Project – 01 Report

By Aparna Cleetus (axc190011), Obuli Vignesh Rangasamy (oxr170630)

Linear Regression (Melbourne Housing Dataset)

-- 01 Loading required packages --

```r
#loading packages
require(MASS)
require(ISLR)
require(corrplot)
library(tidyverse)
library(Metrics)
```

```
> require(MASS)
Loading required package: MASS
> require(ISLR)
Loading required package: ISLR
> require(corrplot)
Loading required package: corrplot
corrplot 0.84 loaded
> library(tidyverse)
— Attaching packages ———————————————
✓ ggplot2 3.3.1     ✓ purrr   0.3.4
✓ tibble  3.0.1     ✓ dplyr   1.0.0
✓ tidyr   1.1.0     ✓ stringr 1.4.0
✓ readr   1.3.1     ✓ forcats 0.5.0
— Conflicts ———————————————————
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
x dplyr::select() masks MASS::select()
```

-- 02 Reading and exploring the dataset --

```r
#load data
house_data = read.csv("/Users/obulivignesh/desktop/Melbourne_housing_FULL.csv",
                      stringsAsFactors = FALSE, quote = "")
names(house_data)
head(house_data)
dim(house_data)
```

```
> names(house_data)
 [1] "Suburb"        "Address"       "Rooms"         "Type"          "Price"
     "Method"        "SellerG"       "Date"          "Distance"
[10] "Postcode"      "Bedroom2"      "Bathroom"      "Car"           "Landsize"
     "BuildingArea"  "YearBuilt"     "CouncilArea"   "Lattitude"
[19] "Longtitude"    "Regionname"    "Propertycount"
```

```
> head(house_data)
```

A data.frame: 6 × 21

	Suburb	Address	Rooms	Type	Price	Method	SellerG
	<chr>	<chr>	<chr>	<chr>	<int>	<chr>	<chr>
1	Abbotsford	68 Studley St	2	h	NA	SS	Jellis
2	Abbotsford	85 Turner St	2	h	1480000	S	Biggin
3	Abbotsford	25 Bloomburg St	2	h	1035000	S	Biggin
4	Abbotsford	18/659 Victoria St	3	u	NA	VB	Rounds
5	Abbotsford	5 Charles St	3	h	1465000	SP	Biggin
6	Abbotsford	40 Federation La	3	h	850000	PI	Biggin

```
> dim(house_data)
[1] 34857    21
```

-- 03 Eliminating records that do not have price value –

- We have used is.na to find if there are any missing values and eliminate the rows having the missing values
- We are using as.numeric function to generate a correlation matrix to find the covariance among the different predictor variables

```
#data preparation
#elimite records which donot have house price
which(is.na(house_data))
sum(is.na(house_data))
new_house_data = na.omit(house_data)
dim(new_house_data)
new_house_data$Rooms =  as.numeric(new_house_data$Rooms)
new_house_data$Price =  as.numeric(new_house_data$Price)
new_house_data$Distance = as.numeric(new_house_data$Distance)
new_house_data$Propertycount = as.numeric(new_house_data$Propertycount)
new_house_data$Bathroom = as.numeric(new_house_data$Bathroom)
new_house_data$Car = as.numeric(new_house_data$Car)
new_house_data$Landsize = as.numeric(new_house_data$Landsize)
new_house_data$Longtitude = as.numeric(new_house_data$Longtitude)
new_house_data$BuildingArea = as.numeric(new_house_data$BuildingArea)
```

```
> dim(new_house_data)
[1] 8887   21
```

-- 04 Exploratory data analysis --

```
#exploratory data analysis
summary(new_house_data)
```

