# HOUSE PRICE PREDICTION

**BY**

**NWUDO CHIKAEZE FIDELIS**

**TABLE OF CONTENT**

1) INTRODUCTION---------------------------------------------------------------------------------------------------2

- Description of variables-------------------------------------------------------------------------------------------2

- Source of data----------------------------------------------------------------------------------------------------------2

2) IMPORT LIBRARIES AND DATA-------------------------------------------------------------------------------3

3) EXPLORATORY DATA ANALYSIS---------------------------------------------------------------------------5

- Identification of missing value---------------------------------------------------------------------------------5

- Correlation plot---------------------------------------------------------------------------------------------------------6

- Histograms and boxplots of data-------------------------------------------------------------------------------8

- Outlier detection-------------------------------------------------------------------------------------------------------8

- Cook's Distance--------------------------------------------------------------------------------------------------------9

- Identify multi-collinearity ----------------------------------------------------------------------------------------12

4) Model Building and Selection---------------------------------------------------------------------------------13

- Selection of best subset models using the "olsrr" package-----------------------------------------------14

5) Model adequacy Checks and Remedial Measures-----------------------------------------------------16

- Model Adequacy Checks------------------------------------------------------------------------------------------16

- Breusch-Pagan Test--------------------------------------------------------------------------------------------------18

- Remedial measures--------------------------------------------------------------------------------------------------19

6) Conclusion-------------------------------------------------------------------------------------------------------------23

## 1.0 INTRODUCTION

## DESCRIPTION OF DATA

The house price prediction data set was built for regression analysis, and it includes 414 observations with six (6) predictor variables and one (1) dependent variable. My data set has 5 numeric predictors, 1 categorical predictor and a numeric target variable. Throughout my analysis, I used a significance level of 0.05

Predictors.

- Date of purchase: date the house was bought

- House age: Median age of a house within a block. A lower number is a newer building.

- MRT station proximity: Location of the house with respect to MRT station

- stores: Number of stores within proximity of the house.

- Latitude: A measure of how far north a house is. A higher value is farther north.

- Longitude: A measure of how far west a house is. A higher value is farther west.

Target variable

- House price: House value per unit area.

**SOURCE**

I got this dataset from Kaggle. This real estate data set originates from UCI Machine Learning Repository.

## 2.0 LIBRARIES AND DATA

```
library(tidyverse)

## ── Attaching packages ─────────────────────────────────────── tidyverse
1.3.2 ──
## ✓ ggplot2 3.4.0      ✓ purrr   0.3.5
## ✓ tibble  3.1.8      ✓ dplyr   1.0.10
## ✓ tidyr   1.2.1      ✓ stringr 1.4.1
## ✓ readr   2.1.3      ✓ forcats 0.5.2
## ── Conflicts ──────────────────────────────────────────
tidyverse_conflicts() ──
## ✗ dplyr::filter() masks stats::filter()
## ✗ dplyr::lag()    masks stats::lag()

library(corrplot)

## corrplot 0.92 loaded

library(MASS)

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

library(olsrr)

##
## Attaching package: 'olsrr'
##
## The following object is masked from 'package:MASS':
##
##     cement
##
## The following object is masked from 'package:datasets':
##
##     rivers

library(gridExtra)

##
## Attaching package: 'gridExtra'
##
## The following object is masked from 'package:dplyr':
##
##     combine

P1 <- read_csv("Real estate.csv")
```

```
## Rows: 414 Columns: 8
## ── Column specification
────────────────────────────────────────────────────────
## Delimiter: ","
## dbl (8): No, X1 transaction date, X2 house age, X3 distance to the nearest
M...
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this
message.
```

```
PD <- P1[,-c(1)]
head(PD)
```

```
## # A tibble: 6 × 7
##   `X1 transaction date` `X2 house age` X3 dist…¹ X4 nu…² X5 la…³ X6 lo…⁴ Y
hou…⁵
##                   <dbl>          <dbl>     <dbl>   <dbl>   <dbl>   <dbl>
<dbl>
## 1                 2013.           32        84.9      10    25.0    122.
37.9
## 2                 2013.           19.5     307.        9    25.0    122.
42.2
## 3                 2014.           13.3     562.        5    25.0    122.
47.3
## 4                 2014.           13.3     562.        5    25.0    122.
54.8
## 5                 2013.            5       391.        5    25.0    122.
43.1
## 6                 2013.            7.1    2175.        3    25.0    122.
32.1
## # … with abbreviated variable names ¹`X3 distance to the nearest MRT
station`,
## #   ²`X4 number of convenience stores`, ³`X5 latitude`, ⁴`X6 longitude`,
## #   ⁵`Y house price of unit area`
```

```
X1 <- PD$`X1 transaction date`
X2 <- PD$`X2 house age`
X3 <- PD$`X3 distance to the nearest MRT station`
X4 <- PD$`X4 number of convenience stores`
X5 <- PD$`X5 latitude`
X6 <- PD$`X6 longitude`
Y <- PD$`Y house price of unit area`
```

```
summary(PD)
```

```
##  X1 transaction date  X2 house age    X3 distance to the nearest MRT
station
##  Min.   :2013        Min.   : 0.000   Min.   :  23.38
##  1st Qu.:2013        1st Qu.: 9.025   1st Qu.: 289.32
##  Median :2013        Median :16.100   Median : 492.23
```

```
##  Mean   :2013        Mean   :17.713   Mean   :1083.89
##  3rd Qu.:2013        3rd Qu.:28.150   3rd Qu.:1454.28
##  Max.   :2014        Max.   :43.800   Max.   :6488.02
##  X4 number of convenience stores  X5 latitude     X6 longitude
##  Min.   : 0.000                  Min.   :24.93   Min.   :121.5
##  1st Qu.: 1.000                  1st Qu.:24.96   1st Qu.:121.5
##  Median : 4.000                  Median :24.97   Median :121.5
##  Mean   : 4.094                  Mean   :24.97   Mean   :121.5
##  3rd Qu.: 6.000                  3rd Qu.:24.98   3rd Qu.:121.5
##  Max.   :10.000                  Max.   :25.01   Max.   :121.6
##  Y house price of unit area
##  Min.   :  7.60
##  1st Qu.: 27.70
##  Median : 38.45
##  Mean   : 37.98
##  3rd Qu.: 46.60
##  Max.   :117.50
```

