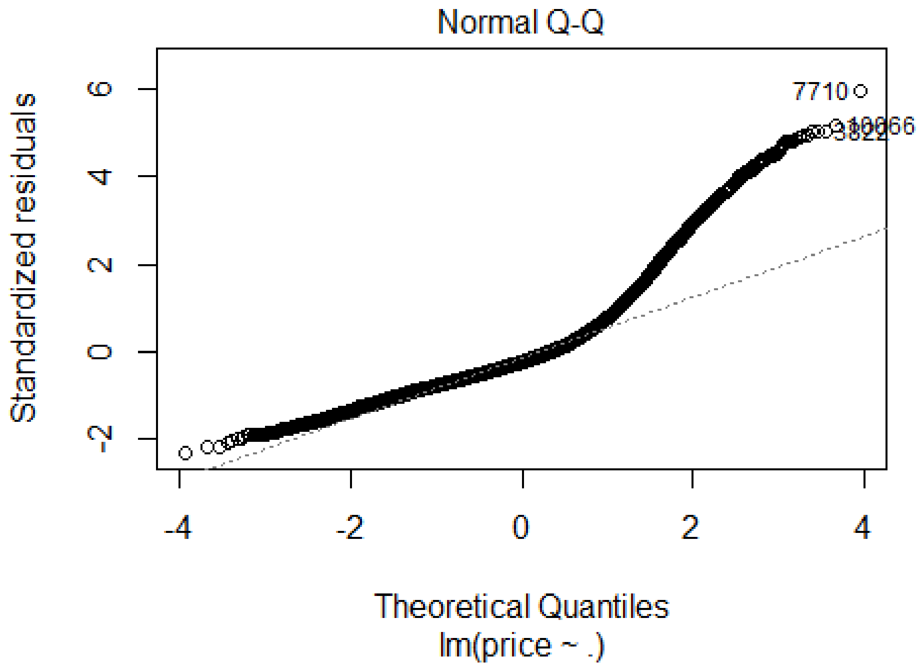


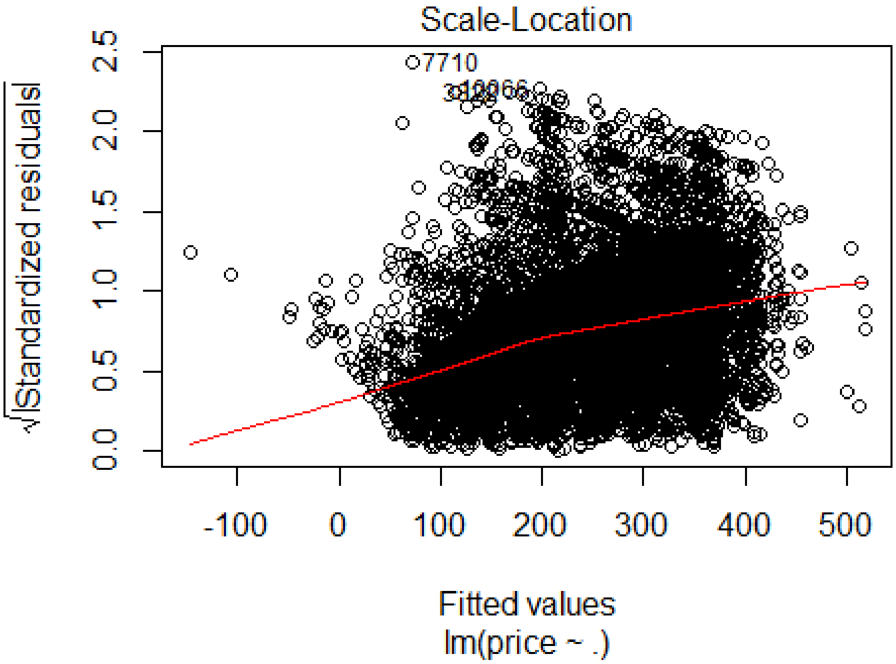
```
## too many categorical columns for neighbourhood and not too significant for
my neighbourhoods. Hence dropping it.
Istanbul_m2 = lm(price ~ latitude+longitude+room_type+minimum_nights+number_o
f_reviews+reviews_per_month+calculated_host_listings_count+availability_365,d
ata = training_Istanbul)
summary(Istanbul_m2)#Adjusted R-squared:  0.2085 F-statistic: 366.9
```

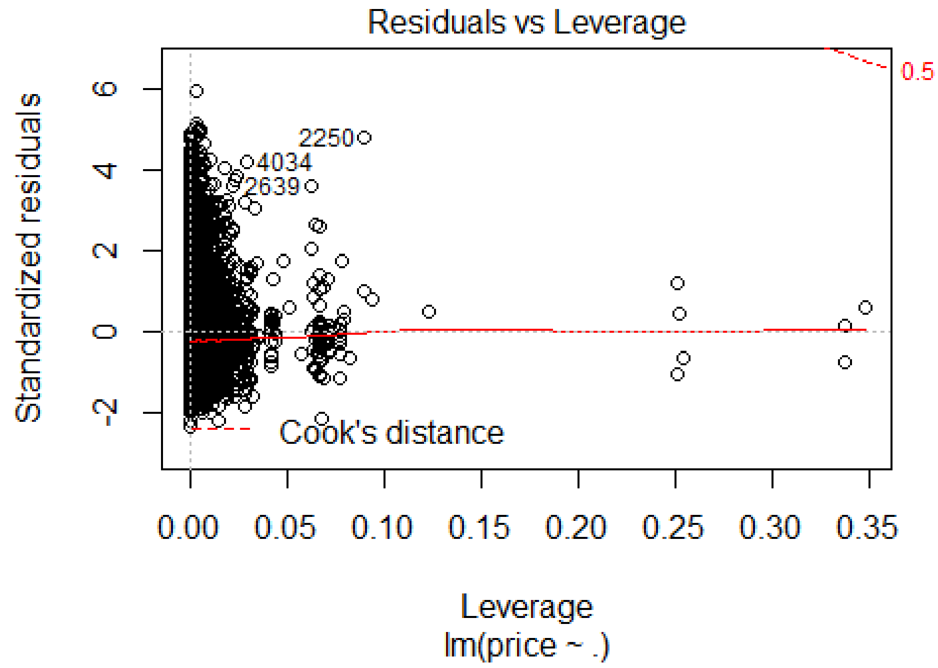
```
##
## Call:
## lm(formula = price ~ latitude + longitude + room_type + minimum_nights +
##     number_of_reviews + reviews_per_month + calculated_host_listings_count
+
##     availability_365, data = training_Istanbul)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -353.06  -97.50  -41.87   51.94  874.40
##
## Coefficients:
##                                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    -7.139e+03  1.524e+03  -4.685 2.83e-06 ***
## latitude                        1.726e+02  3.413e+01   5.057 4.33e-07 ***
## longitude                       1.170e+01  1.297e+01   0.902 0.366991
## room_typePrivate room          -1.521e+02  2.977e+00 -51.100  < 2e-16 ***
## room_typeShared room           -1.933e+02  8.415e+00 -22.969  < 2e-16 ***
## minimum_nights                 -1.226e-01  4.760e-02  -2.575 0.010025 *
## number_of_reviews               4.769e-02  8.153e-02   0.585 0.558569
## reviews_per_month              -2.532e+01  1.992e+00 -12.710  < 2e-16 ***
## calculated_host_listings_count  7.218e-01  1.922e-01   3.756 0.000174 ***
## availability_365                1.996e-01  1.053e-02  18.959  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 158.3 on 12491 degrees of freedom
## Multiple R-squared:  0.2091, Adjusted R-squared:  0.2085
## F-statistic: 366.9 on 9 and 12491 DF,  p-value: < 2.2e-16
```

```
plot(Istanbul_m1)
```


Residuals vs Fitted
Residuals
1000
500
0
-500
7710
3822
10066
-100
0
100
200
300
400
500
Fitted values
lm(price ~ .)


Normal Q-Q
Standardized residuals
6
4
2
0
-2
7710
10066
3822
-4
-2
0
2
4
Theoretical Quantiles
lm(price ~ .)





```
# dropping longitude var as p-value > .05
Istanbul_m3 = lm(price ~ latitude+room_type+minimum_nights+reviews_per_month+
calculated_host_listings_count+availability_365,data = training_Istanbul)
summary(Istanbul_m3)#Adjusted R-squared:  0.2086 F-statistic: 471.6
```

```
##
## Call:
## lm(formula = price ~ latitude + room_type + minimum_nights +
##     reviews_per_month + calculated_host_listings_count + availability_365,
##     data = training_Istanbul)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -353.24  -97.58  -41.88   51.75  874.35
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   -6.537e+03  1.370e+03  -4.772 1.84e-06 ***
## latitude                       1.662e+02  3.338e+01   4.978 6.51e-07 ***
## room_typePrivate room         -1.521e+02  2.966e+00 -51.268  < 2e-16 ***
## room_typeShared room          -1.935e+02  8.412e+00 -23.003  < 2e-16 ***
## minimum_nights                -1.229e-01  4.759e-02  -2.582 0.009845 **
## reviews_per_month             -2.465e+01  1.659e+00 -14.859  < 2e-16 ***
## calculated_host_listings_count  7.312e-01  1.907e-01   3.834 0.000127 ***
## availability_365               1.995e-01  1.051e-02  18.984  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 158.3 on 12493 degrees of freedom
## Multiple R-squared:  0.209,  Adjusted R-squared:  0.2086
## F-statistic: 471.6 on 7 and 12493 DF,  p-value: < 2.2e-16

## number of reviews and review per month have multicollinearity

##Hence keeping only 1 of them.

Istanbul_m4 = lm(price ~ latitude+room_type+minimum_nights+number_of_reviews+
calculated_host_listings_count+availability_365,data = training_Istanbul)
summary(Istanbul_m4)#Adjusted R-squared:  0.1984 F-statistic: 442.9

##
## Call:
## lm(formula = price ~ latitude + room_type + minimum_nights +
##     number_of_reviews + calculated_host_listings_count + availability_365,
##     data = training_Istanbul)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -344.84  -98.59  -44.03   50.45  877.46
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   -7.101e+03  1.378e+03  -5.154 2.59e-07 ***
## latitude                       1.797e+02  3.358e+01   5.350 8.96e-08 ***
## room_typePrivate room         -1.469e+02  2.964e+00 -49.571  < 2e-16 ***
## room_typeShared room           -1.859e+02  8.447e+00 -22.007  < 2e-16 ***
## minimum_nights                 -9.957e-02  4.786e-02  -2.080 0.037522 *
## number_of_reviews              -5.254e-01  6.833e-02  -7.688 1.60e-14 ***
## calculated_host_listings_count  7.165e-01  1.934e-01   3.706 0.000212 ***
## availability_365                2.048e-01  1.058e-02  19.359  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 159.3 on 12493 degrees of freedom
## Multiple R-squared:  0.1988, Adjusted R-squared:  0.1984
## F-statistic: 442.9 on 7 and 12493 DF,  p-value: < 2.2e-16
```