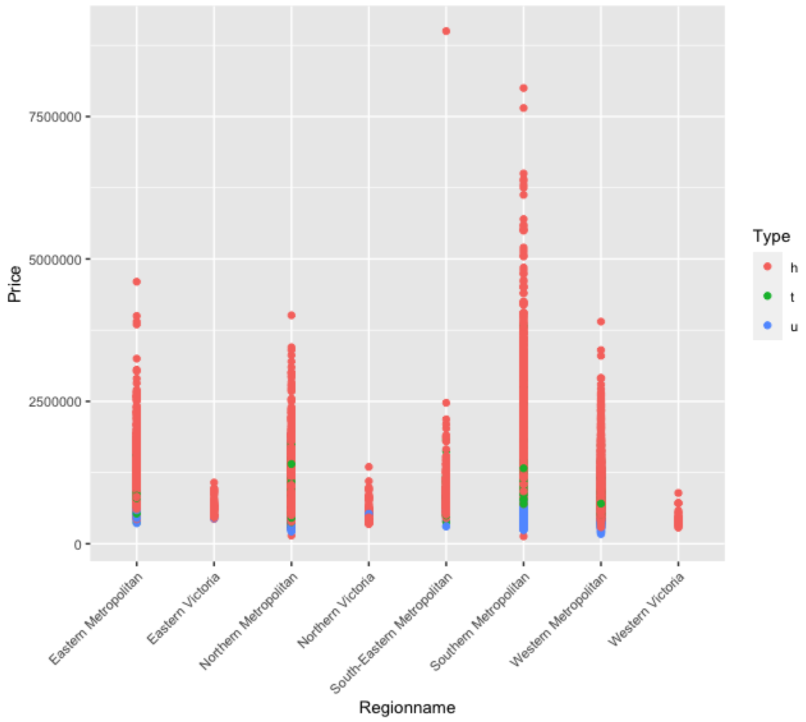
- From summary we see that the datasets have both quantitative and qualitative predictors
- The summary of each of these is listed below

```
> summary(new_house_data)
    Suburb             Address              Rooms             Type               Price             Method             SellerG
 Length:8887        Length:8887        Min.   : 1.000   Length:8887        Min.   : 131000   Length:8887        Length:8887
 Class :character   Class :character   1st Qu.: 2.000   Class :character   1st Qu.: 641000   Class :character   Class :character
 Mode  :character   Mode  :character   Median : 3.000   Mode  :character   Median : 900000   Mode  :character   Mode  :character
                                       Mean   : 3.099                      Mean   :1092902
                                       3rd Qu.: 4.000                      3rd Qu.:1345000
                                       Max.   :12.000                      Max.   :9000000
     Date             Distance        Postcode            Bedroom2         Bathroom          Car            Landsize         BuildingArea
 Length:8887        Min.   : 0.0   Length:8887        Min.   : 0.000   Min.   :1.000   Min.   : 0.000   Min.   :    0.0   Min.   :   0.0
 Class :character   1st Qu.: 6.4   Class :character   1st Qu.: 2.000   1st Qu.:1.000   1st Qu.: 1.000   1st Qu.:  212.0   1st Qu.: 100.0
 Mode  :character   Median :10.2   Mode  :character   Median : 3.000   Median :2.000   Median : 2.000   Median :  478.0   Median : 132.0
                    Mean   :11.2                      Mean   : 3.078   Mean   :1.646   Mean   : 1.692   Mean   :  523.5   Mean   : 149.3
                    3rd Qu.:13.9                      3rd Qu.: 4.000   3rd Qu.:2.000   3rd Qu.: 2.000   3rd Qu.:  652.0   3rd Qu.: 180.0
                    Max.   :47.4                      Max.   :12.000   Max.   :9.000   Max.   :10.000   Max.   :42800.0   Max.   :3112.0
   YearBuilt     CouncilArea          Lattitude        Longtitude      Regionname        Propertycount
 Min.   :1196   Length:8887        Min.   :-38.17   Min.   :144.4   Length:8887        Min.   :  249
 1st Qu.:1945   Class :character   1st Qu.:-37.86   1st Qu.:144.9   Class :character   1st Qu.: 4382
 Median :1970   Mode  :character   Median :-37.80   Median :145.0   Mode  :character   Median : 6567
 Mean   :1966                      Mean   :-37.80   Mean   :145.0                      Mean   : 7476
 3rd Qu.:2000                      3rd Qu.:-37.75   3rd Qu.:145.1                      3rd Qu.:10331
 Max.   :2019                      Max.   :-37.41   Max.   :145.5                      Max.   :21650
```