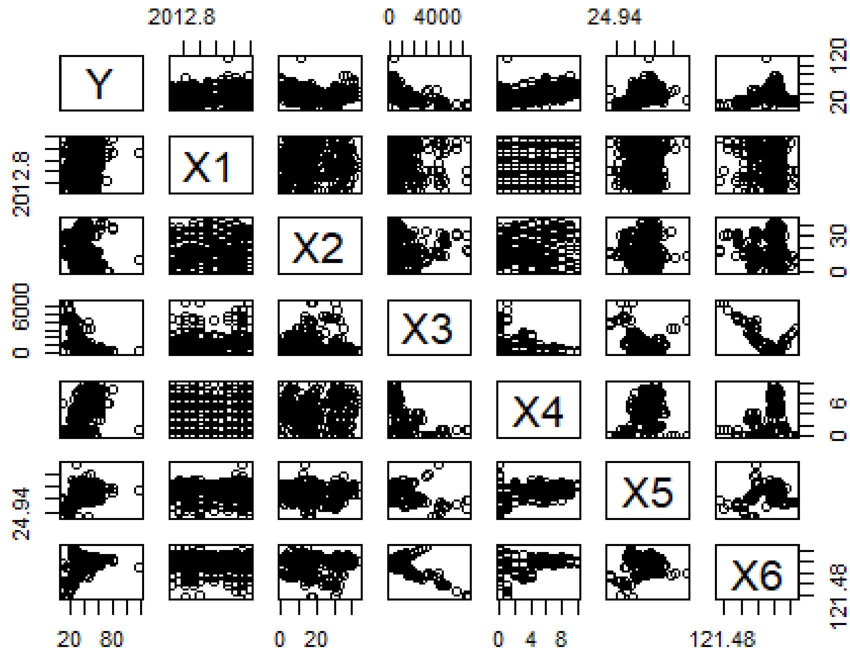
3.0 EXPLORATORY DATA ANALYSIS

3.1 Identification of missing value.

```
sum(is.na(PD))

## [1] 0
```

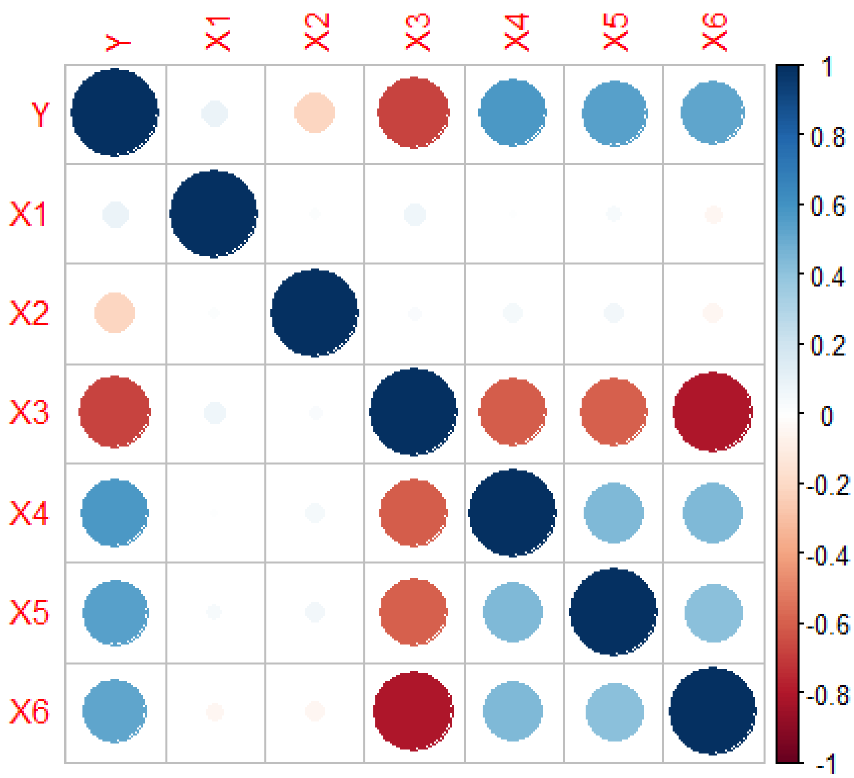
3.2 Correlation plot

```
xy.mat <- cbind( Y,X1, X2, X3, X4, X5, X6)
pairs(xy.mat)
```




```
df.Cor <- cor(xy.mat)
df.Cor

##               Y           X1          X2          X3          X4
X5
## Y   1.00000000  0.087490606 -0.21056705 -0.67361286  0.571004911
0.54630665
## X1  0.08749061  1.000000000  0.01754877  0.06087995  0.009635445
0.03505776
## X2 -0.21056705  0.017548767  1.00000000  0.02562205  0.049592513
0.05441990
## X3 -0.67361286  0.060879953  0.02562205  1.00000000 -0.602519145 -
0.59106657
## X4  0.57100491  0.009635445  0.04959251 -0.60251914  1.000000000
0.44414331
## X5  0.54630665  0.035057756  0.05441990 -0.59106657  0.444143306
1.00000000
## X6  0.52328651 -0.041081778 -0.04852005 -0.80631677  0.449099007
0.41292394
##              X6
## Y   0.52328651
## X1 -0.04108178
## X2 -0.04852005
## X3 -0.80631677
## X4  0.44909901
## X5  0.41292394
## X6  1.00000000
```

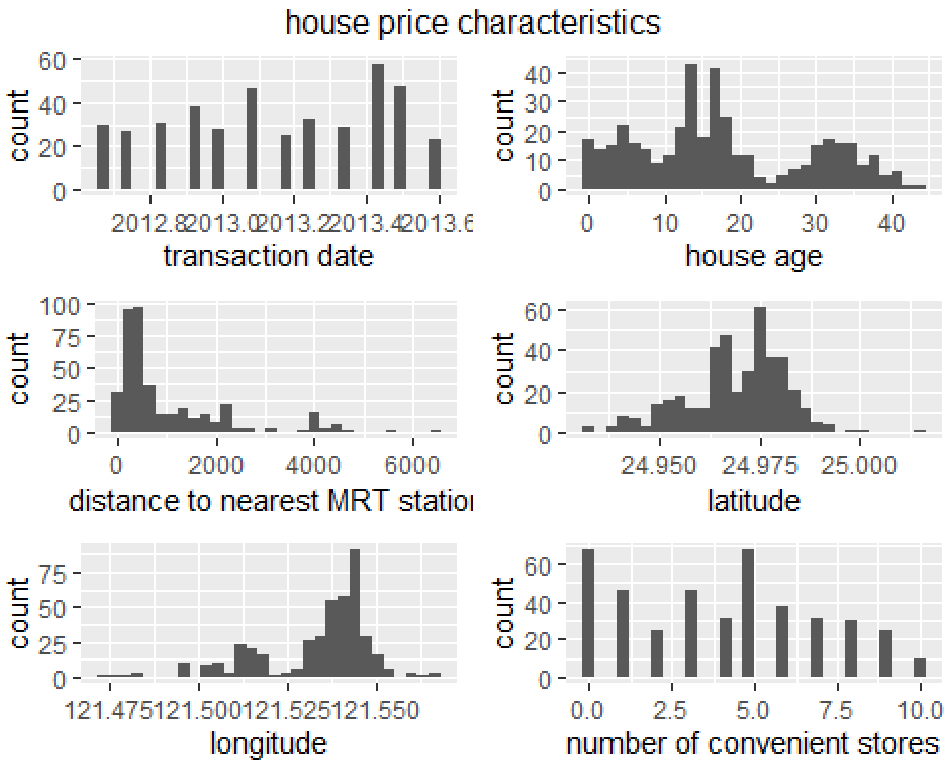
```
corrplot(df.Cor)
```




Firstly,Y and X1 have a very weak positive association. Y and X2 and Y and X3 show a negative correlation when viewed in the correlation matrix, but Y and X4, Y and X5 and Y and X6 have similar positive correlations. Also, X3 and X6 have a very strong negative correlation.