#Checking for best model with Step function

```
stepIstanbul <- step(Istanbul_m1, direction = "backward") ## full model
```

```
## Start:  AIC=126117
## price ~ neighbourhood + latitude + longitude + room_type + minimum_nights
+
##     number_of_reviews + reviews_per_month + calculated_host_listings_count
+
##     availability_365
##
##                                  Df Sum of Sq       RSS    AIC
## - calculated_host_listings_count  1     21898 298569054 126116
## - latitude                        1     33053 298580209 126116
## <none>                                        298547156 126117
## - number_of_reviews               1     49892 298597048 126117
## - minimum_nights                  1    107452 298654608 126120
## - longitude                       1   1289636 299836792 126169
## - reviews_per_month               1   4516051 303063207 126303
## - availability_365                1   6740633 305287789 126394
## - neighbourhood                  38  14343316 312890472 126628
## - room_type                       2  65660894 364208050 128598
##
## Step:  AIC=126115.9
## price ~ neighbourhood + latitude + longitude + room_type + minimum_nights
+
##     number_of_reviews + reviews_per_month + availability_365
##
##                     Df Sum of Sq       RSS    AIC
## - latitude           1     31214 298600268 126115
## <none>                           298569054 126116
## - number_of_reviews  1     56429 298625483 126116
## - minimum_nights     1    105952 298675006 126118
## - longitude          1   1279541 299848595 126167
## - reviews_per_month  1   4508361 303077415 126301
## - availability_365   1   6751696 305320750 126393
## - neighbourhood     38  14674710 313243764 126640
## - room_type          2  65735673 364304727 128599
## 
## Step:  AIC=126115.2
## price ~ neighbourhood + longitude + room_type + minimum_nights + 
##     number_of_reviews + reviews_per_month + availability_365
## 
##                      Df Sum of Sq       RSS    AIC
## <none>                            298600268 126115
## - number_of_reviews  1     55364 298655632 126116
## - minimum_nights     1    105708 298705976 126118
## - longitude          1   1334415 299934683 126169
## - reviews_per_month  1   4494942 303095210 126300
## - availability_365   1   6772819 305373087 126394
## - neighbourhood     38  15275268 313875536 126663
## - room_type          2  65715141 364315409 128598
```

```
stepIstanbul
```

```
## 
## Call:
## lm(formula = price ~ neighbourhood + longitude + room_type + 
##     minimum_nights + number_of_reviews + reviews_per_month + 
##     availability_365, data = training_Istanbul)
## 
## Coefficients:
##                (Intercept)     neighbourhoodArnavutkoy  
##                 -1.587e+04                   1.482e+02  
##      neighbourhoodAtasehir        neighbourhoodAvcilar  
##                 -1.055e+02                   7.842e+01  
##      neighbourhoodBagcilar   neighbourhoodBahcelievler  
##                  7.103e+01                   4.185e+01  
##      neighbourhoodBakirkoy     neighbourhoodBasaksehir  
##                  8.564e+01                   1.944e+02  
##    neighbourhoodBayrampasa       neighbourhoodBesiktas  
##                 -5.313e+01                  -2.318e+01  
##        neighbourhoodBeykoz     neighbourhoodBeylikduzu  
##                 -2.880e+01                   1.780e+02  
##       neighbourhoodBeyoglu   neighbourhoodBuyukcekmece  
##                  1.675e+01                   2.427e+02  
##       neighbourhoodCatalca       neighbourhoodCekmekoy  
##                  3.544e+02                  -1.527e+02  
##       neighbourhoodEsenler       neighbourhoodEsenyurt  
##                  4.261e+01                   1.168e+02  
##          neighbourhoodEyup          neighbourhoodFatih  
##                  4.331e+00                   6.030e+01  
## neighbourhoodGaziosmanpasa       neighbourhoodGungoren  
##                  4.112e+01                   3.038e+01  
##       neighbourhoodKadikoy      neighbourhoodKagithane  
##                 -7.800e+01                  -2.854e+01
##        neighbourhoodKartal   neighbourhoodKucukcekmece
##                 -1.663e+02                   5.591e+01
##       neighbourhoodMaltepe         neighbourhoodPendik
##                 -1.258e+02                  -2.506e+02
##    neighbourhoodSancaktepe        neighbourhoodSariyer
##                 -2.109e+02                  -3.043e+01
##          neighbourhoodSile        neighbourhoodSilivri
##                 -3.275e+02                   4.921e+02
##         neighbourhoodSisli    neighbourhoodSultanbeyli
##                 -7.790e-01                  -2.302e+02
##    neighbourhoodSultangazi          neighbourhoodTuzla
##                  8.423e+00                  -2.241e+02
##      neighbourhoodUmraniye        neighbourhoodUskudar
##                 -1.287e+02                  -7.139e+01
##   neighbourhoodZeytinburnu                   longitude
##                  4.089e+01                   5.578e+02
##      room_typePrivate room        room_typeShared room
##                 -1.512e+02                  -1.822e+02
##             minimum_nights           number_of_reviews
##                 -9.791e-02                  -1.216e-01
##          reviews_per_month            availability_365
##                 -2.696e+01                   1.735e-01
```

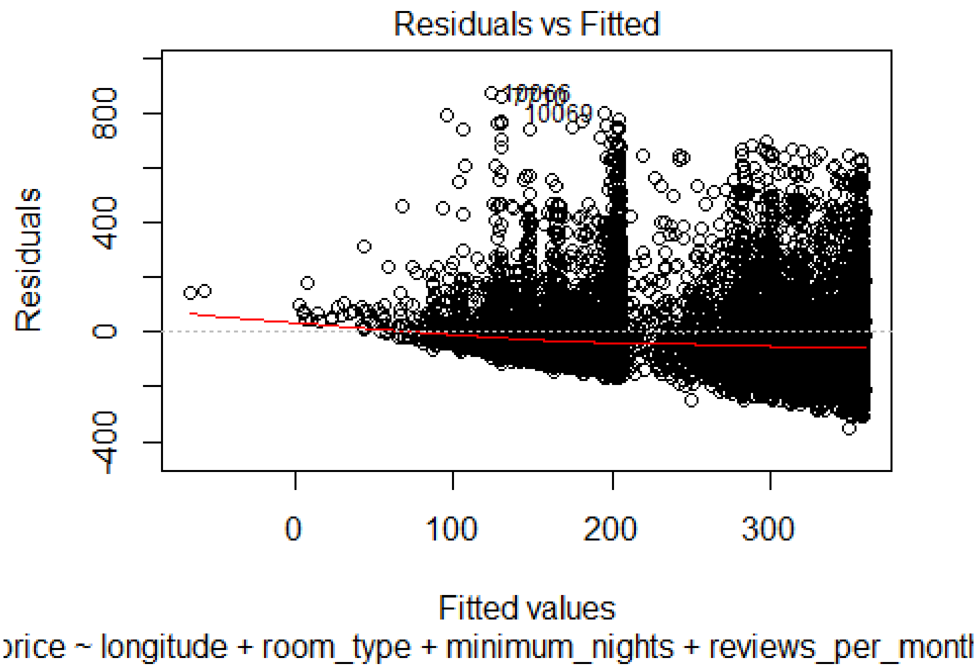
```
summary(stepIstanbul)
```

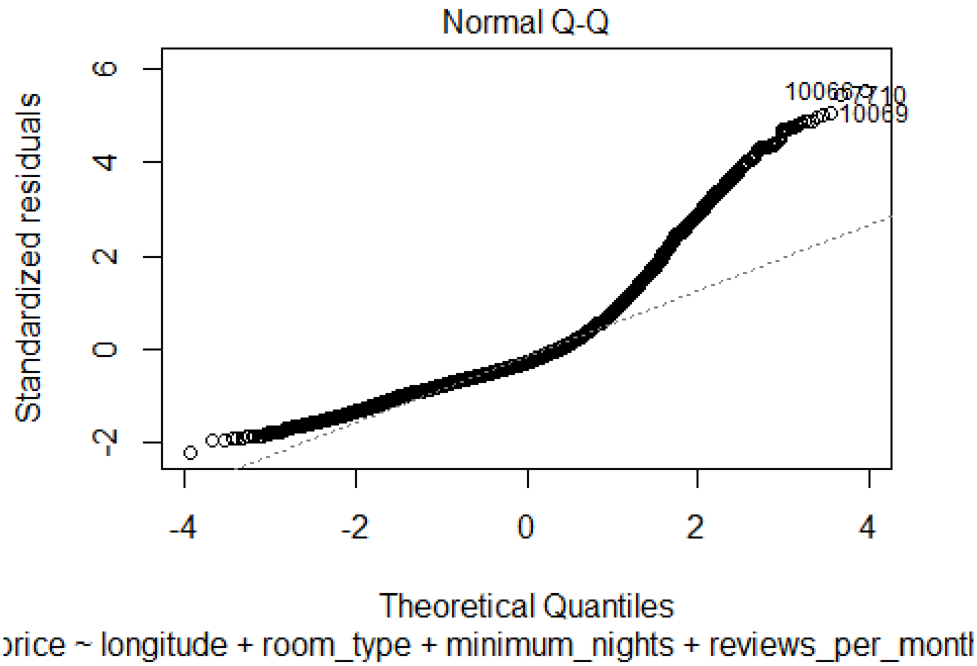
```
## 
## Call:
## lm(formula = price ~ neighbourhood + longitude + room_type + 
##     minimum_nights + number_of_reviews + reviews_per_month + 
##     availability_365, data = training_Istanbul)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -359.70  -95.36  -37.59   49.95  919.31 
## 
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)    
## (Intercept)                -1.587e+04  2.176e+03  -7.294 3.18e-13 ***
## neighbourhoodArnavutkoy     1.482e+02  8.353e+01   1.774 0.076104 .  
## neighbourhoodAtasehir      -1.055e+02  1.755e+01  -6.010 1.91e-09 ***
## neighbourhoodAvcilar        7.842e+01  3.568e+01   2.198 0.027996 *  
## neighbourhoodBagcilar       7.103e+01  3.068e+01   2.315 0.020620 *  
## neighbourhoodBahcelievler   4.185e+01  2.716e+01   1.541 0.123326    
## neighbourhoodBakirkoy       8.564e+01  2.634e+01   3.251 0.001152 ** 
## neighbourhoodBasaksehir     1.944e+02  3.361e+01   5.785 7.42e-09 ***
## neighbourhoodBayrampasa    -5.313e+01  4.626e+01  -1.149 0.250731    
## neighbourhoodBesiktas      -2.318e+01  1.608e+01  -1.442 0.149444    
## neighbourhoodBeykoz        -2.880e+01  2.439e+01  -1.181 0.237765    
## neighbourhoodBeylikduzu     1.780e+02  4.291e+01   4.148 3.37e-05 ***
## neighbourhoodBeyoglu        1.675e+01  1.685e+01   0.994 0.320229
## neighbourhoodBuyukcekmece   2.427e+02  4.680e+01   5.185 2.19e-07 ***
## neighbourhoodCatalca        3.544e+02  1.048e+02   3.381 0.000725 ***
## neighbourhoodCekmekoy      -1.527e+02  3.079e+01  -4.957 7.24e-07 ***
## neighbourhoodEsenler        4.261e+01  4.579e+01   0.931 0.352095
## neighbourhoodEsenyurt       1.168e+02  3.733e+01   3.130 0.001749 **
## neighbourhoodEyup           4.331e+00  2.534e+01   0.171 0.864271
## neighbourhoodFatih          6.030e+01  1.771e+01   3.404 0.000665 ***
## neighbourhoodGaziosmanpasa  4.112e+01  3.053e+01   1.347 0.178068
## neighbourhoodGungoren       3.038e+01  3.079e+01   0.987 0.323757
## neighbourhoodKadikoy       -7.800e+01  1.511e+01  -5.163 2.47e-07 ***
## neighbourhoodKagithane     -2.854e+01  1.862e+01  -1.533 0.125416
## neighbourhoodKartal        -1.663e+02  2.170e+01  -7.665 1.92e-14 ***
## neighbourhoodKucukcekmece   5.591e+01  3.222e+01   1.735 0.082770 .
## neighbourhoodMaltepe       -1.258e+02  1.752e+01  -7.178 7.46e-13 ***
## neighbourhoodPendik        -2.506e+02  2.422e+01 -10.348  < 2e-16 ***
## neighbourhoodSancaktepe    -2.109e+02  3.559e+01  -5.927 3.17e-09 ***
## neighbourhoodSariyer       -3.043e+01  1.819e+01  -1.673 0.094427 .
## neighbourhoodSile          -3.275e+02  5.524e+01  -5.928 3.14e-09 ***
## neighbourhoodSilivri        4.921e+02  7.816e+01   6.297 3.15e-10 ***
## neighbourhoodSisli         -7.790e-01  1.662e+01  -0.047 0.962610
## neighbourhoodSultanbeyli   -2.302e+02  4.403e+01  -5.228 1.74e-07 ***
## neighbourhoodSultangazi     8.423e+00  4.813e+01   0.175 0.861064
## neighbourhoodTuzla         -2.241e+02  3.131e+01  -7.158 8.63e-13 ***
## neighbourhoodUmraniye      -1.287e+02  2.009e+01  -6.406 1.55e-10 ***
## neighbourhoodUskudar       -7.139e+01  1.642e+01  -4.346 1.39e-05 ***
## neighbourhoodZeytinburnu    4.089e+01  2.892e+01   1.414 0.157371
## longitude                   5.578e+02  7.476e+01   7.461 9.19e-14 ***
## room_typePrivate room      -1.512e+02  2.958e+00 -51.120  < 2e-16 ***
## room_typeShared room       -1.822e+02  8.286e+00 -21.987  < 2e-16 ***
## minimum_nights             -9.791e-02  4.663e-02  -2.100 0.035765 *
## number_of_reviews          -1.216e-01  8.004e-02  -1.520 0.128628
## reviews_per_month          -2.696e+01  1.969e+00 -13.693  < 2e-16 ***
## availability_365            1.735e-01  1.032e-02  16.808  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 154.8 on 12455 degrees of freedom
## Multiple R-squared:  0.2452, Adjusted R-squared:  0.2425
## F-statistic: 89.92 on 45 and 12455 DF,  p-value: < 2.2e-16
```