-- 05 Some interesting plots –

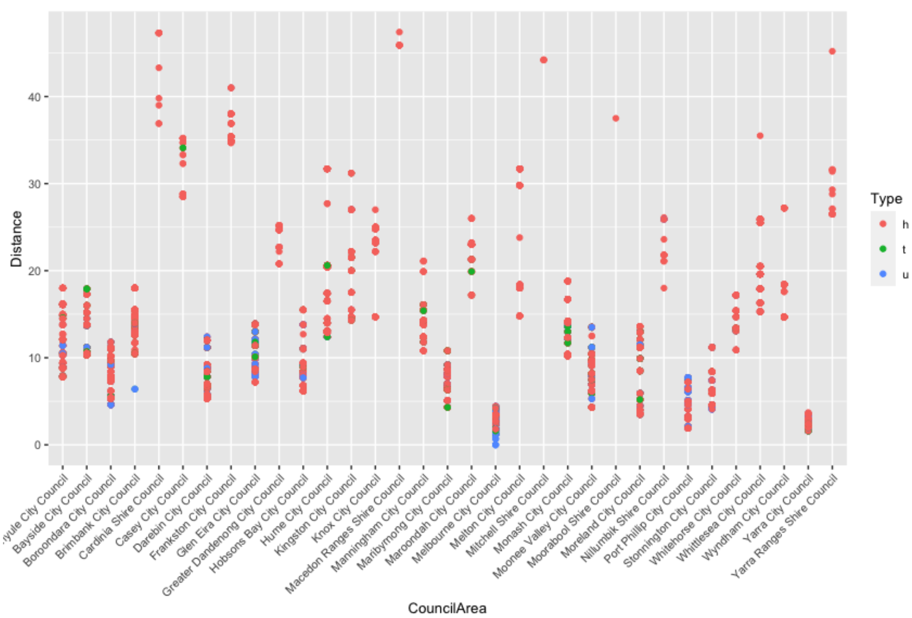
- Below graph shows us how the price varies based on the region name and the type of houses, which is color coded

```
ggplot(data = new_house_data, aes(x = Regionname, y = Price, color = Type))
+ geom_point() + theme(text = element_text(size=10), axis.text.x = element_text(angle=45, hjust=1))
```

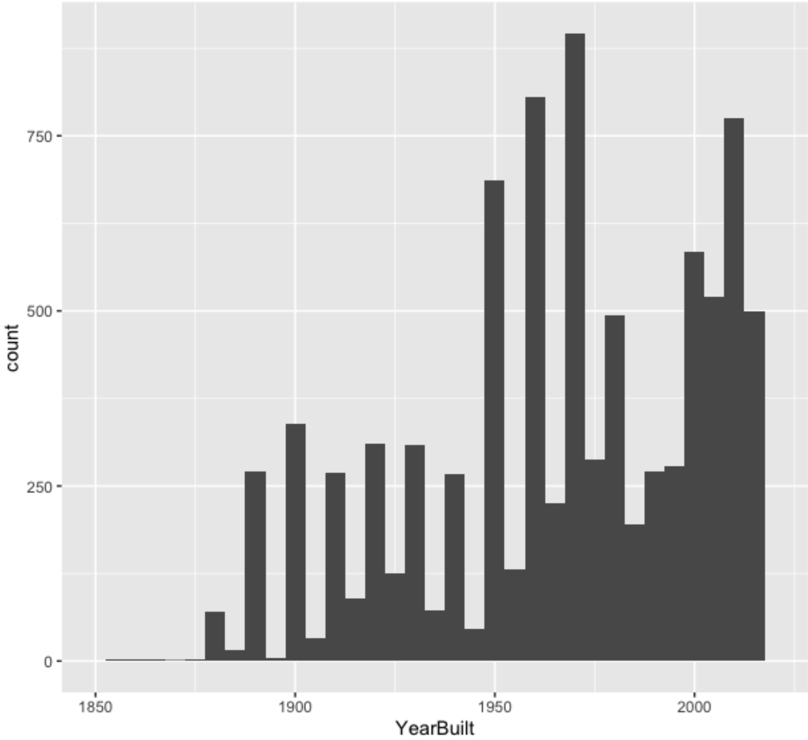


- From the above plot we can conclude that the house of type 'h' in South Eastern Metropolitan and Southern Metropolitan seems to be the costliest. And the U – unit houses in all the other regions seems to have lowest prices

```
ggplot(data = new_house_data, aes(x = CouncilArea, y = Distance, color = Type))
+ geom_point() + theme(text = element_text(size=10), axis.text.x = element_text(angle=45, hjust=1))
```