3.3 Histograms and Boxplots of data

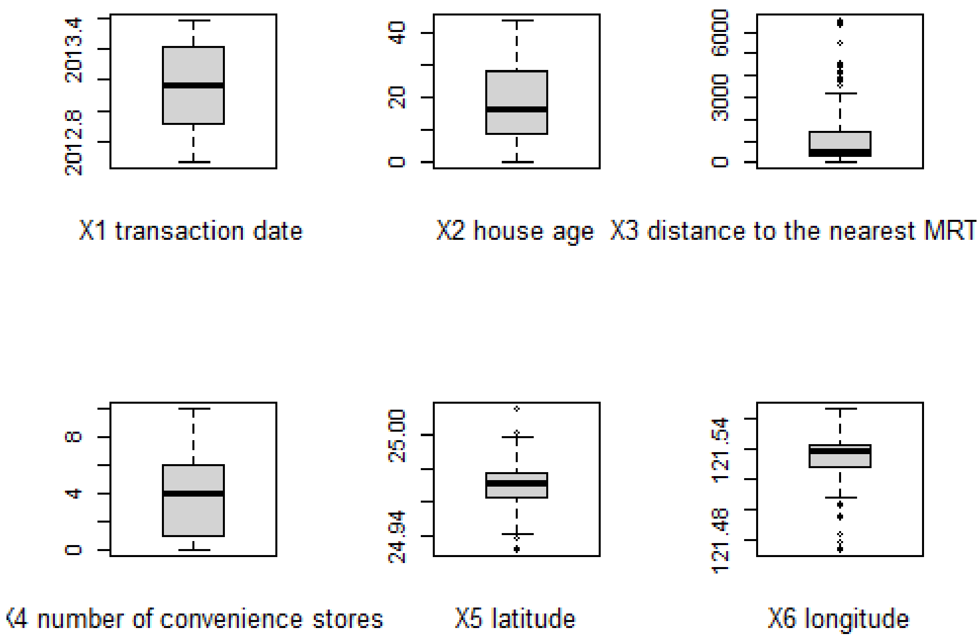
```
p1 <- ggplot(data = PD, mapping = aes(x = `X1 transaction date`)) +
geom_histogram() + xlab("transaction date")
p2 <- ggplot(data = PD, mapping = aes(x = `X2 house age`)) + geom_histogram()
+ xlab("house age")
p3 <- ggplot(data = PD, mapping = aes(x = `X3 distance to the nearest MRT
station`)) + geom_histogram() + xlab("distance to nearest MRT station")
p4 <- ggplot(data = PD, mapping = aes(x = `X4 number of convenience stores`))
+ geom_histogram() + xlab("number of convenient stores")
p4 <- ggplot(data = PD, mapping = aes(x = `X5 latitude`)) + geom_histogram()
+ xlab("latitude")
p5 <- ggplot(data = PD, mapping = aes(x = `X6 longitude`)) + geom_histogram()
+ xlab("longitude")
p6 <- ggplot(data = PD, mapping = aes(x = `X4 number of convenience stores`))
+ geom_histogram() + xlab("number of convenient stores")


grid.arrange(p1,p2,p3,p4,p5,p6, nrow=3, top="house price characteristics")
```




### 3.4 OUTLIER DETECTION

There are many ways to deal with influential points including: removing these points, replacing these points with some value like the mean or median, or simply keeping the points in the model. But in project I used the Cook's Distance to to identify influential points. Cook's distance, often denoted Di, is used in regression analysis to identify influential data points that may negatively affect your regression model. A data point that has a large value for Cook's Distance indicates that it strongly influences the fitted values. I begin by plotting box plots of all variables in my data.

```
oldpar = par(mfrow = c(2,3))
for ( i in 1:6 ) {
  boxplot(PD[[i]])
  mtext(names(PD)[i], cex = 0.8, side = 1, line = 2)
}
```