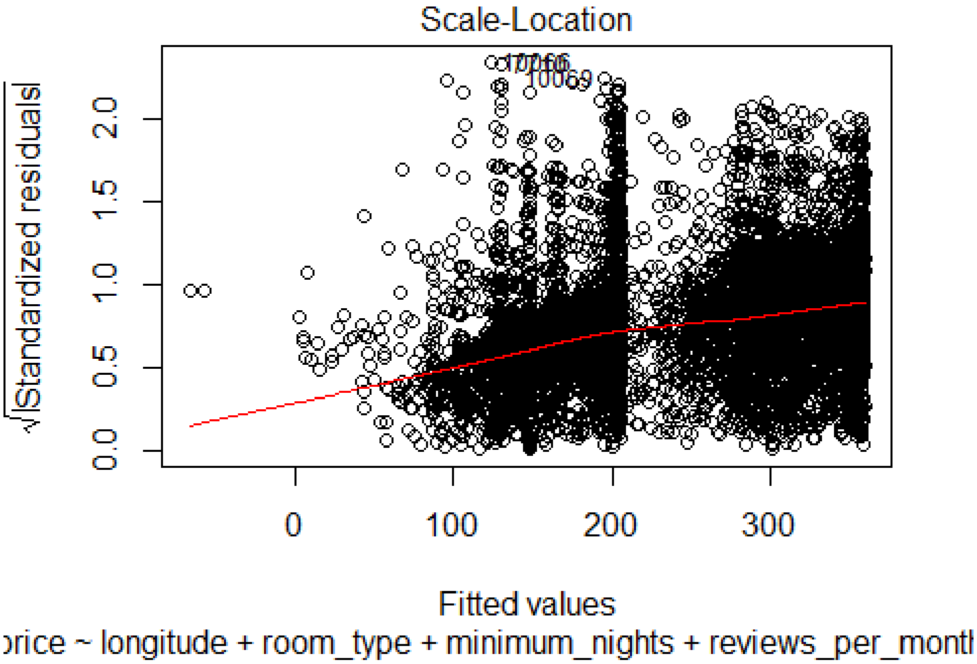
```
#Trying with longitude instead of lattitude, also taking out calc.hostlisting
AS PER Stepaic output


Istanbul_m5= lm(price ~ longitude+room_type+minimum_nights+reviews_per_month+
availability_365,data = training_Istanbul)
summary(Istanbul_m5)#Adjusted R-squared:  0.2065 F-statistic: 542
```

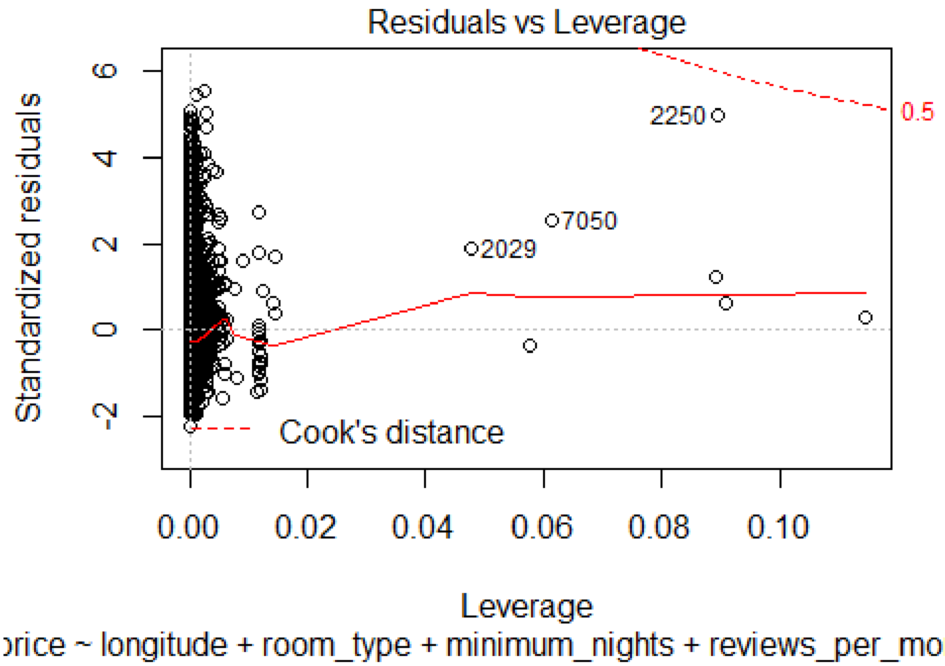
```
## 
## Call:
## lm(formula = price ~ longitude + room_type + minimum_nights + 
##     reviews_per_month + availability_365, data = training_Istanbul)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -349.13  -98.81  -43.23   52.20  872.89 
## 
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)    
## (Intercept)            371.89304  368.17241   1.010  0.31246    
## longitude               -3.06784   12.70195  -0.242  0.80915    
## room_typePrivate room -153.19840    2.97037 -51.575  < 2e-16 ***
## room_typeShared room  -195.77763    8.41636 -23.262  < 2e-16 ***
## minimum_nights          -0.12690    0.04766  -2.663  0.00776 ** 
## reviews_per_month      -24.43524    1.65485 -14.766  < 2e-16 ***
## availability_365         0.20560    0.01038  19.808  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 158.5 on 12494 degrees of freedom
## Multiple R-squared:  0.2065, Adjusted R-squared:  0.2061 
## F-statistic:   542 on 6 and 12494 DF,  p-value: < 2.2e-16
```

```
plot(Istanbul_m5)
```


Residuals vs Fitted
Residuals
Fitted values
ɔrice ~ longitude + room_type + minimum_nights + reviews_per_montl
800
400
0
-400
0
100
200
300
10066
17710
10069


Normal Q-Q
Standardized residuals
Theoretical Quantiles
ɔrice ~ longitude + room_type + minimum_nights + reviews_per_montl
6
4
2
0
-2
-4
-2
0
2
4
10066
17710
10069





```
#Removing longitude as its not significant
Istanbul_m6= lm(price ~ room_type+minimum_nights+reviews_per_month+availabili
ty_365,data = training_Istanbul)
summary(Istanbul_m6)
```

```
##
## Call:
## lm(formula = price ~ room_type + minimum_nights + reviews_per_month +
##     availability_365, data = training_Istanbul)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -349.13  -98.94  -43.19   52.09  872.89
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)            282.97426    3.53582  80.031  < 2e-16 ***
## room_typePrivate room -153.24610    2.96369 -51.708  < 2e-16 ***
## room_typeShared room  -195.76541    8.41589 -23.261  < 2e-16 ***
## minimum_nights          -0.12679    0.04765  -2.661  0.00781 **
## reviews_per_month      -24.44406    1.65439 -14.775  < 2e-16 ***
## availability_365         0.20570    0.01037  19.833  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 158.5 on 12495 degrees of freedom
## Multiple R-squared:  0.2065, Adjusted R-squared:  0.2062
## F-statistic: 650.4 on 5 and 12495 DF,  p-value: < 2.2e-16
```

```
# Residual standard error: 158.5 on 12495 degrees of freedom
# Multiple R-squared:  0.2065,  Adjusted R-squared:  0.2062
# F-statistic: 650.4 on 5 and 12495 DF,  p-value: < 2.2e-16
summary(Istanbul_m3)
```

```
##
## Call:
## lm(formula = price ~ latitude + room_type + minimum_nights +
##     reviews_per_month + calculated_host_listings_count + availability_365,
##     data = training_Istanbul)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -353.24  -97.58  -41.88   51.75  874.35
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   -6.537e+03  1.370e+03  -4.772 1.84e-06 ***
## latitude                       1.662e+02  3.338e+01   4.978 6.51e-07 ***
## room_typePrivate room         -1.521e+02  2.966e+00 -51.268  < 2e-16 ***
## room_typeShared room          -1.935e+02  8.412e+00 -23.003  < 2e-16 ***
## minimum_nights                -1.229e-01  4.759e-02  -2.582 0.009845 **
## reviews_per_month             -2.465e+01  1.659e+00 -14.859  < 2e-16 ***
## calculated_host_listings_count  7.312e-01  1.907e-01   3.834 0.000127 ***
## availability_365               1.995e-01  1.051e-02  18.984  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 158.3 on 12493 degrees of freedom
## Multiple R-squared:  0.209,  Adjusted R-squared:  0.2086
## F-statistic: 471.6 on 7 and 12493 DF,  p-value: < 2.2e-16
```

```
#Residual standard error: 158.3 on 12493 degrees of freedom
#Multiple R-squared:  0.209,    Adjusted R-squared:  0.2086
#F-statistic: 471.6 on 7 and 12493 DF,  p-value: < 2.2e-16

AIC(Istanbul_m3)
```

```
## [1] 162103.1
```

```
AIC(Istanbul_m6)
```

```
## [1] 162138.4
```

## comparing models with anova##

```
anova(Istanbul_m3,Istanbul_m6)
```

```
## Analysis of Variance Table
##
## Model 1: price ~ latitude + room_type + minimum_nights + reviews_per_month
+
##     calculated_host_listings_count + availability_365
## Model 2: price ~ room_type + minimum_nights + reviews_per_month + availabi
lity_365
##   Res.Df       RSS Df Sum of Sq      F    Pr(>F)
## 1  12493 312919412
## 2  12495 313904125 -2   -984713 19.657 2.996e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
AIC(Istanbul_m3)
```

```
## [1] 162103.1
```

```
#anova(fit1, fit2)
#step <- stepAIC(Istanbul_m1, direction="both")
#step$anova # display results
```

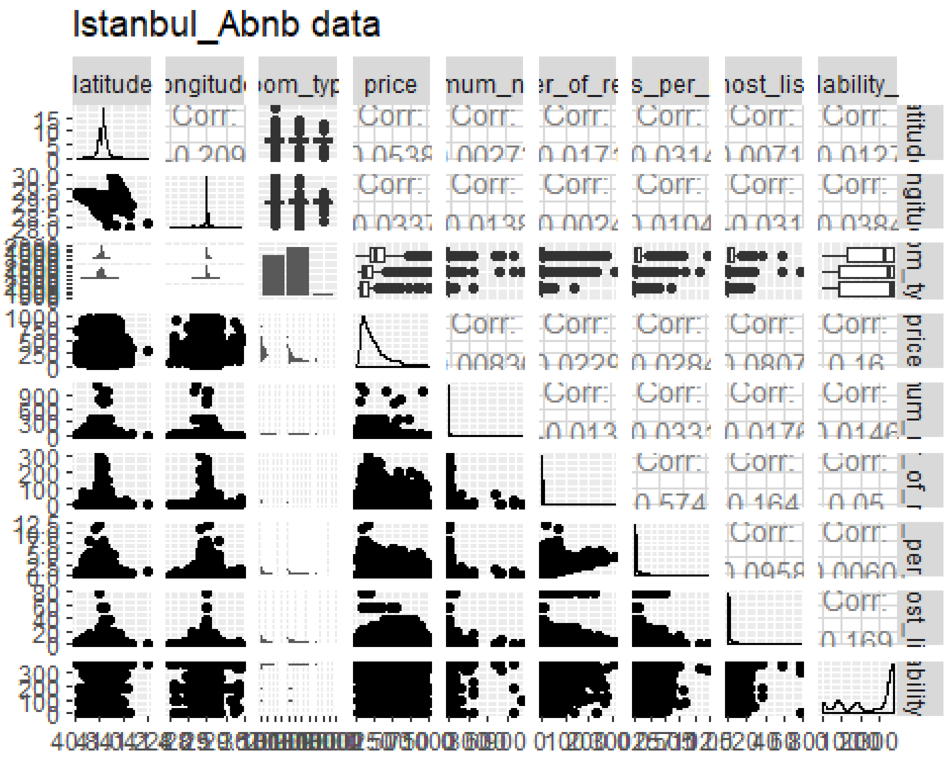
#This shows that AIC reduces a little when 'Calculated host listings' and lattitude are included.

#We would go ahead with model 'Istanbul_m3' as it has high Fstat, its

including at least 1 of the locations variable and its covering little more variance than other models.

```
ggpairs(data=training_Istanbul[,-1], title="Istanbul_Abnb data")
```

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



#As per pairs plot, there is not much correlation between variables and Price

#Number of reviews and reviews per month is correlated so we are

including only 1 of them in regression.

```
#Printing the range of coefficients with 95% confidence intervals
x=confint(Istanbul_m3,level=0.95)
x
```

```
##                                        2.5 %        97.5 %
## (Intercept)                     -9221.9273665 -3.852139e+03
## latitude                          100.7503434  2.316271e+02
## room_typePrivate room            -157.8913245 -1.462625e+02
## room_typeShared room             -209.9838496 -1.770070e+02
## minimum_nights                     -0.2161567 -2.957798e-02
## reviews_per_month                 -27.8974032 -2.139482e+01
## calculated_host_listings_count      0.3573903  1.105071e+00
## availability_365                    0.1789202  2.201237e-01
```

# Assessing Outliers

```
library(car)
```

```
## Warning: package 'car' was built under R version 3.6.3
```

```
## Loading required package: carData
```

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

```
## The following object is masked from 'package:psych':
## 
##     logit
```