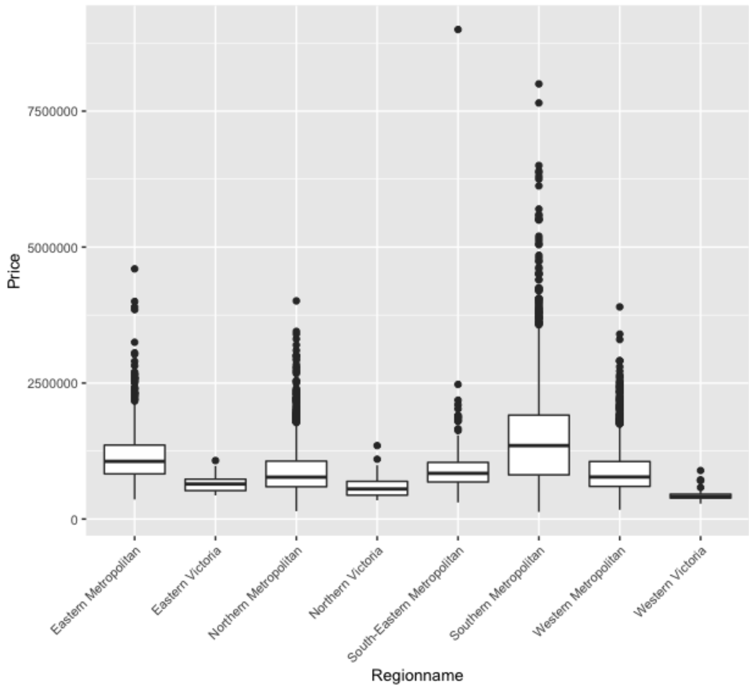


```
ggplot(data = new_house_data, aes(x=YearBuilt)) + geom_histogram(binwidth = 5) + xlim(1850,2020)
```



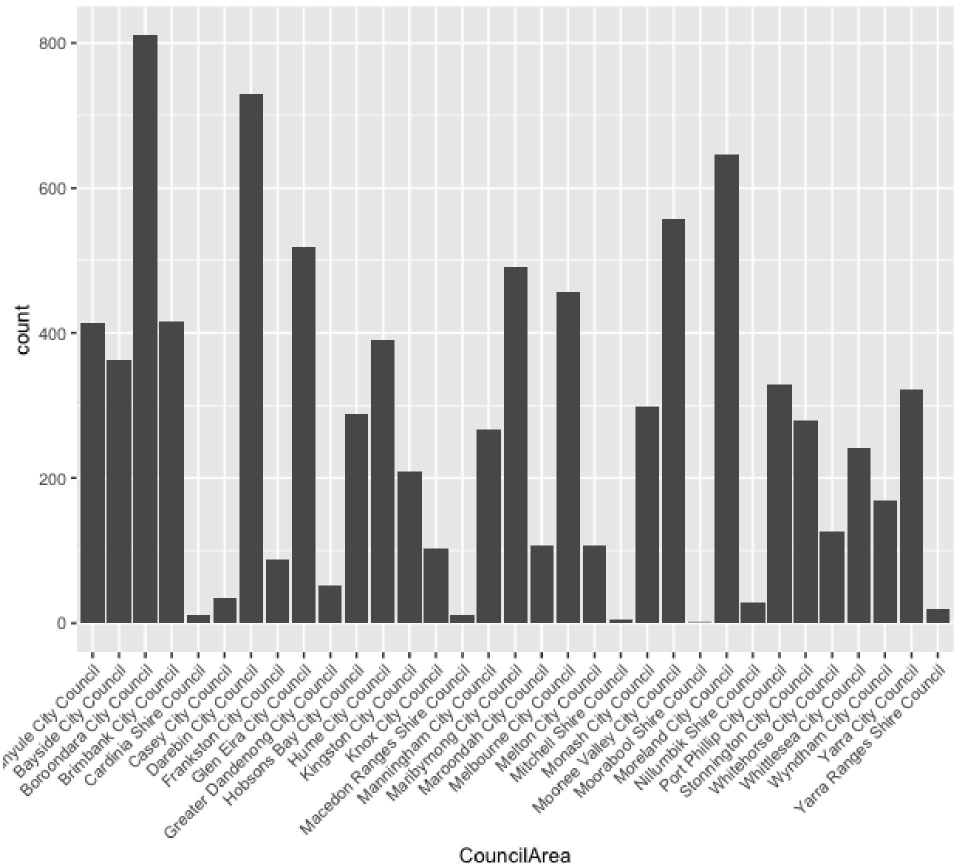
- From the above histogram for the built year, we can conclude that a large number of houses were constructed in the years between 1950 and 1975

```
ggplot(data = new_house_data, aes(x = Regionname, y = Price))
+ geom_boxplot() + theme(text = element_text(size=10), axis.text.x = element_text(angle=45, hjust=1))
```