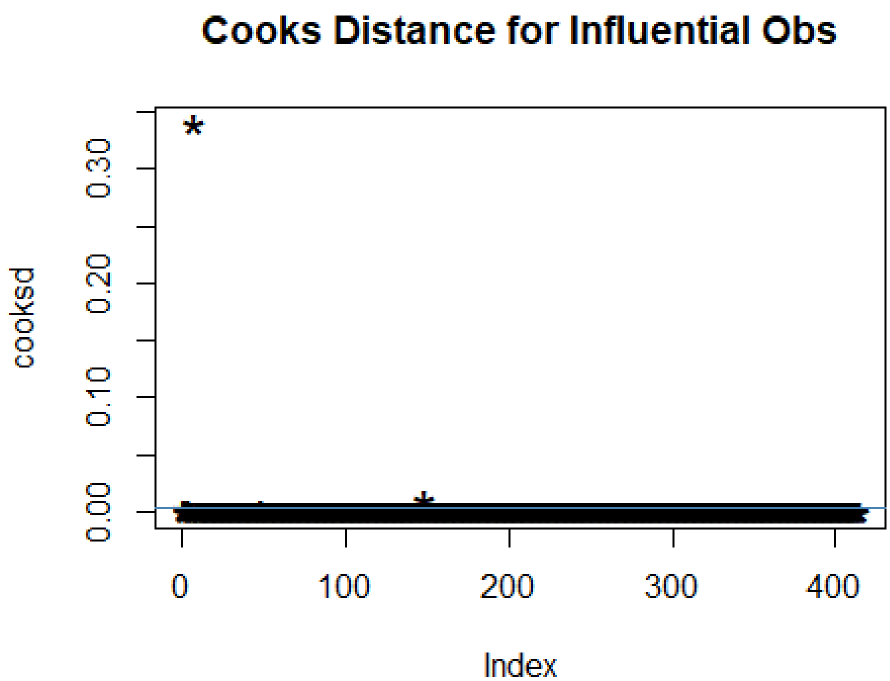



```
par(oldpar)
```

### 3.5 Cook's Distance

For each variables, we consider observations that lie outside 1.5 * IQR as outliers. Then I fit my model with the dataset with outliers before finding the cook's distance for each observation in the dataset. I then plot the cook's distance with a horizontal line at 4/n to see which observations exceed the threshold. Thus, we would identify any observation above the cut off line as influential data points that have a negative impact on the regression model.

```
outliers = c()
for ( i in 1:6 ) {
  stats = boxplot.stats(PD[[i]])$stats
  bottom_outlier_rows = which(PD[[i]] < stats[1])
  top_outlier_rows = which(PD[[i]] > stats[5])
  outliers = c(outliers , top_outlier_rows[ !top_outlier_rows %in% outliers ]
)
  outliers = c(outliers , bottom_outlier_rows[ !bottom_outlier_rows %in%
outliers ] )
}

mod = lm(Y ~ ., data = PD)
cooksd = cooks.distance(mod)
plot(cooksd, pch = "*", cex = 2, main = "Cooks Distance for Influential Obs")
abline(h = 4*mean(cooksd, na.rm = T), col = "steelblue")
```




```
head(PD[cooksd > 4 * mean(cooksd, na.rm=T), ])

## # A tibble: 2 × 7
##   `X1 transaction date` `X2 house age` X3 dist…¹ X4 nu…² X5 la…³ X6 lo…⁴ Y
hou…⁵
##                   <dbl>          <dbl>     <dbl>   <dbl>   <dbl>   <dbl>
<dbl>
## 1                 2013.           20.3      288.       6    25.0    122.
46.7
## 2                 2014.           16.4     3781.       0    24.9    122.
45.1
## # … with abbreviated variable names ¹`X3 distance to the nearest MRT
station`,
## #   ²`X4 number of convenience stores`, ³`X5 latitude`, ⁴`X6 longitude`,
## #   ⁵`Y house price of unit area`
```

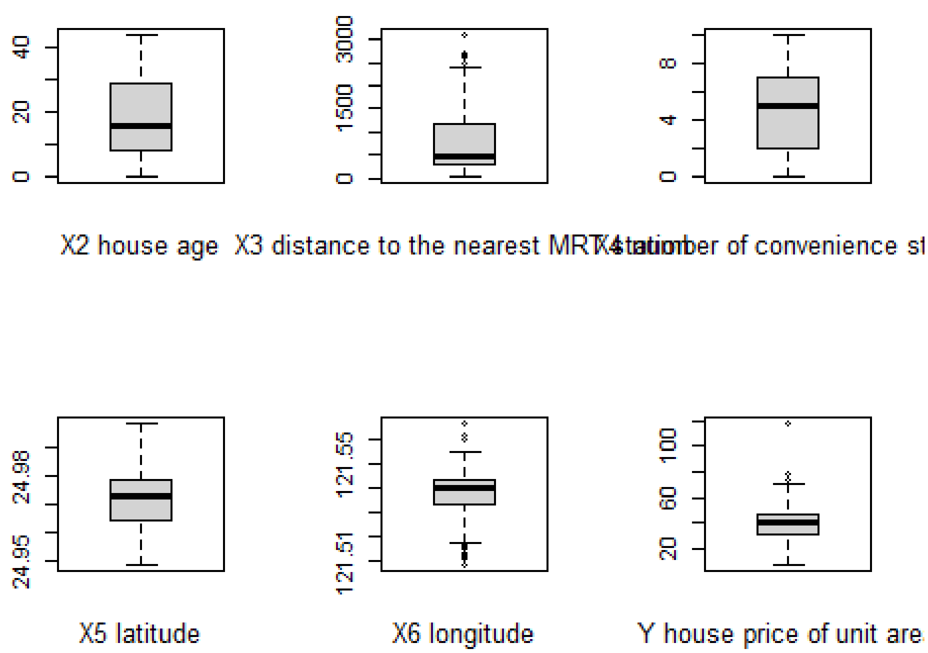
Taking out the outliers from the data set

```
coutliers = as.numeric(rownames(PD[cooksd > 4 * mean(cooksd, na.rm=T), ]))
outliers = c(outliers , coutliers[ !coutliers %in% outliers ] )

PD1 = PD[-outliers, ]
summary(PD1)

##  X1 transaction date  X2 house age   X3 distance to the nearest MRT
station
##  Min.   :2013        Min.   : 0.00   Min.   :  23.38
```

```
##  1st Qu.:2013        1st Qu.: 8.00   1st Qu.: 279.17
##  Median :2013        Median :15.60   Median : 462.87
##  Mean   :2013        Mean   :17.47   Mean   : 752.50
##  3rd Qu.:2013        3rd Qu.:28.98   3rd Qu.:1145.86
##  Max.   :2014        Max.   :43.80   Max.   :3085.17
##  X4 number of convenience stores  X5 latitude     X6 longitude
##  Min.   : 0.000                  Min.   :24.95   Min.   :121.5
##  1st Qu.: 2.000                  1st Qu.:24.96   1st Qu.:121.5
##  Median : 5.000                  Median :24.97   Median :121.5
##  Mean   : 4.457                  Mean   :24.97   Mean   :121.5
##  3rd Qu.: 6.750                  3rd Qu.:24.98   3rd Qu.:121.5
##  Max.   :10.000                  Max.   :25.00   Max.   :121.6
##  Y house price of unit area
##  Min.   :  7.60
##  1st Qu.: 30.60
##  Median : 40.05
##  Mean   : 39.85
##  3rd Qu.: 47.38
##  Max.   :117.50
```

BOXPLOTS AFTER THE INFLUENCING OUTLIERS HAVE BEEN TAKEN OUT

```
newpar = par(mfrow = c(2,3))
for ( i in 2:7) {
  boxplot(PD1[[i]])
  mtext(names(PD1)[i], cex = 0.8, side = 1, line = 2)
}
```



```
par(newpar)

Y_1 <- PD1$`Y house price of unit area`
X_1 <- PD1$`X1 transaction date`
X_2 <- PD1$`X2 house age`
X_3 <- PD1$`X3 distance to the nearest MRT station`
X_4 <- PD1$`X4 number of convenience stores`
X_5 <- PD1$`X5 latitude`
X_6 <- PD1$`X6 longitude`
```

3.6 TEST FOR MULTICOLLINEARITY

Multicollinearity occurs when independent variables in a regression model are correlated. Multicollinearity causes a lot of things such as the estimated standard deviations of the regression coefficients becoming large when predictor variables in the model are highly correlated. Also the extra sum of squares associated with a predictor variables may vary. There are informal ways to check for multicollinearity but in this project I used the variance inflation factor.

