INTRODUCTION / DEFINING THE DATA SET

The analyzed data at hand is about properties which includes various apartments/flats, loft, condominium, and guest house to be rented out to tenants in Berlin, Germany. Not only this the data also entails finding a right apartment which is a crucial step, while considering several factors for all the properties in Berlin **i.e** security deposit, how many people that property can accommodate, number of bedrooms, bathrooms, beds, and number of reviews and review scores on cleanliness. The data contains in depth relevant information of hosts including their ids, names, about themselves, current location, and their response rate.

The continuous dependent variable is property in this regression model will be the price that has been listed against a particular housing. as it depends on other independent variables which has already been mentioned above.

When considering a **property**, the first and foremost basic thought that comes in mind is of price and space. As price varies according to the space (which includes number of bedrooms and bathrooms etc. For that reason, looking for a property for rent should be adjusted according to one's budget. Popular or elite areas obviously are expensive as compared to backward areas. Tenants also reflect up on the importance of having a transit stop nearby their property. This would eventually help the tenants to have easy access to different places of the city at affordable rates.

Thus, this means that a tenant when hunting for a property should keep in mind of these factors which well suits his/her needs.

Regression Model

For this regression model I will explain everything step by step.

First, Load the required packages and analyze data set.

cleaning data is also necessary to perform regression. So, for that below mentioned program will be performed.

Before we start our model, we need to download/install some libraries. After installing we can load those libraires with the following command.

```
library(tidyverse)
library(openxlsx)
library(readxl)
library(spdep)
library(fastDummies)
```

```
library(olsrr)
library(modelr)
library(knitr)
```

In order to loading the excel file in RStudio following command is required.

```
data <- read.xlsx("Airbnb_listings.xlsx")
```

Now, in order to proceed with our regression model. We need to select dependent variable and independent variables. After that has been taken care of, we need to prepare our dummy variables.

```
data <- data %>% select(price, beds,host_neighbourhood,
                        latitude, longitude, neighbourhood_group_cleansed)
data <- data %>% mutate(price=gsub("\\$","",price),
                        price=gsub("\\,","",price),
                        price=as.numeric(price))


data <- data %>% na.omit()
data <- data %>% dummy_cols(select_columns="host_neighbourhood")
data <- data %>% na.omit()
```

Just to be sure you can check the results upper commands and rectify any errors.

```
data%>%head
```

Change your data into nominal data and transfer them in the set of du we can reduce our data set to 1000 observations with the help of a single command, which has been mentioned below as dummy variables.

```
set.seed(25)
new.data <- data %>% sample_n(size=1000)
```

Carry out the linear regression explaining the price and estimate the model with independent variables.

```
model <- price~beds + host_neighbourhood_
reg.results <- new.data %>% lm(formula=model)
```

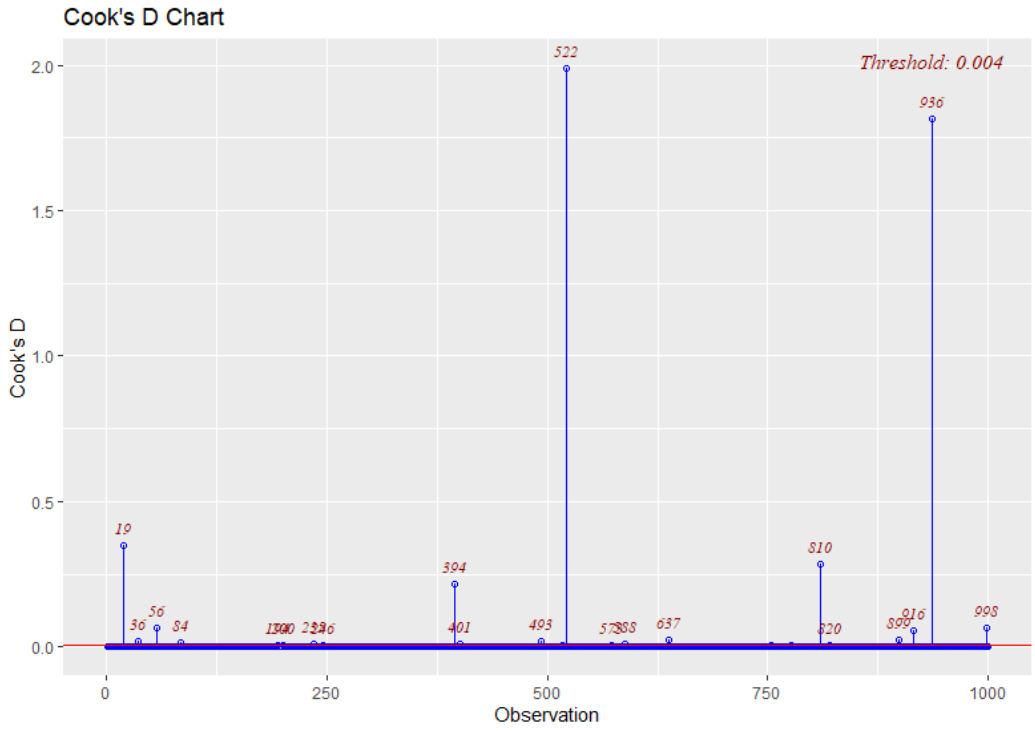
We can check the results with the following formula.

```
reg.results %>% ols_vif_tol()
```

```
Variables Tolerance      VIF
1                beds 0.9892437 1.010873
2 host_neighbourhood_ 0.9892437 1.010873
```