- From the above box plot, it is evident that houses in Southern Metropolitan are more expensive overall, and the most expensive house of all, which is located in South Eastern Metropolitan, can be seen as an outlier

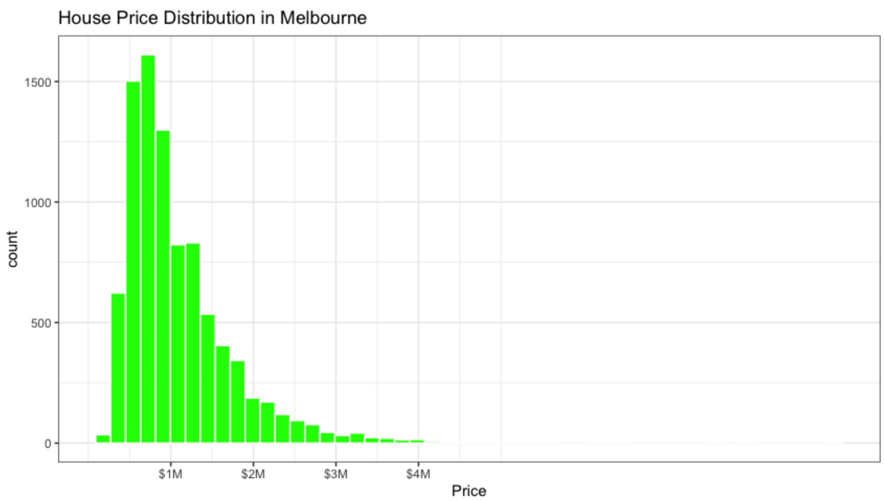
```
ggplot(data = new_house_data, aes(x = CouncilArea)) + geom_bar()
+ theme(text = element_text(size=10), axis.text.x = element_text(angle=45, hjust=1))
```



- From the above bar graph, we can conclude that Baroondara City Council has the highest number of houses and Moorabool Shire Council has the least

-- 06 The target variable --

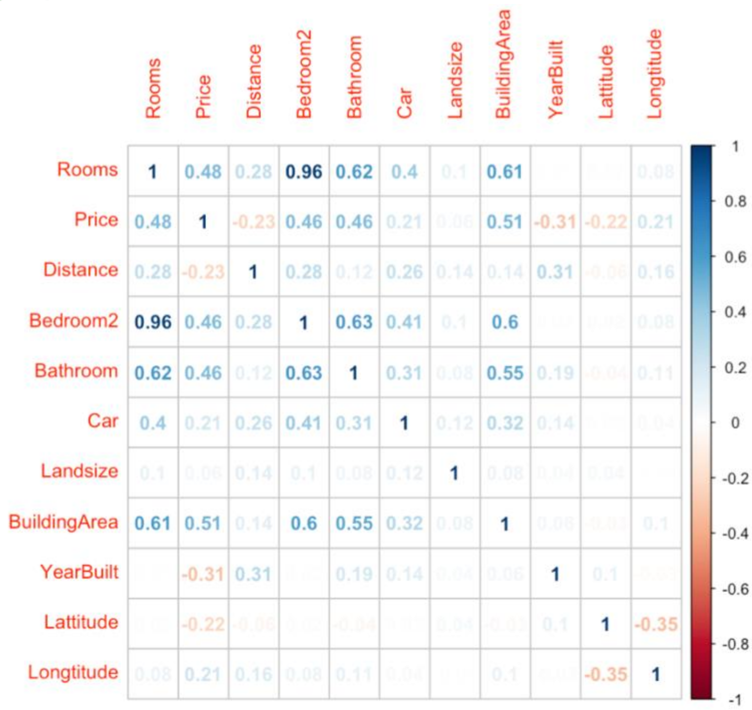
```
ggplot(data=new_house_data ,aes(x=Price)) +geom_histogram(bins = 50,color = "white", fill = "Green")
+scale_x_continuous(breaks = c(1000000,2000000,3000000,4000000),labels = c("$1M","$2M","$3M","$4M"))
+ggtitle("House Price Distribution in Melbourne")+theme_bw()
```