I used variance inflation factor method which is a formal method of detecting the presence of multicollinearity to measure how much the variances of the estimated regression coefficients are inflated as compared to when the predictor variables are not linearly related.

```
xmat <- cbind (X_1, X_2, X_3, X_5, X_6)
rxx <- cor(xmat)
rxx

##              X_1          X_2         X_3         X_5         X_6
## X_1  1.000000000  0.009603967  0.04539664  0.04840874 -0.01817612
## X_2  0.009603967  1.000000000 -0.06181577  0.10957984  0.01186430
## X_3  0.045396640 -0.061815770  1.00000000 -0.35720267 -0.53688699
## X_5  0.048408744  0.109579840 -0.35720267  1.00000000  0.05569235
## X_6 -0.018176119  0.011864299 -0.53688699  0.05569235  1.00000000

rxx.inv <- solve(rxx)
mean(diag(rxx.inv))

## [1] 1.265951
```

- From the analysis, the mean of the variance inflation factor is 1.265951 and it indicates that it is not a severe case of of multicollinearity since it is not greater than 5

4.0 MODEL SELECTION

For this analysis, I chose to use the Best Subset Regression selection.

- Three criterion for model selection were examined;

1) Mallow's Ck criterion: Under this criterion, we seek the model with a Ck value that is small and near k. A small Ck value indicates that the total mean squared error for that model is small.

2) AIC Criterion: The Akaike Information Criterion (AIC) is selected based on the model with smallest AIC.

3) SBC Criterion: The Schwarz' Bayesian Criterion(SBC) is selected based on the model with the smallest SBC

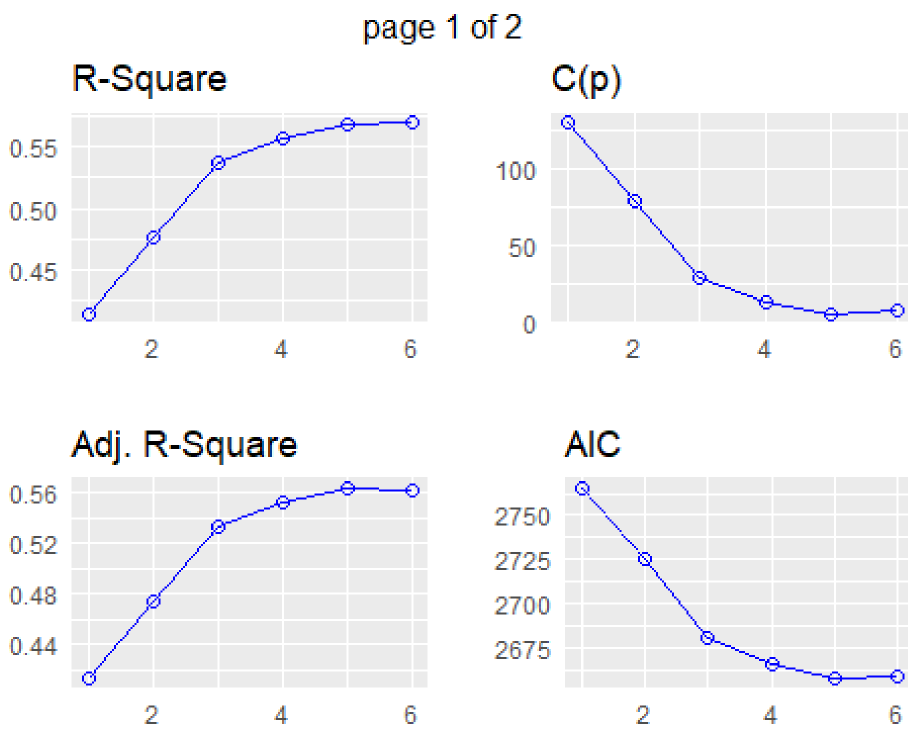
4) R2 adjusted: The model with the largest R2 value is selected.

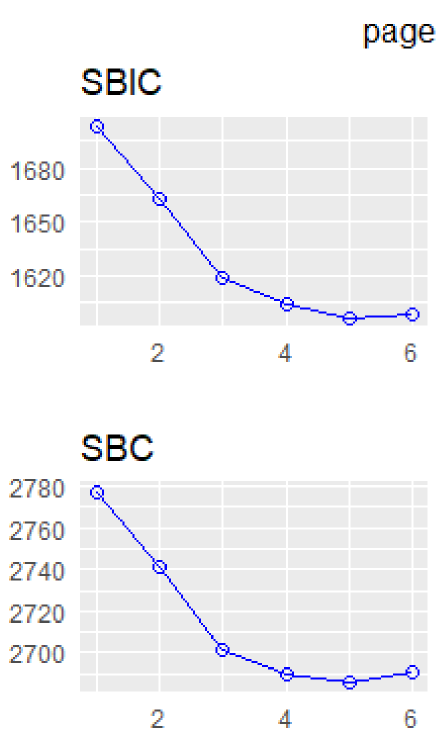
```
model <- lm(Y_1 ~ X_1 + X_2 + X_3 + X_4 + X_5 + X_6, data = PD1)
k = ols_step_best_subset(model)
k

##        Best Subsets Regression
## --------------------------------------
## Model Index    Predictors
## --------------------------------------
##      1         X_3
##      2         X_2 X_3
##      3         X_2 X_3 X_5
##      4         X_2 X_3 X_4 X_5
##      5         X_1 X_2 X_3 X_4 X_5
##      6         X_1 X_2 X_3 X_4 X_5 X_6
## --------------------------------------
##
##                                                          Subsets Regression
Summary
## ---------------------------------------------------------------------------
---------------------------------------------------------------
##                        Adj.        Pred
## Model    R-Square    R-Square    R-Square     C(p)         AIC
SBIC           SBC          MSEP         FPE        HSP        APC
## ---------------------------------------------------------------------------
---------------------------------------------------------------
##   1        0.4140      0.4124       0.408    129.6162    2765.2082
1702.7100    2776.9809    35217.6517     94.6684     0.2538     0.5923
##   2        0.4761      0.4733      0.4667     78.6759    2725.3176
1662.9224    2741.0146    31571.0165     85.0910     0.2281     0.5324
##   3        0.5367      0.5329       0.526     29.0176    2681.3518
1619.5545    2700.9731    27995.4724     75.6537     0.2029     0.4733
##   4        0.5577      0.5529      0.5436     13.0955    2665.9864
1604.5381    2689.5319    26797.9340     72.6086     0.1947     0.4543
##   5        0.5692      0.5633      0.5533      5.3478    2658.1915
```

```
1597.0421    2685.6613    26176.3534    71.1110    0.1907    0.4449
##   6        0.5696      0.5625      0.5514      7.0000    2659.8373
1598.7375    2691.2313    26223.0228    71.4247    0.1916    0.4469
## ----------------------------------------------------------------------------
--------------------------------------------------------------
## AIC: Akaike Information Criteria
##  SBIC: Sawa's Bayesian Information Criteria
##  SBC: Schwarz Bayesian Criteria
##  MSEP: Estimated error of prediction, assuming multivariate normality
##  FPE: Final Prediction Error
##  HSP: Hocking's Sp
##  APC: Amemiya Prediction Criteria
```