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

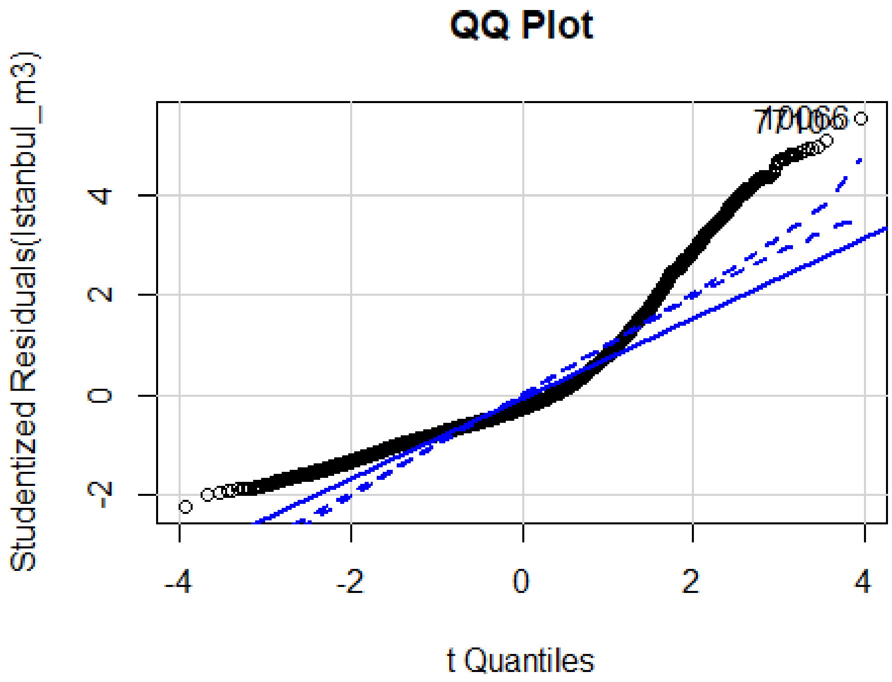
```
## The following object is masked from 'package:purrr':
## 
##     some
```

```
outlierTest(Istanbul_m3) #
```

```
##       rstudent unadjusted p-value Bonferroni p
## 10066 5.538459         3.1131e-08   0.00038917
## 7710  5.454939         4.9923e-08   0.00062408
## 10069 5.099618         3.4532e-07   0.00431690
## 10109 4.952181         7.4346e-07   0.00929390
## 2250  4.942391         7.8171e-07   0.00977220
## 11885 4.914136         9.0306e-07   0.01128900
## 9688  4.887561         1.0336e-06   0.01292100
## 10298 4.855595         1.2148e-06   0.01518600
## 226   4.824827         1.4178e-06   0.01772400
## 11121 4.821643         1.4406e-06   0.01800900
```

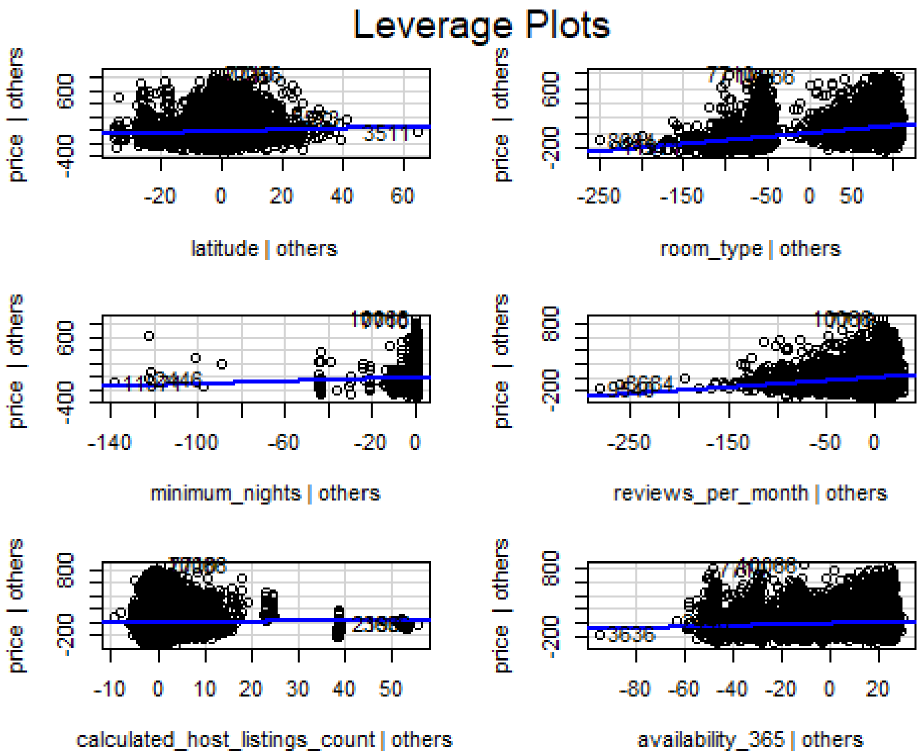
```
#Outliers are detected for the observations where Price is very high(Price>90
0)
#The record at given row numbers are outliers.
```

```
#qqplot for plotting studentized residuals
qqPlot(Istanbul_m3, main="QQ Plot")
```

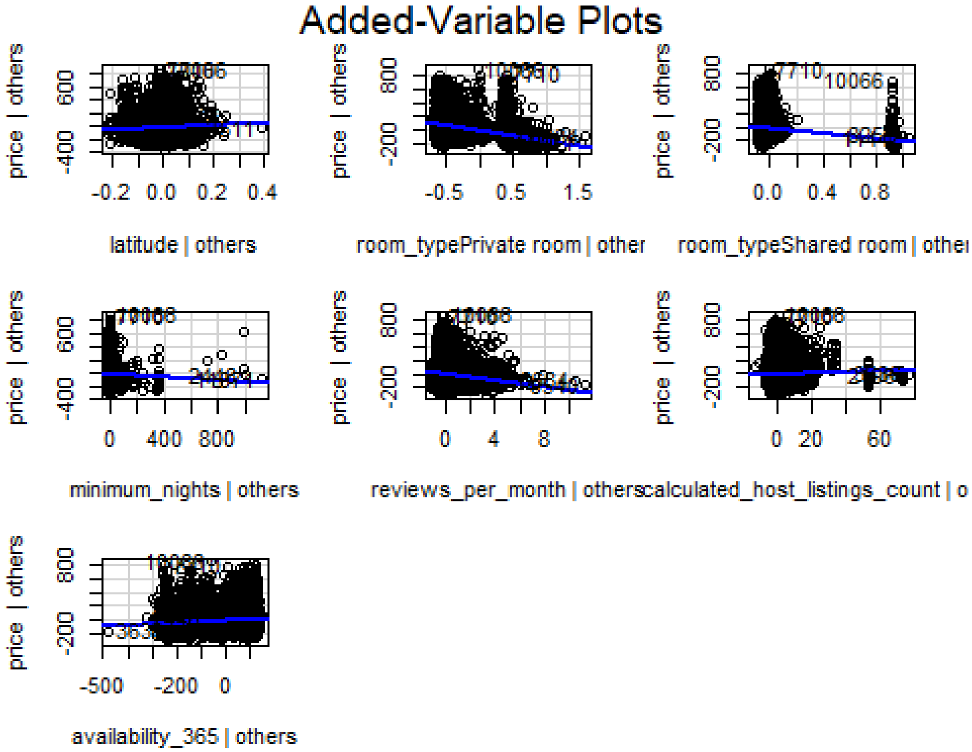


```
## [1]  7710 10066
```

```
leveragePlots(Istanbul_m3) # leverage plots
```



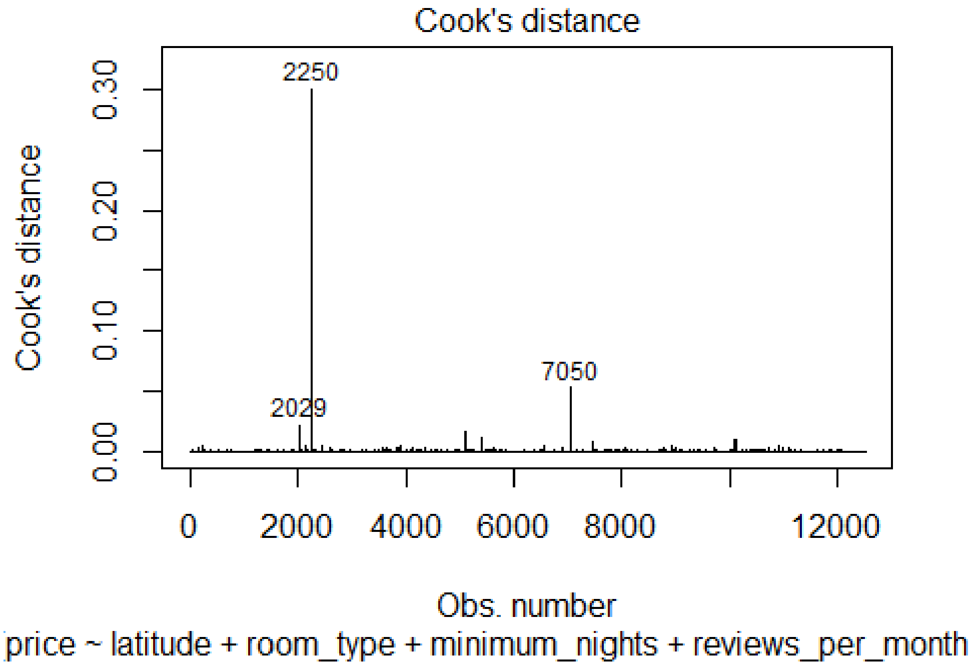
```r
# Influential Observations
# added variable plots
avPlots(Istanbul_m3)
```



```
#The above plots give the row numbers of some influential observations
.
```

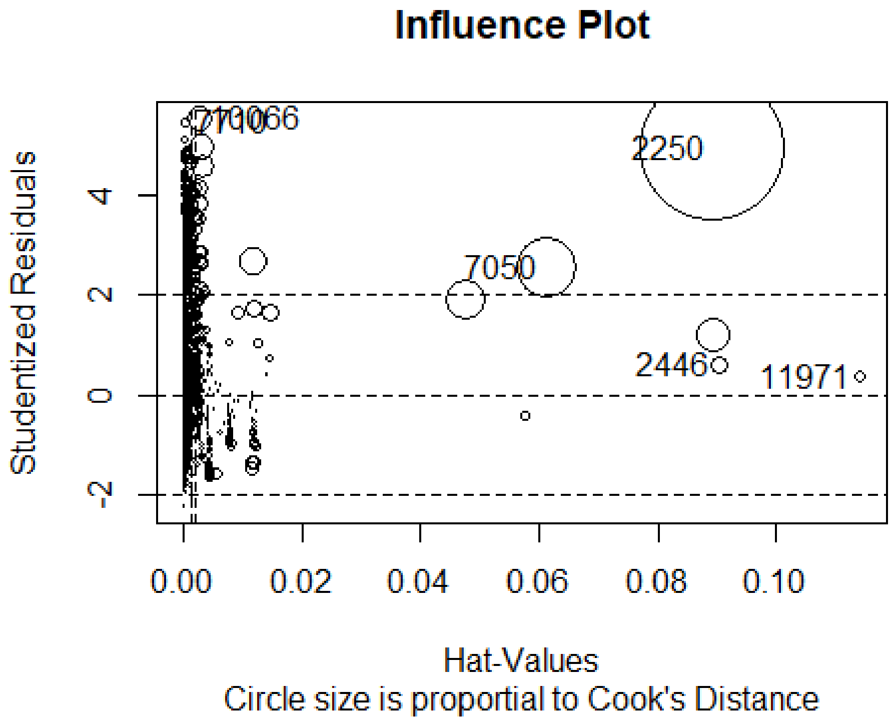
# Cook's D plot to find out the data points which strongly influences the fitted values.

```
#Bar Plot of Cook's distance to detect observations that strongly infl
uence fitted values of the model. Cook's distance was introduced by Am
erican statistician R Dennis Cook in 1977. It is used to identify infl
uential data points. It depends on both the residual and leverage i.e
it takes it account both the x value and y value of the observation.
#A data point having a large cook's d indicates that the data point st
rongly influences the fitted values.
# identify D values > 4/(n-k-1)
#cooks dist formula below
cutoff <- 4/((nrow(training_Istanbul)-length(Istanbul_m3$coefficients)-2))
plot(Istanbul_m3, which=4, cook.levels=cutoff)
```



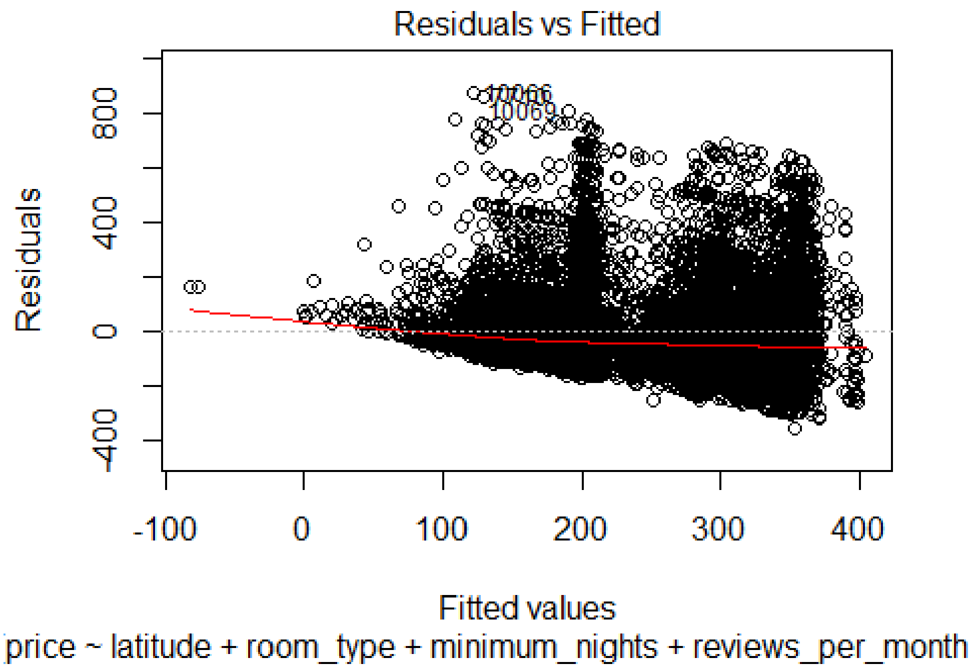
# Representation of above data using Influence Plot