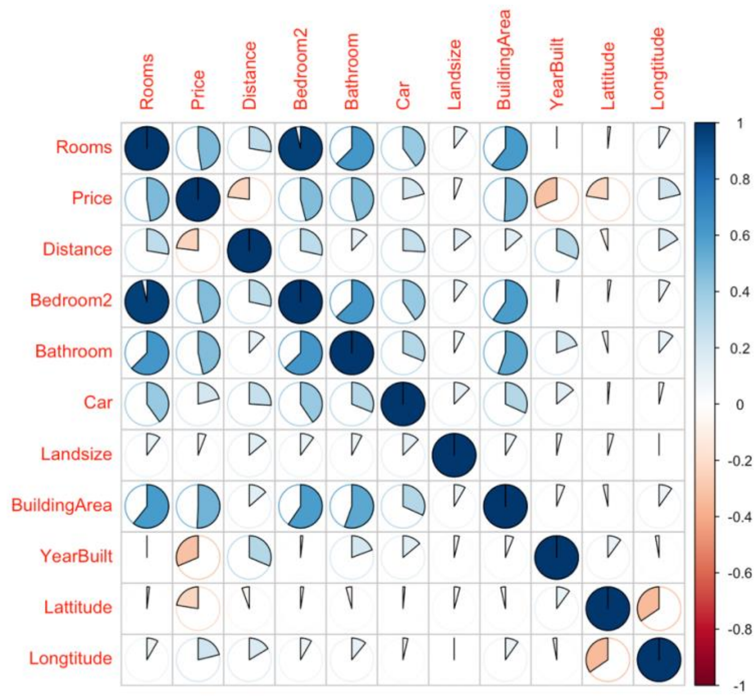


- The target variable is heavily right skewed implying that the mean of the housing prices > median price.
- Mean of Melbourne housing price is $1,050,172 (Australian Dollars)

-- 07 Correlation between the numeric data –

```r
#correlation between the numeric data
head(new_house_data)
house_data_numeric = new_house_data[c(3,5,9,11,12:16,18,19)]
house_data_numeric <- na.omit(house_data_numeric)
head(house_data_numeric)
M = cor(house_data_numeric)
corrplot(M,method = "number")
corrplot(M,method = "pie")
```





-- 08 Removing the highly correlated variable --

```
#Bedrooms are highly correlated with rooms and bathrooms, so they are not considered for the further analysis.
#Adding bedrooms would give a biased result in most modeling.
house_data_numeric = asdata.frame(house_data_numeric[-c(3)])
head(house_data_numeric)
```

- Bedrooms are highly correlated with rooms and bathrooms, so they are not considered for the further analysis
- Adding bedrooms would give a biased result in most modeling

-- 09 Splitting data into training and test set --

```
#split the training and test data
set.seed(100)
sample_size = ceiling(nrow(house_data_numeric) * 0.8)
train_index = sample(nrow(house_data_numeric), sample_size)

training_data = house_data_numeric[train_index, ]
test_data = house_data_numeric[-train_index, ]
```