• We see that Model 5 with X1, X2, X3, X4 and X5 as predictor variables is selected based on the R2 adjusted criterion because this model has the largest value of R2 adjusted. The C(p) criterion leads to model 5 with predictor variables because the C(p) value for this model is near k=6 and is small. This 5 predictor variable model is also selected by the AIC and SBC criterion because it has the smallest AIC and SBC value. So based on all these, I choose model 5 as my best model.

```
plot(k)
```




```
model_new <- lm(Y_1 ~ X_1 + X_2 + X_3 + as_factor(X_4) + X_5)
```

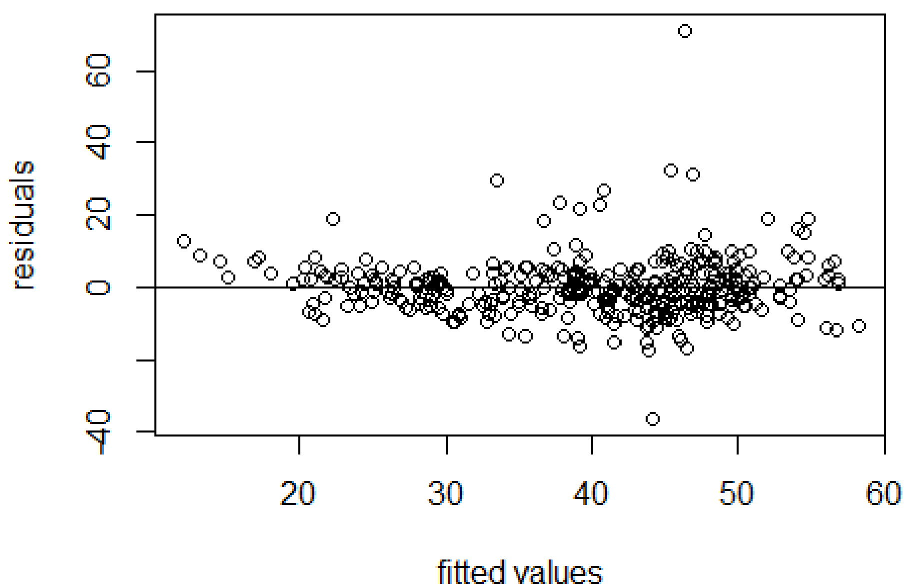
5.0 MODEL ADEQUACY CHECKING AND REMEDIAL MEASURES

5.1 MODEL ADEQUACY CHECKING

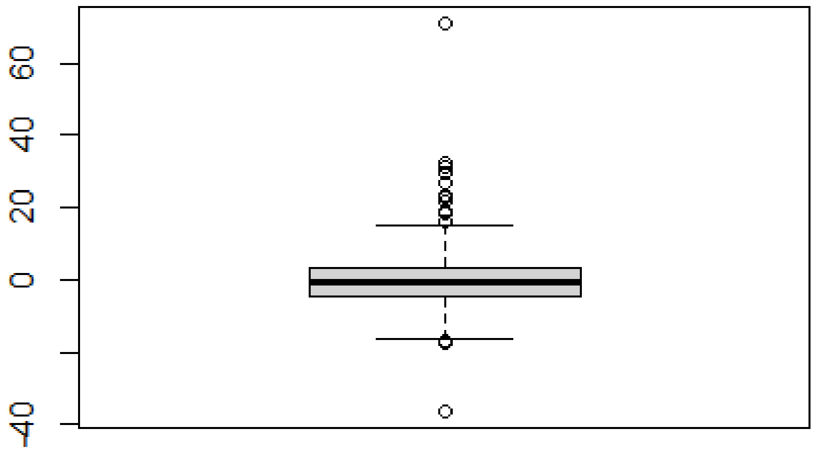
In this section I will be verifying the validity the assumptions underlying the linear regression model. These assumptions include: - Linearity of Regression Function - Constant error Variance - Independence of error terms - Normal distribution of error terms

I used graphical and formal statistical test to examine the model assumptions. I plotted graphs of residuals against fitted values, box plot of residuals and normal probability plot of residuals.

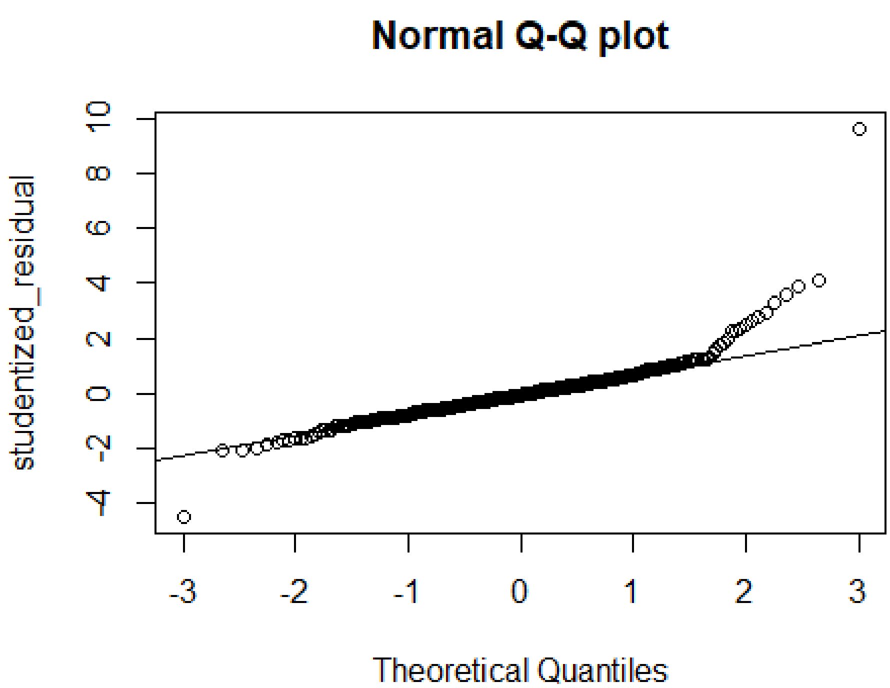
```
plot(fitted(model_new), resid(model_new), xlab = "fitted values", ylab =
"residuals", abline(0,0))
```




```
boxplot(resid(model_new))
```




```
qqnorm(y = studres(model_new), main = "Normal Q-Q plot", xlab = "Theoretical
Quantiles", ylab = "studentized_residual", plot.it = TRUE)
qqline(y= studres(model_new), distribution = qnorm)
```



- The graph of residuals against fitted values is not that, so i went further to perform a formal statistical test for non constant variance.