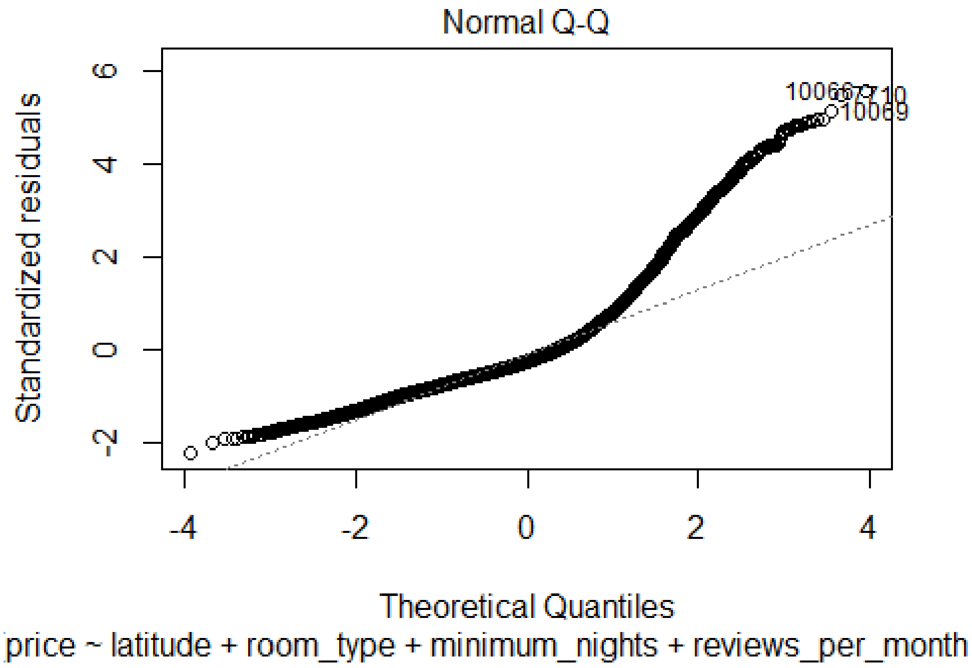
```
influencePlot(Istanbul_m3, id.method="identify", main="Influence Plot", sub="
Circle size is proportial to Cook's Distance" )

## Warning in plot.window(...): "id.method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "id.method" is not a graphical paramete
r

## Warning in axis(side = side, at = at, labels = labels, ...): "id.method" i
s not
## a graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "id.method" i
s not
## a graphical parameter

## Warning in box(...): "id.method" is not a graphical parameter

## Warning in title(...): "id.method" is not a graphical parameter

## Warning in plot.xy(xy.coords(x, y), type = type, ...): "id.method" is not
a
## graphical parameter
```

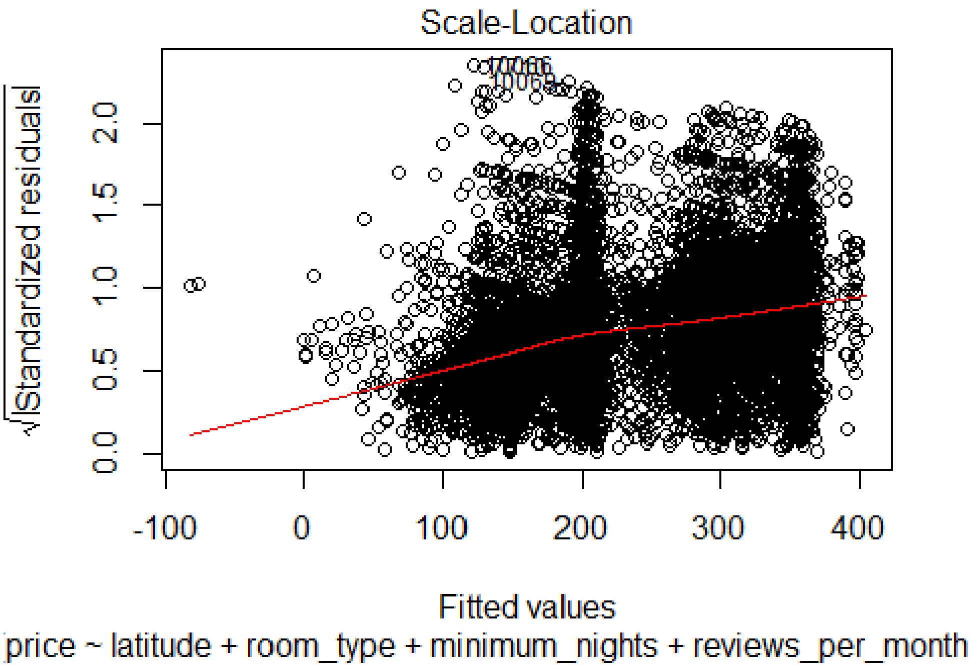


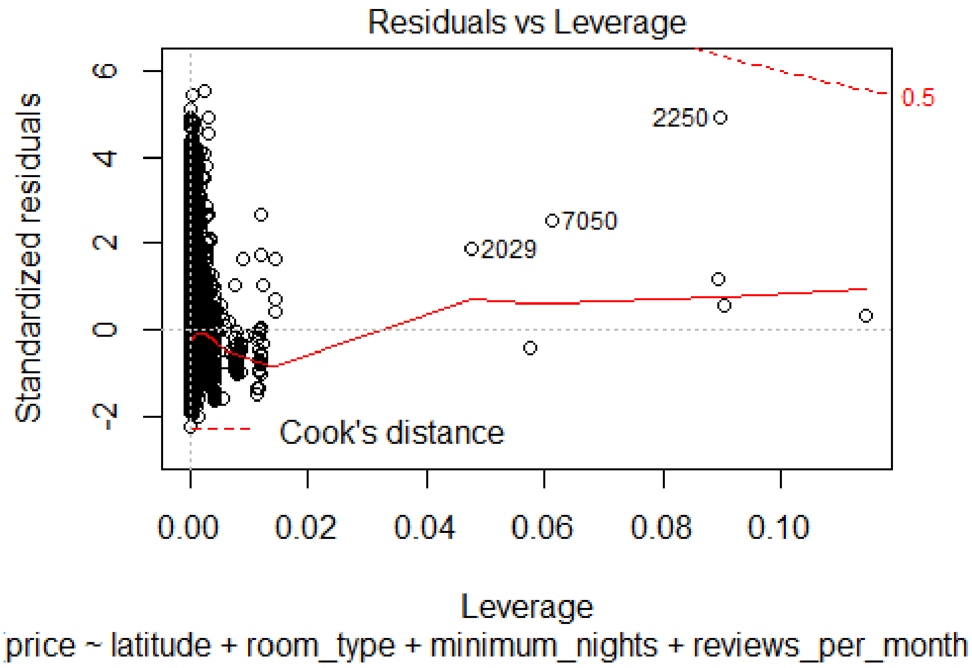
```
##          StudRes          Hat       CookD
## 2250  4.9423907 0.0894948505 0.299561614
## 2446  0.5843878 0.0903865147 0.004242110
## 7050  2.5407198 0.0614570177 0.052814270
## 7710  5.4549390 0.0004247059 0.001576754
## 10066 5.5384587 0.0026279940 0.010079168
## 11971 0.3456914 0.1141843309 0.001925666
```

```r
#bigger circles here means more cooks dist -- thats because oy x or y outliers
#These points negatively influence our model results.
######################################################################
# Normality of Residuals

#Plotting residuals and fitted values
plot(Istanbul_m3)
```


Residuals vs Fitted
Residuals
800
400
0
-400
10066
10069
-100
0
100
200
300
400
Fitted values
price ~ latitude + room_type + minimum_nights + reviews_per_month

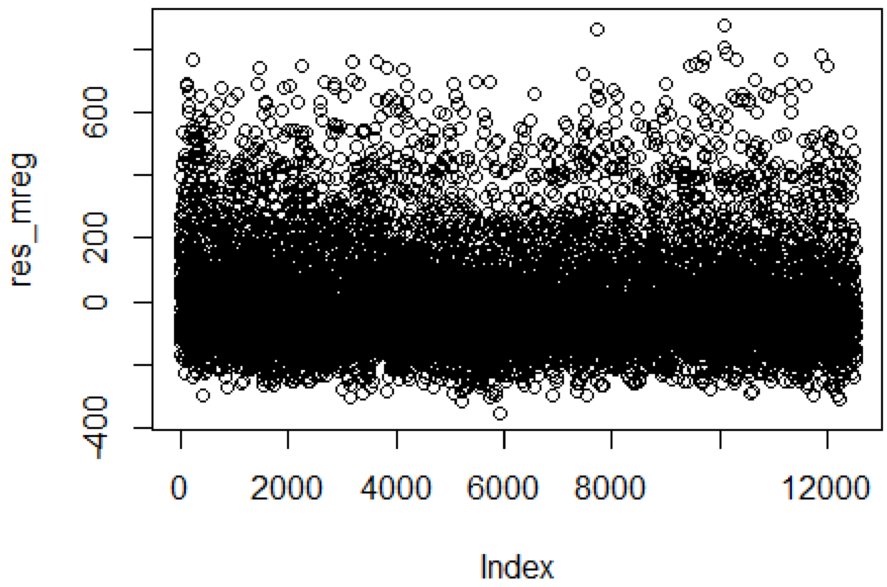

Normal Q-Q
Standardized residuals
6
4
2
0
-2
10066
7710
10069
-4
-2
0
2
4
Theoretical Quantiles
price ~ latitude + room_type + minimum_nights + reviews_per_month