-- 10 Building and training the model --

```
#Train the model
model = lm(Price ~ Landsize + Bedroom2 +  Distance + Car + Bathroom + BuildingArea
           + YearBuilt + Lattitude + Longtitude , data = training_data)
summary(model)
```

```
> summary(model)

Call:
lm(formula = Price ~ Landsize + Bedroom2 + Distance + Car + Bathroom +
    BuildingArea + YearBuilt + Lattitude + Longtitude, data = training_data)

Residuals:
     Min       1Q   Median       3Q      Max
-5400597  -226914   -50515   147920  8089719

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)  -1.427e+08  6.651e+06 -21.459  < 2e-16 ***
Landsize      3.146e+01  4.642e+00   6.778 1.31e-11 ***
Bedroom2      1.533e+05  8.319e+03  18.426  < 2e-16 ***
Distance     -3.204e+04  8.922e+02 -35.912  < 2e-16 ***
Car           6.354e+04  6.088e+03  10.437  < 2e-16 ***
Bathroom      2.110e+05  1.016e+04  20.760  < 2e-16 ***
BuildingArea  2.006e+03  7.566e+01  26.512  < 2e-16 ***
YearBuilt    -5.244e+03  1.636e+02 -32.063  < 2e-16 ***
Lattitude    -1.200e+06  6.246e+04 -19.207  < 2e-16 ***
Longtitude    7.441e+05  4.816e+04  15.452  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 444200 on 7100 degrees of freedom
Multiple R-squared:  0.5838,    Adjusted R-squared:  0.5833
F-statistic:  1107 on 9 and 7100 DF,  p-value: < 2.2e-16
```

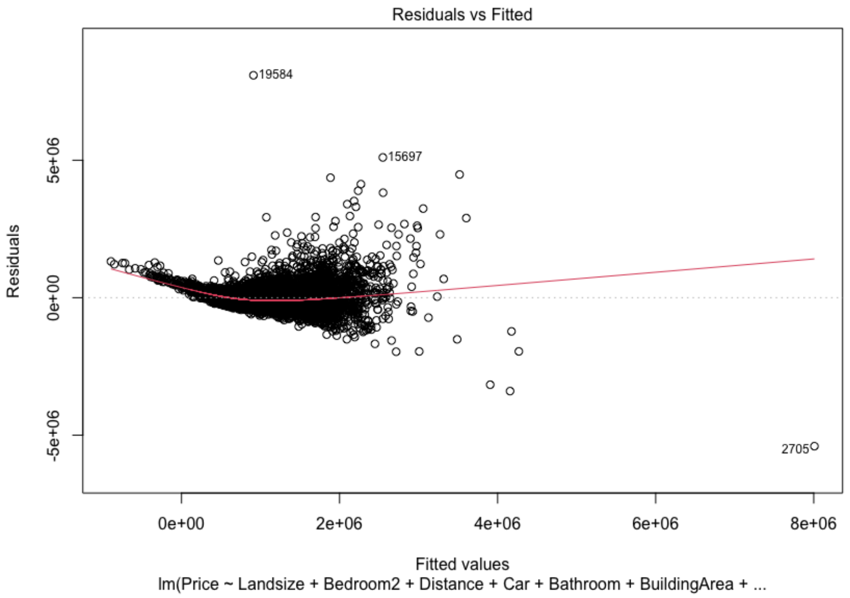
- We can conclude that all these predictors - Landsize, Bedrooms, Distance, Car, Bathrooms, Year Built, Lattitude and Longitude – are helpful in predicting the house price since the p-value for each of these predictors are very small

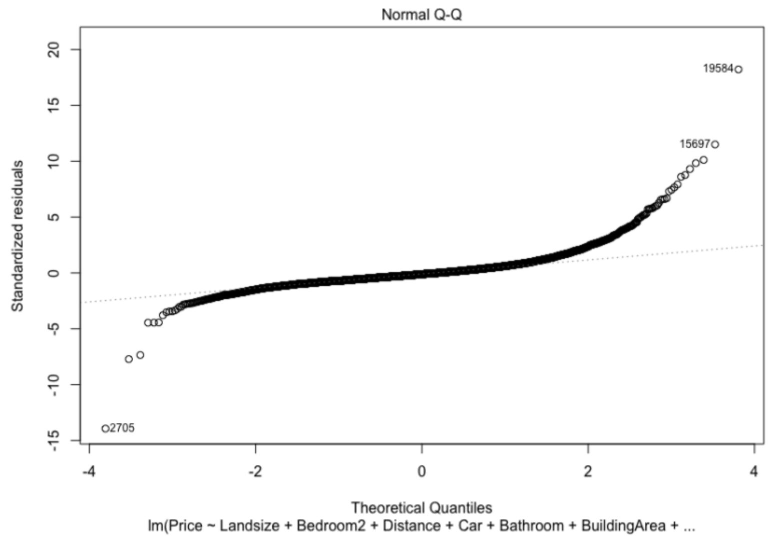
- The above model seems to have given the best results in terms of R-squared value of 0.5838 and F-statistic of 1107 on p=9 and n=7100 DF. The p-value of the entire model is found to be 2.2e-16 and hence we can conclude this is the best fit