5.2 BREUSCH_PAGAN TEST

A formal statistical test for non constant variance was conducted called the Breusch-Pagan test.The null-hypothesis of this test is that the model is homoscedasticity (The residuals are distributed with equal variance). The alternative hypothesis is that that the model is heteroscedastic (The residuals are not distributed with equal variance). Its assumptions include:

1. Independent error terms.
2. normally distributed error terms
3. variances increase exponentially as the predictor increases.

model: Y = Bo + B1X1 + B2X2 + B3X3 + B4X7 + B5X5 Null hypothesis: B1=B2=B3=B4=B5=0, Ha: At least one B != 0

The test statistic is given by:

$$\chi_0 = \frac{n^2}{2} * \frac{SSR^*}{SSE^2}$$

$$H_0$$

is rejected if for a fixed alpha value,

$$\chi_0 > \chi^2(1-\alpha, 1)$$

where

$$\chi_0 > \chi^2(1-\alpha, 1)$$

is the

$$(1-\alpha)100$$

percentile of the chi square distribution with 1 degrees of freedom

```
anova(model)

## Analysis of Variance Table
##
## Response: Y_1
##            Df  Sum Sq Mean Sq  F value    Pr(>F)
## X_1         1   606.2   606.2   8.6467  0.003484 **
## X_2         1  2633.9  2633.9  37.5671 2.280e-09 ***
## X_3         1 26281.5 26281.5 374.8474 < 2.2e-16 ***
## X_4         1  1711.3  1711.3  24.4074 1.187e-06 ***
## X_5         1  2790.6  2790.6  39.8024 8.094e-10 ***
## X_6         1    24.4    24.4   0.3478  0.555728
## Residuals 367 25731.3    70.1
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
xres <- resid(model)
xsq <- xres ^ 2
res.lm <- lm(xsq ~ X_1 + X_2 + X_3 + as_factor(X_4) + X_5)
anova(res.lm)
```

```
## Analysis of Variance Table
##
## Response: xsq
##                 Df   Sum Sq Mean Sq F value Pr(>F)
## X_1              1   175138  175138  1.9614 0.1622
## X_2              1    38896   38896  0.4356 0.5097
## X_3              1   171905  171905  1.9252 0.1661
## as_factor(X_4)  10   824468   82447  0.9233 0.5116
## X_5              1    99121   99121  1.1101 0.2928
## Residuals      359 32056143   89293
```

```
Xo <- ((374^2)/2) * (32056143/(25731.3^2))
Xo
```

```
## [1] 3386.11
```

```
X_crit <- qchisq(0.95,359)
X_crit
```

```
## [1] 404.1821
```

Since Xo is greater than X-critical, we reject the null hypothesis and as a result, the assumption of constant variance is not valid.

## 5.3 REMEDIAL MEASURES

There are various transformations to correct non constant variance. I used the log transformation on the target values.

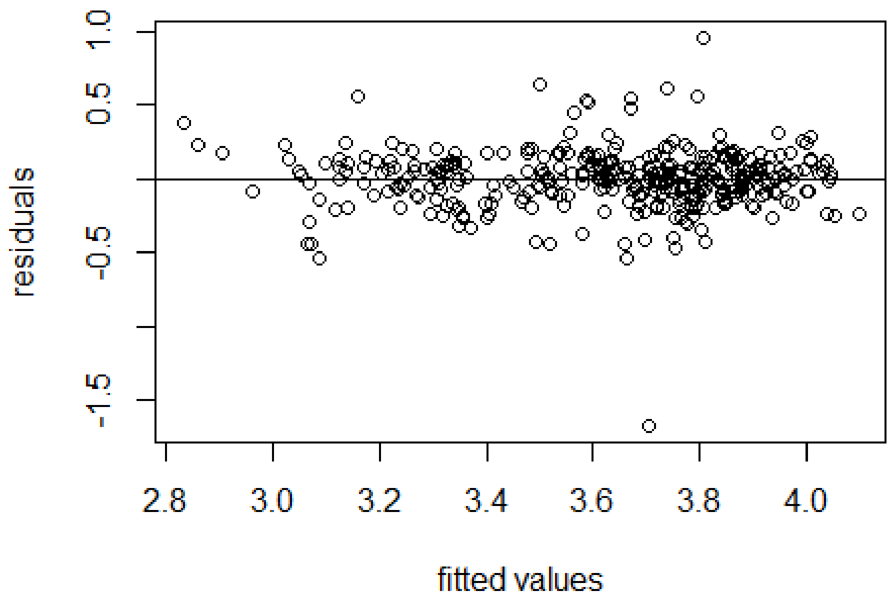
```
model1 <- lm(log(Y_1) ~ X_1 + X_2 + X_3 + X_4 + X_5 + X_6, data = PD1)
w = ols_step_best_subset(model1)
w
```

```
##         Best Subsets Regression
## --------------------------------------
## Model Index    Predictors
## --------------------------------------
##      1         X_3
##      2         X_3 X_5
##      3         X_2 X_3 X_5
##      4         X_2 X_3 X_4 X_5
##      5         X_1 X_2 X_3 X_4 X_5
##      6         X_1 X_2 X_3 X_4 X_5 X_6
## --------------------------------------
```