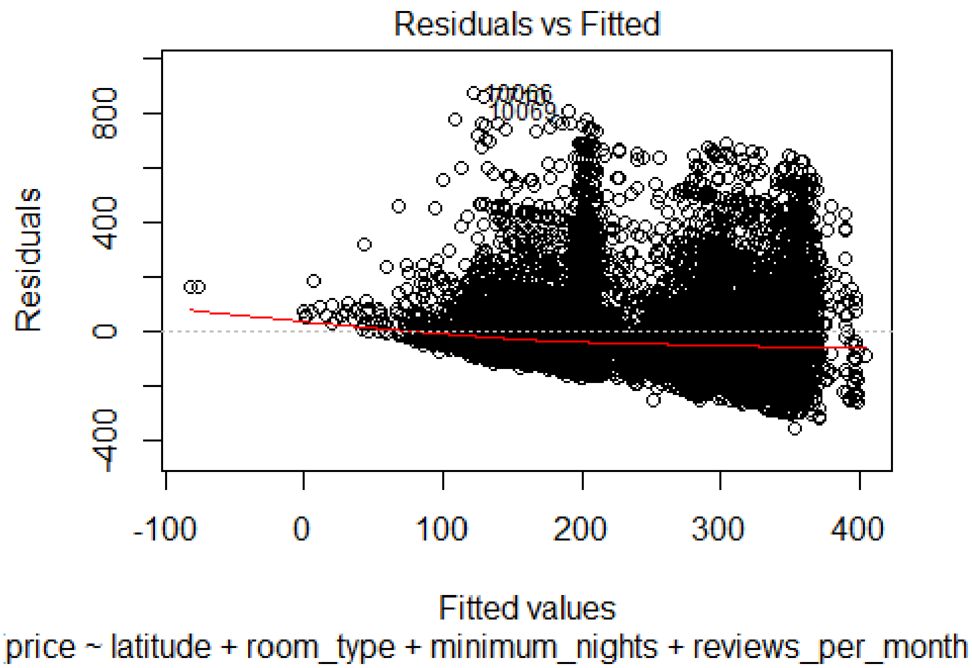


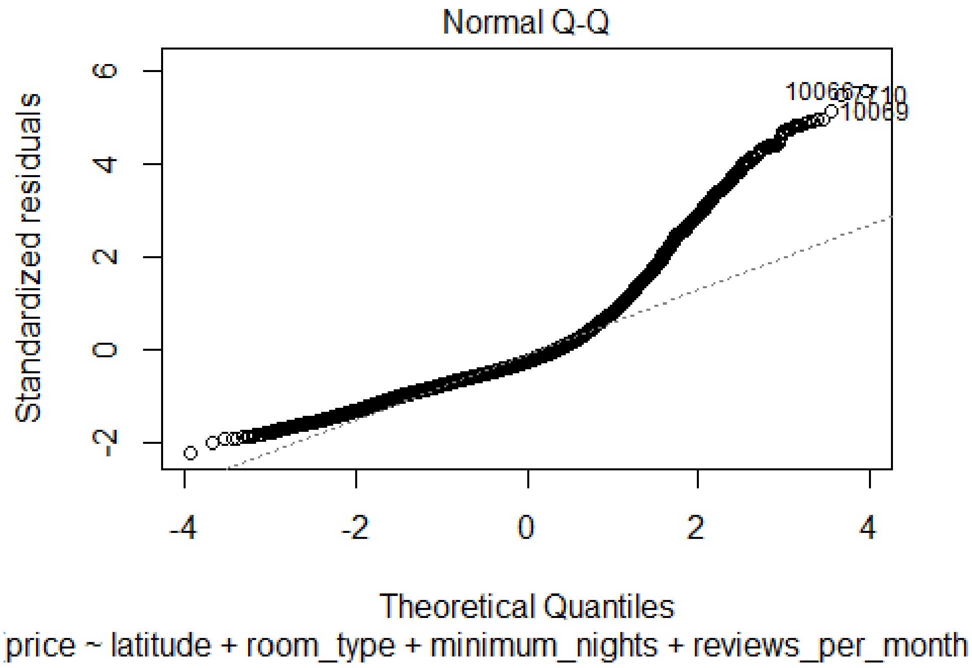
```
#Plotting Residuals
```

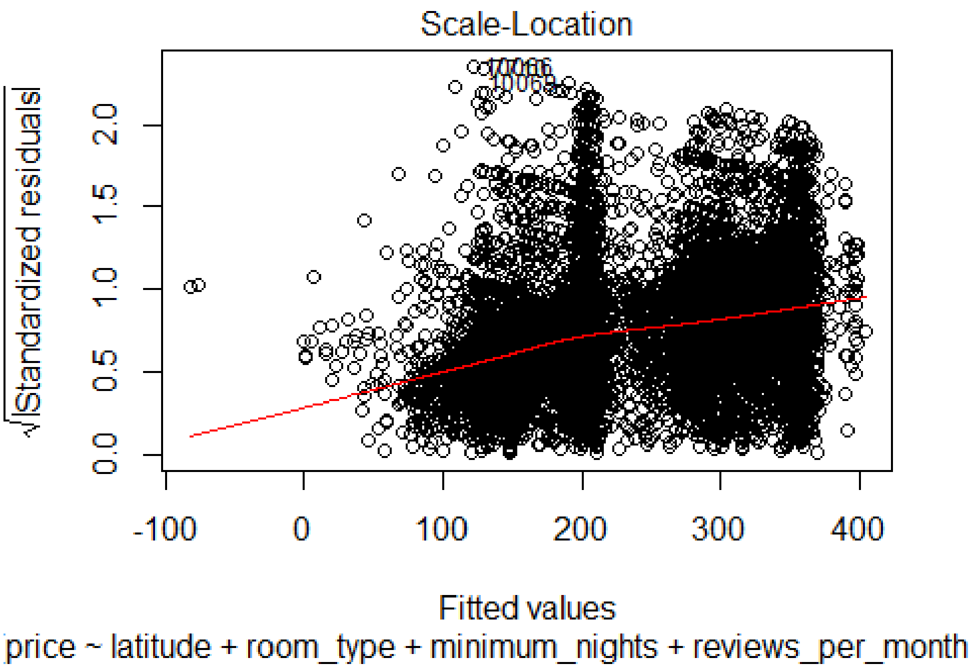
```
res_mreg <- Istanbul_m3$residuals
plot(res_mreg)
```

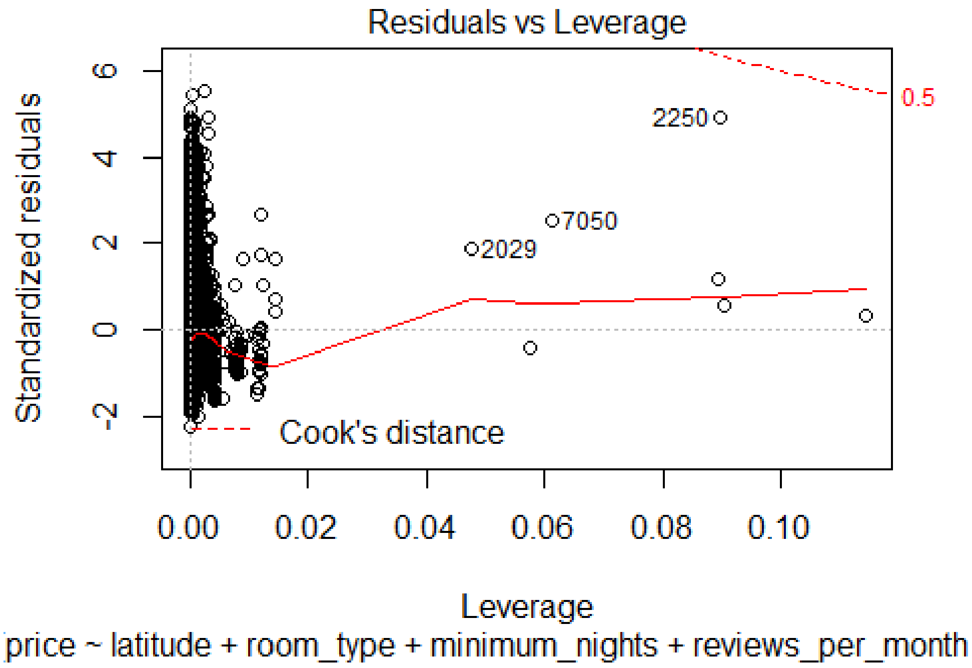


```
#There is no pattern in residuals
plot(Istanbul_m3)
```


Residuals vs Fitted
Residuals
800
400
0
-400
10066
10069
-100
0
100
200
300
400
Fitted values
price ~ latitude + room_type + minimum_nights + reviews_per_month


Normal Q-Q
Standardized residuals
6
4
2
0
-2
10066
17710
10069
-4
-2
0
2
4
Theoretical Quantiles
price ~ latitude + room_type + minimum_nights + reviews_per_month



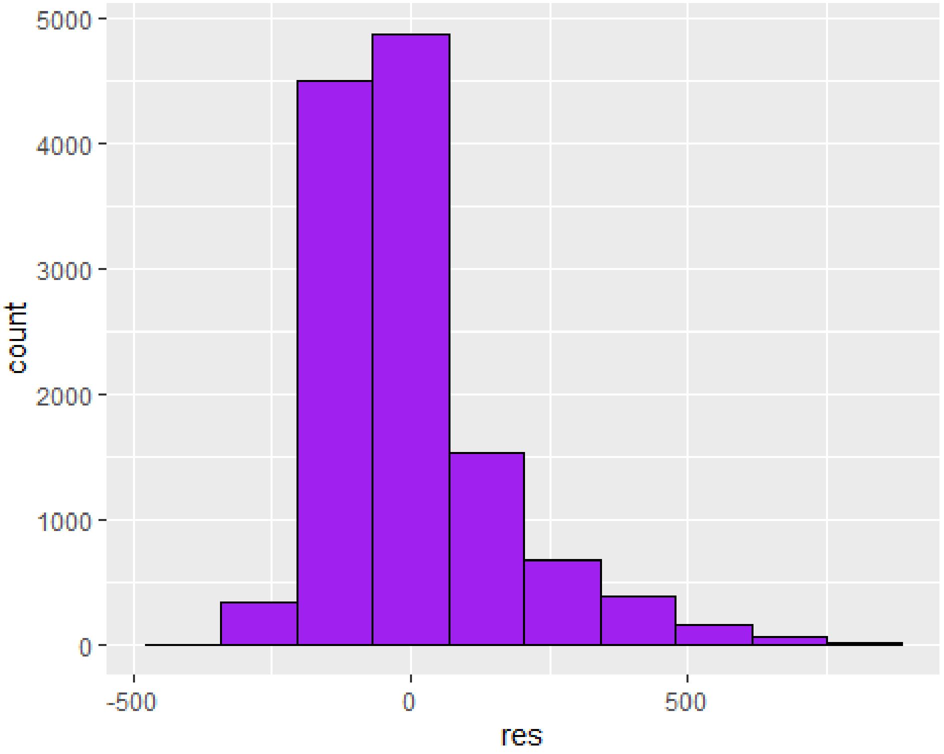


```
summary(res_mreg)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
## -353.24  -97.58  -41.88    0.00   51.75  874.35
```

```
#Mean of residuals = zero
```

### plotting residuals histogram

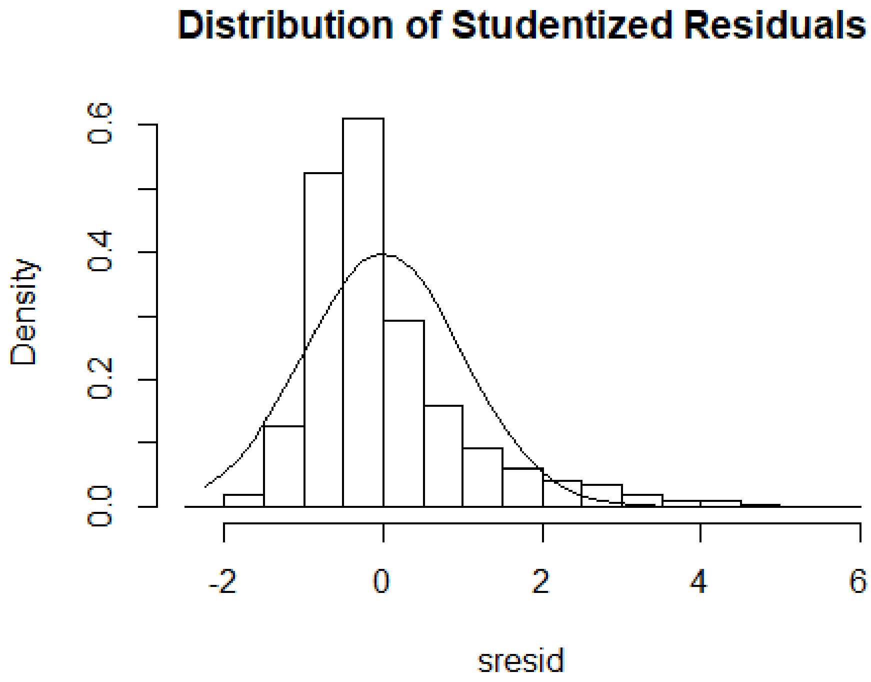
```
resdf = data.table('res'=Istanbul_m3$residuals)
ggplot(resdf ,aes(x=res)) + geom_histogram(bins=10,fill ='purple',color='blac
k')
```



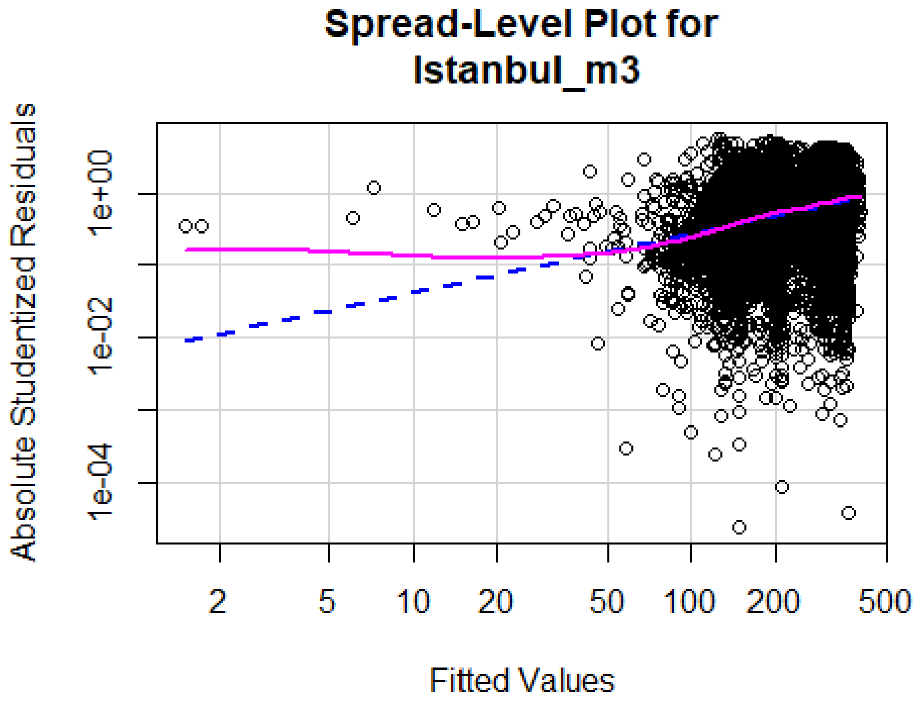
# distribution of studentized residuals

```
library(MASS)

##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select

## The following objects are masked from 'package:fma':
##
##     cement, housing, petrol

sresid <- studres(Istanbul_m3)
hist(sresid, freq=FALSE,
     main="Distribution of Studentized Residuals")
xfit<-seq(min(sresid),max(sresid),length=40)
```

```
yfit<-dnorm(xfit)
lines(xfit, yfit)
```



```
#Above histograms shows that residuals plotted are normally
distributed
#So our model Istanbul_m3 is good
#/////
# Evaluate homoscedasticity
# non-constant error variance test
ncvTest(Istanbul_m3)

## Non-constant Variance Score Test
## Variance formula: ~ fitted.values
## Chisquare = 685.7009, Df = 1, p = < 2.22e-16

# plot studentized residuals vs. fitted values
spreadLevelPlot(Istanbul_m3)

## Warning in spreadLevelPlot.lm(Istanbul_m3):
## 3 negative fitted values removed
```



```
##
## Suggested power transformation:  0.1815572
```