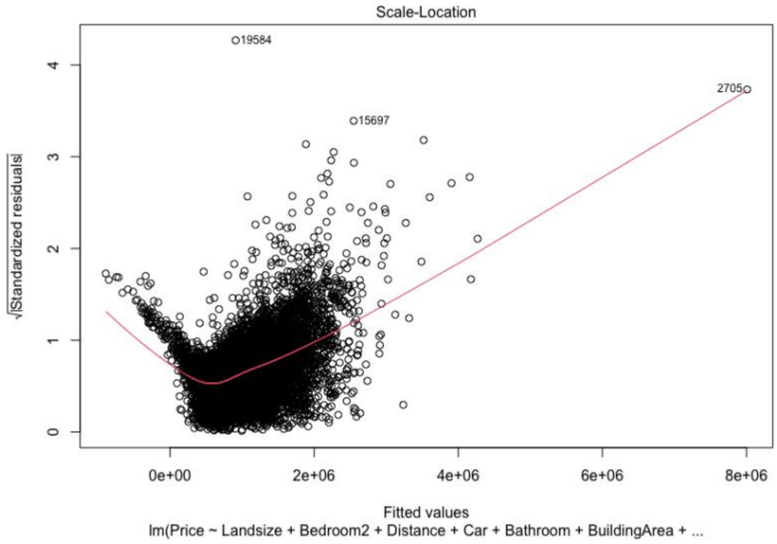
-- 11 Validate, Calculate MSE and Plot the model --

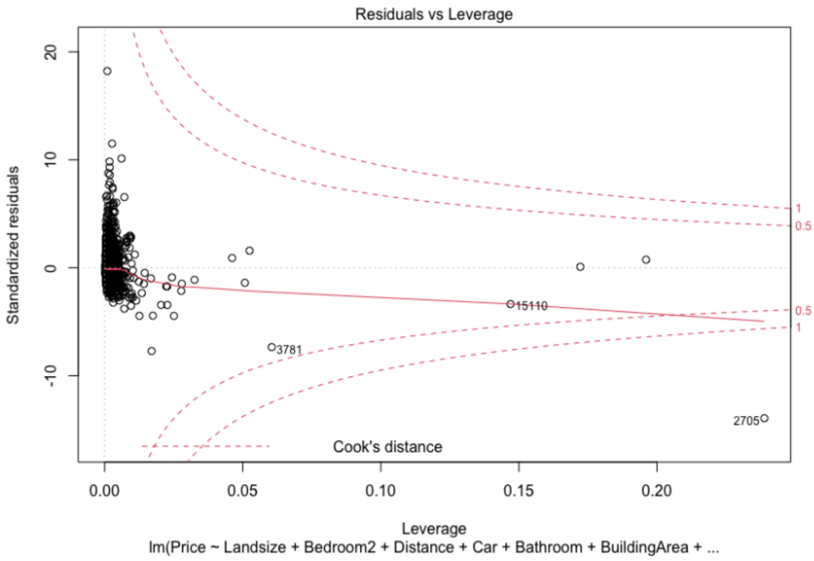
```
#Validate the model
predicted_price = data.frame(predict(model, test_data))

#MSE of the model
head(predicted_price)
mse(test_data$Price, predicted_price$predict.model..test_data.)

plot(model)
```

```
> #Validate the model
> predicted_price = data.frame(predict(model, test_data))
> #MSE of the model
> head(predicted_price)
  predict.model..test_data.
7                  969695.8
26                 892497.8
36                1325148.9
38                 764848.9
44                 958681.5
45                1174549.8
> mse(test_data$Price, predicted_price$predict.model..test_data.)
[1] 177148419545
```









-- End of the report --