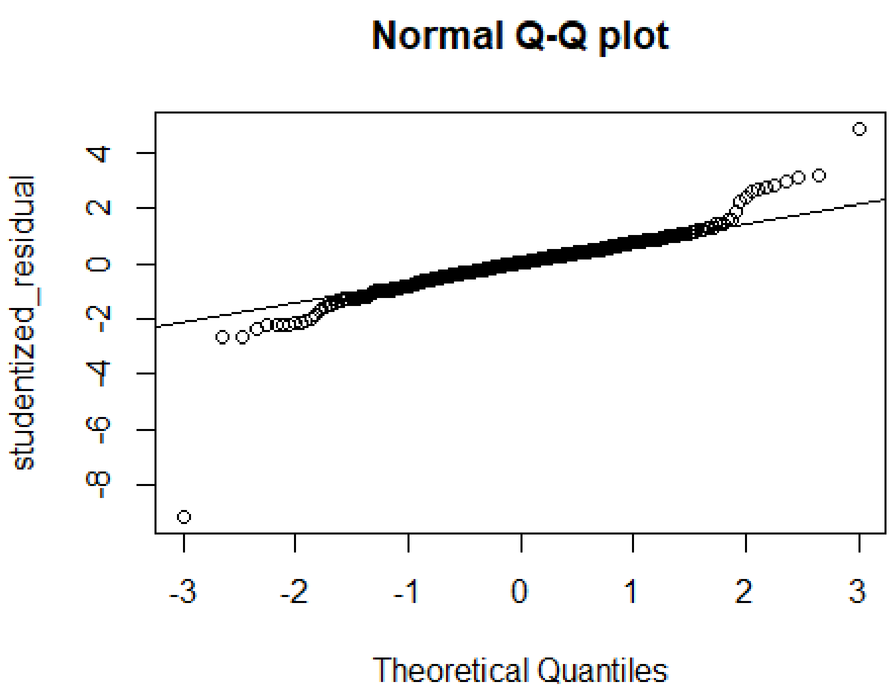
```
##
##                                                    Subsets Regression
Summary
## ----------------------------------------------------------------------------
----------------------------------------------------------
##                       Adj.        Pred
## Model    R-Square    R-Square    R-Square      C(p)         AIC
SBIC          SBC         MSEP        FPE        HSP       APC
## ----------------------------------------------------------------------------
----------------------------------------------------------
##   1        0.4531      0.4517      0.4471    169.7582      17.6794    -
1045.1037     29.4521     22.7137    0.0611    2e-04    0.5528
##   2        0.5275      0.5249      0.5184     98.3834     -34.9670    -
1097.6000    -19.2700     19.6791    0.0530    1e-04    0.4802
##   3        0.5969      0.5937      0.5865     31.8089     -92.4466    -
1154.2986    -72.8253     16.8310    0.0455    1e-04    0.4118
##   4        0.6197      0.6156      0.6074     11.3554    -112.1754    -
1173.5775    -88.6298     15.9242    0.0431    1e-04    0.3906
##   5        0.6272      0.6222      0.6134      5.9281    -117.6501    -
1178.8186    -90.1803     15.6516    0.0425    1e-04    0.3849
##   6        0.6282      0.6221      0.6127      7.0000    -116.5947    -
1177.6944    -85.2006     15.6548    0.0426    1e-04    0.3860
## ----------------------------------------------------------------------------
----------------------------------------------------------
## AIC: Akaike Information Criteria
##  SBIC: Sawa's Bayesian Information Criteria
##  SBC: Schwarz Bayesian Criteria
##  MSEP: Estimated error of prediction, assuming multivariate normality
##  FPE: Final Prediction Error
##  HSP: Hocking's Sp
##  APC: Amemiya Prediction Criteria
```

After transformation, the best model based on the best subset regression method was still model 5 containing X_1, X_2, X_3, X_4, and X_5. The model diagnostics plot for this fit is shown below.

```
model1_new <- lm(log(Y_1) ~ X_1 + X_2 + X_3 + as_factor(X_4) + X_5)
plot(fitted(model1_new), resid(model1_new), xlab = "fitted values", ylab =
"residuals", abline(0,0))
```




```
qqnorm(y = studres(model1_new), main = "Normal Q-Q plot", xlab = "Theoretical
Quantiles", ylab = "studentized_residual", plot.it = TRUE)
qqline(y= studres(model1_new), distribution = qnorm)
```



- The model diagnostics plot for this fit, shown above indicates that the model assumptions are valid.
- The box plot clearly shows the presence of outliers in the data with its median close to zero.
- In terms of normality, the points on the normal probability plot fall approximately on a straight line aside from the outlying points.

```
anova(model1_new)

## Analysis of Variance Table
##
## Response: log(Y_1)
##                 Df  Sum Sq Mean Sq  F value    Pr(>F)
## X_1              1  0.2864  0.2864   6.7247  0.009898 **
## X_2              1  1.5802  1.5802  37.0974 2.893e-09 ***
## X_3              1 19.7341 19.7341 463.2857 < 2.2e-16 ***
## as_factor(X_4)  10  1.6804  0.1680   3.9450 4.045e-05 ***
## X_5              1  2.7382  2.7382  64.2819 1.536e-14 ***
## Residuals      359 15.2920  0.0426
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

summary(model1_new)

##
## Call:
## lm(formula = log(Y_1) ~ X_1 + X_2 + X_3 + as_factor(X_4) + X_5)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.67583 -0.09211  0.00821  0.09916  0.95814
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -4.611e+02  8.307e+01  -5.551 5.54e-08 ***
## X_1               1.038e-01  3.875e-02   2.678 0.007752 **
## X_2              -7.563e-03  9.620e-04  -7.862 4.45e-14 ***
## X_3              -2.318e-04  2.433e-05  -9.526  < 2e-16 ***
## as_factor(X_4)1   6.787e-02  4.852e-02   1.399 0.162757
## as_factor(X_4)2   4.769e-02  5.497e-02   0.868 0.386230
## as_factor(X_4)3   8.910e-02  4.740e-02   1.880 0.060983 .
## as_factor(X_4)4   8.409e-02  5.171e-02   1.626 0.104813
## as_factor(X_4)5   1.307e-01  4.608e-02   2.835 0.004836 **
## as_factor(X_4)6   1.585e-01  5.189e-02   3.054 0.002428 **
## as_factor(X_4)7   1.575e-01  5.391e-02   2.921 0.003712 **
## as_factor(X_4)8   2.071e-01  5.447e-02   3.803 0.000168 ***
## as_factor(X_4)9   2.138e-01  5.873e-02   3.640 0.000313 ***
## as_factor(X_4)10  1.692e-01  8.055e-02   2.100 0.036421 *
## X_5               1.025e+01  1.279e+00   8.018 1.54e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 0.2064 on 359 degrees of freedom
## Multiple R-squared:  0.6298, Adjusted R-squared:  0.6154
## F-statistic: 43.63 on 14 and 359 DF,  p-value: < 2.2e-16
```

However, the ANOVA table shown after the plots show that X4 with 1,2,3 and 4 convenient houses should be dropped from the model.

selected model:

Y = -461.1 + 0.1038X2 - 0.00756X3 - 0.00023X3 + 0.1307(5 stores) + 0.1585(6 Stores) + 0.1575(7 Stores) + 0.2(8 stores) + 0.213(9 Stores) + 0.169(10 stores) + 10.25X5

## 6.0 CONCLUSION

We started with 6 predictor variables in the original data set. The best subset regression selection took out longitude(X6) from the set of predictors to be our best model. Here are the significant effects:

Date of purchase: The date of purchase does have a positive linear relationship with house price. On average, the house price will be 0.103 times higher for every additional increase in in date the house was bought.

House age: The age of the house has a negative linear relationship with the house price. On average, the house price will be 0.0076 times lower for every additional increase in the age of the house.

MRT station proximity: The house price will be 0.00023 times lower for any increase in the Proximity to an MRT station.

Latitude: A measure of how far north a house is. The house price will be 10.25 times higher for every additional increase in the latitude of the house.