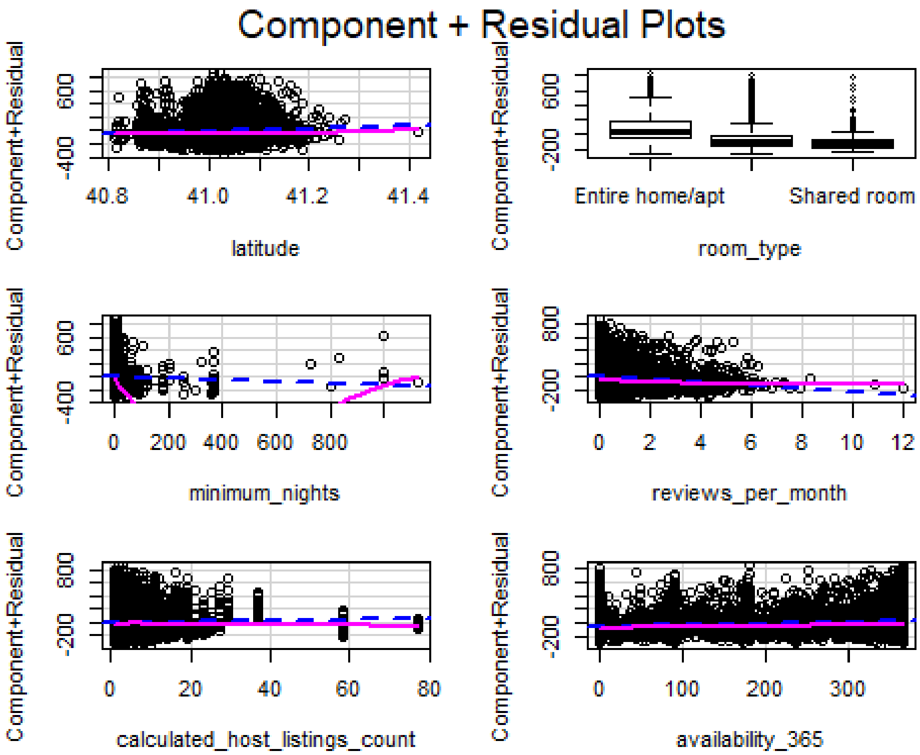
```
#Multi-collinearity
# Evaluate Collinearity
vif(Istanbul_m3) # variance inflation factors
```

```
##                                 GVIF Df GVIF^(1/(2*Df))
## latitude                    1.002653  1        1.001326
## room_type                   1.057827  2        1.014154
## minimum_nights              1.005693  1        1.002843
## reviews_per_month           1.058138  1        1.028658
## calculated_host_listings_count 1.044078  1        1.021802
## availability_365            1.030614  1        1.015192
```

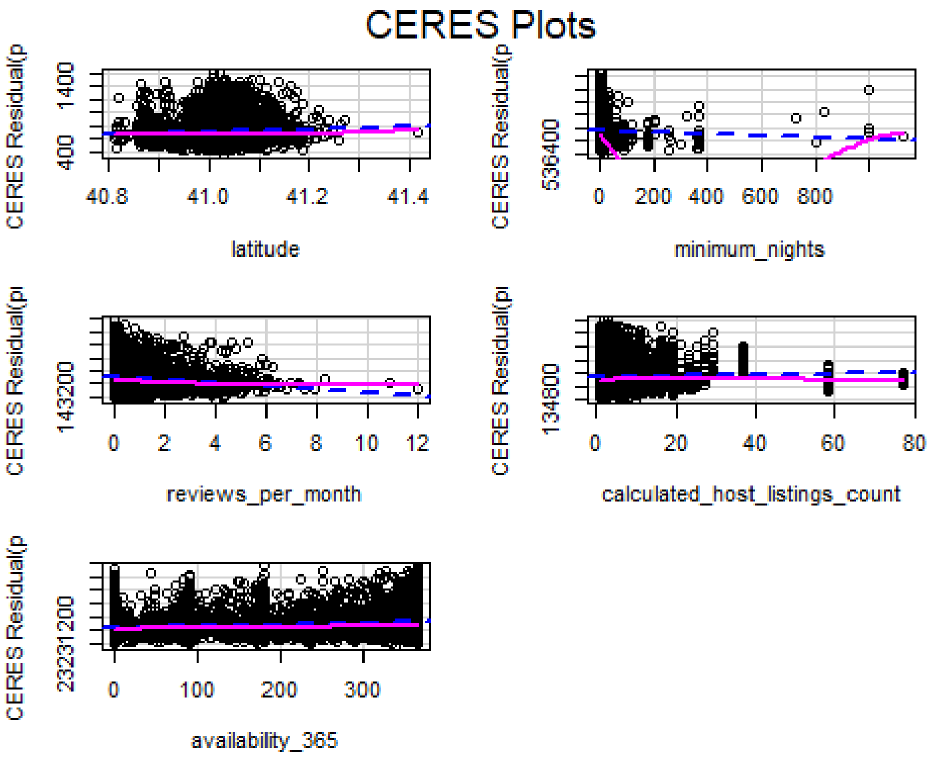
```
sqrt(vif(Istanbul_m3)) > 2 # problem?
```

```
##                                 GVIF    Df GVIF^(1/(2*Df))
## latitude                       FALSE FALSE           FALSE
## room_type                      FALSE FALSE           FALSE
## minimum_nights                 FALSE FALSE           FALSE
## reviews_per_month              FALSE FALSE           FALSE
## calculated_host_listings_count FALSE FALSE           FALSE
## availability_365               FALSE FALSE           FALSE
```

```
#Nonlinearity use below crplots or ceresplots to find out nonlinearity
# component + residual plot
crPlots(Istanbul_m3)
```



```
# Ceres plots
ceresPlots(Istanbul_m3)
```



```
#///////

###
```

# Global test of model assumptions

```
#install.packages("gvlma", lib="/Library/Frameworks/R.framework/Versions/3.5/
Resources/library")
library(gvlma) #gives more thn lm basic regression is same
gvmodel <- gvlma(Istanbul_m3)
summary(gvmodel)

##
## Call:
## lm(formula = price ~ latitude + room_type + minimum_nights +
##     reviews_per_month + calculated_host_listings_count + availability_365,
##     data = training_Istanbul)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -353.24  -97.58  -41.88   51.75  874.35
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   -6.537e+03  1.370e+03  -4.772 1.84e-06 ***
## latitude                       1.662e+02  3.338e+01   4.978 6.51e-07 ***
## room_typePrivate room         -1.521e+02  2.966e+00 -51.268  < 2e-16 ***
## room_typeShared room          -1.935e+02  8.412e+00 -23.003  < 2e-16 ***
## minimum_nights                -1.229e-01  4.759e-02  -2.582 0.009845 **
## reviews_per_month             -2.465e+01  1.659e+00 -14.859  < 2e-16 ***
## calculated_host_listings_count  7.312e-01  1.907e-01   3.834 0.000127 ***
## availability_365               1.995e-01  1.051e-02  18.984  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 158.3 on 12493 degrees of freedom
## Multiple R-squared:  0.209,  Adjusted R-squared:  0.2086
## F-statistic: 471.6 on 7 and 12493 DF,  p-value: < 2.2e-16
##
##
## ASSESSMENT OF THE LINEAR MODEL ASSUMPTIONS
## USING THE GLOBAL TEST ON 4 DEGREES-OF-FREEDOM:
## Level of Significance =  0.05
##
## Call:
##  gvlma(x = Istanbul_m3)
##
##                        Value   p-value                   Decision
## Global Stat        11509.838 0.000e+00 Assumptions NOT satisfied!
## Skewness            5628.723 0.000e+00 Assumptions NOT satisfied!
## Kurtosis            5834.592 0.000e+00 Assumptions NOT satisfied!
## Link Function          3.698 5.447e-02    Assumptions acceptable.
## Heteroscedasticity    42.825 5.986e-11 Assumptions NOT satisfied!
```

# Predicting the Price for the Test set

```
y_pred = predict(Istanbul_m3, newdata = test_Istanbul)
head(y_pred)
```

```
##        1        2        3        4        5        6
## 286.4991 221.0232 339.6111 348.0378 184.4248 203.8438
```