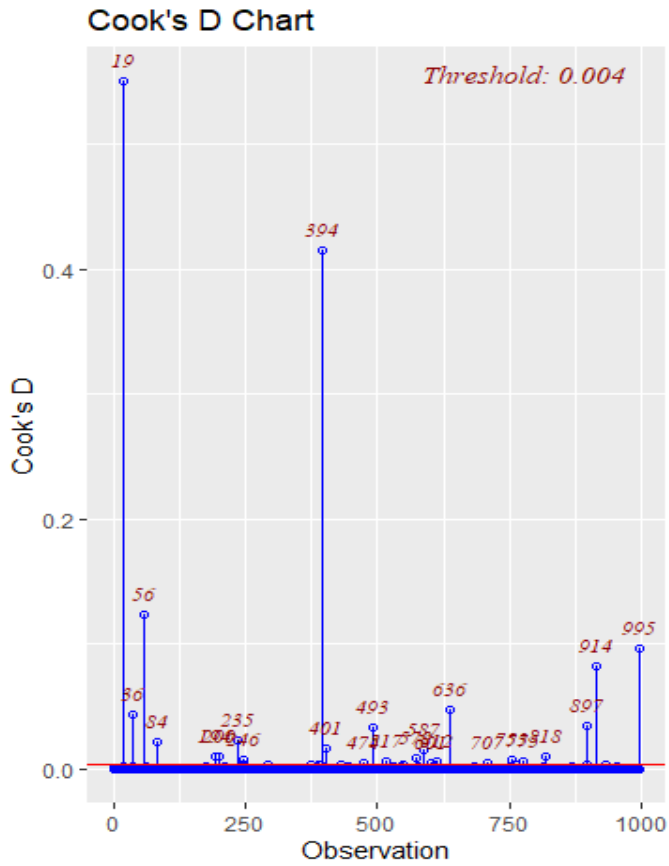
We can also check the graph with the following command.

```
reg.results %>% ols_plot_cooksd_chart()
```



To remove potential outliers use the following command and generate a new graph.

```
new.data <- new.data %>% slice(-522,-810,-936)
reg.results.out <- new.data %>% lm(formula=model)
reg.results.out %>% ols_plot_cooksd_chart()
```



To avoid unexpected errors, we need to change some variables of price from 0 to a bigger digit and for that we can use the following command.

```
new.data <- new.data %>% mutate(price= ifelse(price==0, 1, price))
```

After that we will use Shapiro-Wilk test to check for normal distribution and will perform log with the following formula.

```
model <- log(price)~beds+host_neighbourhood_
reg.results.out <- new.data %>% lm(formula=model)
reg.results.out %>% ols_test_normality()
```

And the result are as follows.

```
-----------------------------------------------
       Test             Statistic       pvalue
-----------------------------------------------
Shapiro-Wilk              0.4488         0.0000
```

```
Kolmogorov-Smirnov        0.2256         0.0000
Cramer-von Mises        114.4353         0.0000
Anderson-Darling        109.5252         0.0000
-----------------------------------------------
```

HETEROSCEDASTICITY

The next thing is test of heteroscedasticity. to test for a constant variance of residuals, the Breusch-Pagan test is used. It suggests that the residuals are not homoscedastic.

The following command can be used to check test summary.

```
reg.results.out %>% ols_test_breusch_pagan()
Breusch Pagan Test for Heteroskedasticity
 ----------------------------------------
 Ho: the variance is constant
 Ha: the variance is not constant

              Data
 ---------------------------------
 Response : price
 Variables: fitted values of price

         Test Summary
 -------------------------------
 DF            =    1
 Chi2          =    156.7679
 Prob > Chi2   =    5.752589e-36
```

```
model <- log(price) ~ beds^2+host_neighbourhood
reg.results.out <- new.data %>% lm(formula=model)
reg.results.out %>% ols_test_breusch_pagan()
```

AUTOCORRELATION:

To test if there are any potential dependencies between the observations the 5 closest observations are used to create spatial weights object.

```
new.data <- new.data %>% add_residuals(reg.results.out, var="Residuals")
coords <- new.data %>% select(longitude, latitude)
coordinates(coords) <- ~ longitude + latitude
neig <-knearneigh(coords,5,longlat = T)
neig <- knn2nb(neig)
neig <- nb2listw(neig,zero.policy=T)

new.data %>% pull(Residuals) %>%
  moran.test(listw=neig, zero.policy = T, na.action = na.omit)
```

```
Moran I test under randomisation
data:  .
weights: neig
Moran I statistic standard deviate = 4.1468, p-value =
1.686e-05
alternative hypothesis: greater
sample estimates:
Moran I statistic       Expectation          Variance
     0.0713471647     -0.0010040161      0.0003044096
```

```
new.data <- new.data %>% mutate(beds.lag=lag.listw(var=beds,x=neig,zero.policy
= T))
model <- log(price)~beds^2+host_neighbourhood+ beds.lag
reg.results.out <- new.data %>% lm(formula=model)
new.data <- new.data %>% add_residuals(reg.results.out, var="Residuals2")
  new.data %>% pull(Residuals2) %>%
    moran.test(listw=neig, zero.policy = T, na.action = na.omit)
```