#Alternatively, forecast function can be used to Predict

```
library(forecast)
head(fitted(Istanbul_m3)) #Printing fitted values
```

```
##        1        2        3        4        5        6
## 359.2821 304.2644 326.6783 364.3025 355.2990 364.9299
```

```
fc = forecast(Istanbul_m3,h=30,newdata = test_Istanbul)
head(fc)
```

```
## $model
##
## Call:
## lm(formula = price ~ latitude + room_type + minimum_nights +
##     reviews_per_month + calculated_host_listings_count + availability_365,
##     data = training_Istanbul)
##
## Coefficients:
##                    (Intercept)                        latitude
##                     -6537.0332                        166.1887
##          room_typePrivate room             room_typeShared room
##                      -152.0769                       -193.4954
##                 minimum_nights                reviews_per_month
##                        -0.1229                        -24.6461
## calculated_host_listings_count                 availability_365
##                         0.7312                          0.1995
##
##
## $mean
##          1          2          3          4          5          6
7
## 286.499132 221.023186 339.611118 348.037785 184.424785 203.843790 281.9619
03
##          8          9         10         11         12         13
14
## 350.044111 355.593201 323.288551 205.784763 353.635557 282.122882 168.6586
```

```
17
##         15         16         17         18         19         20         
21 
## 200.787466 200.475538 206.852143 354.066039  85.651978 341.516144 302.1330
58 
##         22         23         24         25         26         27         
28 
## 206.680833 353.523163 312.649087 289.137331 198.213966 203.727346 341.6057
67 
##         29         30         31         32         33         34         
35 
## 203.584423 147.465484 341.080430 205.819662 312.459113 208.000058 278.9406
64 
##         36         37         38         39         40         41         
42 
## 322.650375 317.781692 344.318842 206.808095 200.948896 197.985103 329.9688
96 
##         43         44         45         46         47         48         
49 
## 354.566423 302.454453 280.695746 204.247741 203.810552 226.208303 345.0823
03 
##         50         51         52         53         54         55         
56 
## 279.970779 335.640241 356.515548 221.340028 158.230301 204.069695 296.2320
68 
##         57         58         59         60         61         62         
63 
## 198.641970 342.557273 202.600586 298.347178 295.090674 358.357144 212.9692
68 
##         64         65         66         67         68         69         
70 
## 344.634197 347.159589 362.657884 353.318615 331.786130 187.198707 207.9961
62 
##         71         72         73         74         75         76         
77 
## 352.221710 326.894411 194.575333 299.055664 304.416188 204.350164 205.1737
71 
##         78         79         80         81         82         83         
84 
## 202.502534 198.736697 174.143104 208.977921 207.293695 268.220802 329.6106
20 
##         85         86         87         88         89         90         
91 
## 205.493932 320.803122 357.429586 264.747778 349.716753 202.523292 191.6626
04 
##         92         93         94         95         96         97         
98 
## 325.544191 270.129153 186.773500 342.981609 203.652673 332.660669 195.3334
96 
##         99        100        101        102        103        104          1
```

```
05
## 317.758362 335.113763 197.422040 179.783811 356.103400 355.774660 179.4592
42
##        106        107        108        109        110        111        1
12
## 356.545342 353.500508 359.141556 356.182984 274.120150 151.117318 203.4547
96
##        113        114        115        116        117        118        1
19
## 144.750908 357.272089 204.910609 356.862381 207.319567 356.018784 335.0978
25
##        120        121        122        123        124        125        1
26
## 323.651356 348.095552 286.035948 353.200268 198.268045 192.564840 292.7140
15
##        127        128        129        130        131        132        1
33
## 203.930208 365.797964 324.981377 201.429202 182.645230 293.659082 205.4390
89 ## 1067 442.3187 550.1287
## 1068 336.0713 443.4771
## 1069 483.3622 590.7846
## 1070 413.5705 521.3789
## 1071 568.1051 675.5187
## 1072 384.6077 492.1917
## 1073 550.4134 657.8214
## 1074 331.3395 438.7498
## 1075 401.9091 509.3104
## 1076 569.1521 676.5644
## 1077 387.8368 495.2630
## 1078 336.7935 444.2055
## 1079 287.6033 395.1443
## 1080 435.6279 543.1344
## 1081 486.2064 593.6210
## 1082 335.5606 442.9718
## 1083 527.0611 634.4704
## 1084 485.9652 593.3734
## 1085 538.1128 645.5162
## 1086 395.5258 502.9341
## 1087 335.6970 443.1083
## 1088 484.1322 591.5468
## 1089 413.0996 520.5058
## 1090 550.8782 658.2865
## 1091 327.3118 434.7242
## 1092 488.7101 596.1265
## 1093 505.5954 613.0135
## 1094 549.8313 657.2783
## 1095 495.6659 603.0753
## 1096 549.5143 656.9624
## 1097 335.8087 443.2127
## 1098 561.4459 668.8522
## 1099 535.6348 643.0871
## 1100 363.8371 471.2513
## 1101 558.8287 666.2344
## 1102 451.7238 559.1542
## 1103 501.7517 609.1581
## 1104 335.8816 443.2931
## 1105 336.5938 444.1393
## 1106 371.8345 479.3707
## 1107 357.2188 464.7560
## 1108 520.2985 627.7036
## 1109 472.5748 580.0126
## 1110 326.9055 434.3185
## 1111 400.8186 508.2205
## 1112 564.1388 671.5468
## 1113 400.0479 507.4473
## 1114 324.9008 432.3161
## 1115 487.8924 595.3073
## 1116 503.0327 610.4473
## 1117 489.5831 596.9994
## 1118 555.9910 663.3965
## 1119 398.1548 505.5588
## 1120 398.3864 505.7905
## 1121 469.7810 577.1978
## 1122 405.3408 512.7412
## 1123 370.3947 477.7945
## 1124 294.6266 402.0965
## 1125 339.9622 447.3774
## 1126 539.7650 647.1978
## 1127 514.5973 622.0895
## 1128 497.7053 605.1134
## 1129 491.2857 598.6964
## 1130 324.3769 431.7930
## 1131 560.5102 667.9139
## 1132 333.9207 441.3313
## 1133 414.5220 521.9273
## 1134 400.9001 508.3021
## 1135 403.7758 511.1763
## 1136 468.5966 576.0058
## 1137 562.0009 669.4116
## 1138 409.6355 517.0378
## 1139 403.4352 510.8359
## 1140 327.8165 435.2288
## 1141 329.8133 437.2234
## 1142 333.2726 440.6831
## 1143 360.5446 467.9609
## 1144 353.0429 460.4469
## 1145 355.9819 463.3860
## 1146 397.0487 504.4523
## 1147 555.2892 662.6915
## 1148 403.1243 510.5249
## 1149 368.4018 475.8011
## 1150 567.7304 675.1443
## 1151 367.6247 475.0309
## 1152 335.6864 443.1288
## 1153 427.4931 534.9379
## 1154 549.6719 657.0751
## 1155 327.9535 435.3658
## 1156 419.5722 526.9940
## 1157 528.0697 635.4701
## 1158 560.5435 667.9496
## 1159 558.0779 665.4827
## 1160 336.8918 444.2967
## 1161 549.7344 657.1376
## 1162 371.2282 478.6271
## 1163 521.2322 628.6342
## 1164 290.9106 398.4508
## 1165 387.6493 495.0473
## 1166 551.3725 658.7820
## 1167 400.8607 508.2609
## 1168 367.5064 474.9280
## 1169 365.4408 472.9729
## 1170 547.2310 654.6336
## 1171 539.0059 646.4082
## 1172 408.7568 516.1585
## 1173 381.8329 489.2431
## 1174 404.7334 512.1416
## 1175 529.0223 636.4548
## 1176 430.4280 537.8879
## 1177 405.0935 51253 121
## [11737] 248 137 211 395 200 200 601 627 237 243 596 596 190 105  58 343 74
9 211
## [11755] 269 127 174 221 100 248 248 148 148 227  79  79 174 190 417 258 17
9 243
## [11773] 169 422 501 701 232  58 701 243  53 100 316 258 148 137 232 211 10
0 190
## [11791] 448 148 111  58 264  74 274 200 121 132 169 195 385 649  53 685 12
1  53
## [11809] 221  90  53 148 290  79  95  90 179 132 269  74 127 269  90 132 63
3 185
## [11827]  63 301 190 179 174 501 411 100  74  84  53 279 132 100 475  74 44
8  84
## [11845] 448 385  63  58 185 295 295 142  69 158 100  53 596 158 200 127 60
1 385
## [11863] 142  74 200 248 211 258 538 248 306 237  53  79 105 596 601 401 47
5 121
## [11881] 264  58 601 100 981 527 501 111  79 179 501 332 148  79 121 163  9
0 401
## [11899] 248 248 232 179  63  58  58 285 285 295 295 417 359 121 506 100 21
1 232
## [11917]  95  69 301 200 237 105 200 438  90 269  74  53  69 100 179 169 44
```

```
8 401
## [11935] 469 902  79 148 100  53 243 100 111 100  58 221 100  69 448 105  7
9 179
## [11953] 401 200 348 237 369  69 801 448 148 179 353  74 301 153 243 148 12
1 369
## [11971]  53 169 148 100  90  63 169 132 227 100  95 121 359 316 359 949 57
5 195
## [11989] 401  95 200  53  53  53  79 237 179 200 121 359 427 316 348 179  8
4  90
## [12007]  90 817 264 127 200  53 348  95 717 248  53 148  69 211 264 501 23
2 200
## [12025] 200 200 301  53 153 221 105 722 506 359 174 248  53  58 248 132 13
7 200
## [12043] 248  53  58 401 269 348 264 100 158 891 211 200 301 200 248 327  5
3 158
## [12061]  53 401 179 185 717 295 274  90 601 142  90 127  69  84 100 501 39
5  53
## [12079] 237  53 105 237  53 100 248 380  84 179 169 100 200  90 174  58 20
0  74
## [12097] 448 274 142 169 148 153 100 743 248 111 596  53 227 337 179 248 14
2 148
## [12115]  79 121 142 132  53  53 148  84 179 132 422 132 121  58 475 100 24
8 469
## [12133] 232  74 295 801 142 169  69 701 301 100 200 169 264 132 522 148 25
8 179
## [12151] 169  79 169 169 185 227 132  79  63  69 100  63 169  90 243 200 10
0  95
## [12169] 206  69 301  53 290 211  63  58  53 211 211 264 248 148  84  58 36
9 121
## [12187] 237 169 237 237  63 232 401 132  90  53  53  63 200  58  90  79  6
3 654
## [12205] 564 685  79 327 385 448 327  53 332 285 443 385 559 490  53 290 19
0 633
## [12223] 195 327  69 580 701 248 316 316 100 105 100  53  53  69 163 100  7
9 232
## [12241] 190 121 105 501 237  84  95 121 200 121  84 506 475 290 475 316 13
2 148
## [12259]  53 237 200  74 116 401 237  63 111 401  53 111 301  53 243  58 25
8  63
## [12277]  74 527 295 417 269 295 359 417 411 100  74 158 211 506 475 100 31
6 232
## [12295] 105  74 269 200 301 501 142 211 148 148 100  84 417 200 285 274 17
4 174
## [12313] 127 264 116 264 274 279 211 243 190 422 179 132 301 527 316 316 14
8 243
## [12331] 685  84 274  69 179 121 448 121 200 185 148  74 148  69 596 301 22
7 148
## [12349] 248  74 227 232  90 148 527 264 237 211 200 269  74 200 148 132  9
0 211
## [12367] 301 195 200 148 348 348  58 105  74 200 327 258  53 111 369  90  6
```

```
3 301
## [12385] 211  74 121  74 148 253 401 401 232 717 211 237 533 701 243 158 79
1 211
## [12403]  53 132  79  90 179 295 269  69 248 237 211 148 132 174 148 596 17
4  69
## [12421] 169 221 200 100 132 475 200 200 301 200  95 121  74  84 148 580 15
3 211
## [12439] 163 248 232 685 448 211 200 322  79 100 121  79 353 353 237 158 14
2 279
## [12457] 111 316 100 448 738 527  53  74 148 290 838 322 301 216 591 295 15
8 211
## [12475] 148  90 401 322 105 227  53 548 169 301 116 185 121 142 121 316 19
0 221
## [12493] 100  79 264 195 148  53 248 237  53
```

#Printing accuracy

```
accuracy(f=fc,x=test_Istanbul,test=NULL,d=NULL,D=NULL)

##                         ME     RMSE      MAE  MPE MAPE      MASE
## Training set -8.054277e-15 158.2136 114.3778 -Inf  Inf 0.8489685
## Test set     -7.163759e-01 157.3863 114.4079 -Inf  Inf 0.8491923

#===============================
```

# Calculate Relative Importance for Each Predictor

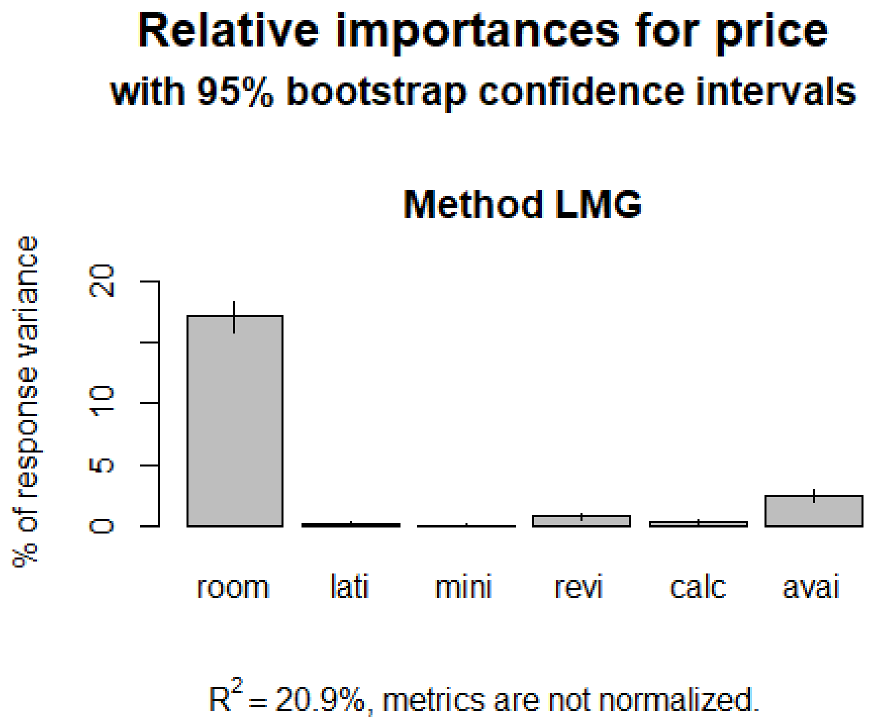
```
#install.packages("relaimpo", lib="/Library/Frameworks/R.framework/Versions/3
.5/Resources/library")
library(relaimpo)

## Warning: package 'relaimpo' was built under R version 3.6.3

## Loading required package: boot

##
## Attaching package: 'boot'

## The following object is masked from 'package:car':
##
##     logit

## The following object is masked from 'package:psych':
##
##     logit

## Loading required package: survey

## Loading required package: grid

## Loading required package: Matrix
```

```
##
## Attaching package: 'Matrix'

## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack

## Loading required package: survival

##
## Attaching package: 'survival'

## The following object is masked from 'package:boot':
##
##     aml

##
## Attaching package: 'survey'

## The following object is masked from 'package:graphics':
##
##     dotchart

## Loading required package: mitools

## This is the global version of package relaimpo.

## If you are a non-US user, a version with the interesting additional metric
pmvd is available

## from Ulrike Groempings web site at prof.beuth-hochschule.de/groemping.
```

```
calc.relimp(Istanbul_m3)
```

```
## Response variable: price
## Total response variance: 31648.55
## Analysis based on 12501 observations
##
## 7 Regressors:
## Some regressors combined in groups:
##         Group  room_type : room_typePrivate room room_typeShared room
##
##  Relative importance of 6 (groups of) regressors assessed:
##  room_type latitude minimum_nights reviews_per_month calculated_host_listi
ngs_count availability_365
##
## Proportion of variance explained by model: 20.9%
## Metrics are not normalized (rela=FALSE).
##
## Relative importance metrics:
##
##                                         lmg
## room_type                      0.1715556366
## latitude                       0.0022928047
## minimum_nights                 0.0001995125
## reviews_per_month              0.0074695423
## calculated_host_listings_count 0.0034771753
## availability_365               0.0240196717
## 
## Average coefficients for different model sizes: 
## 
##                                       1group       2groups       3groups
## latitude                        225.33785729  216.87715626  206.66043701
## room_typePrivate room          -143.82011623 -145.33668458 -146.93514582
## room_typeShared room           -185.74656204 -187.06638068 -188.51823336
## minimum_nights                    0.04988151    0.02133911   -0.01030672
## reviews_per_month                -5.76480185   -9.71717514  -13.57633987
## calculated_host_listings_count    1.89355040    1.63821156    1.39359377
## availability_365                  0.20790240    0.20528333    0.20326464
##                                      4groups       5groups      6groups
## latitude                        194.76228161  181.25113886  166.1887369
## room_typePrivate room          -148.60119368 -150.32013794 -152.0769151
## room_typeShared room           -190.08530429 -191.75028802 -193.4954004
## minimum_nights                   -0.04497915   -0.08255552   -0.1228673
## reviews_per_month               -17.34724493  -21.03524467  -24.6461109
## calculated_host_listings_count    1.16040755    0.93937454    0.7312308
## availability_365                  0.20171780    0.20051355    0.1995219
```

```r
# Bootstrap Measures of Relative Importance (1000 samples)
bootres <- boot.relimp(Istanbul_m3, b = 1000)
rel_imp<-booteval.relimp(bootres) # print result
plot(rel_imp) # plot result
```



#As per above plot, room type plays important role, followed by availability

#Predicting the Price value with our model giving one observation values as input

```
#The actual Price value = 100 for this observation

predict.lm(Istanbul_m3,data.frame(latitude=40.99467,room_type="Private room",
                                  minimum_nights=1,number_of_reviews=0,review
s_per_month=0,
                                  calculated_host_listings_count=1,availabili
ty_365=364))

##        1
## 196.9767
```

Conclusion:

#The predicted value is 198.6143 and actual value is 100