

Data Analysis Project

Arijit Naskar & Spandan Ghoshal

Introduction

In this project, we analyze the “House Sales in King County, USA” dataset. This dataset contains house sale prices for King County, which includes Seattle during the time period May 2014 and May 2015. The dataset observations on 21613 many observations on the variables :-

price : Price is prediction target

bedrooms : Number of bedrooms

bathrooms : Number of bathrooms

sqft_living : Square footage of the home

sqft_lot : Square footage of the lot

floors : Total floors (levels) in house

waterfront : House which has a view to a waterfront

view : Has been viewed

condition : How good the condition is overall

grade : overall grade given to the housing unit, based on King County grading system

sqft_above : Square footage of house apart from basement

sqft_basement : Square footage of the basement

yr_built : Built Year

yr_renovated : Year when house was renovated

We applied various statistical tools to predict the price of a house. We divided the data analysis into two parts. Firstly we performed the exploratory analysis where we analyzed the data by drawing suitable diagrams to depict some dependence of price on different predictors and then made some crucial observations from there. Finally, to make conclusions based on statistical tools i.e testing of hypothesis, inference, estimation and various other regression methods, we answered to some of the observations we made earlier.

Importing the dataset

We import the dataset using R as follows :-

```
Data_House = read.table("E:\\Dekstop\\Project_Data_analysis_IITK\\kc_house_data_to_be_improved.csv")
head(Data_House)

  price bedrooms bathrooms sqft_living sqft_lot floors waterfront view
1 221900         3      1.00     1180     5650       1          0    0
2 538000         3      2.25     2570     7242       2          0    0
3 180000         2      1.00      770    10000       1          0    0
4 604000         4      3.00     1960     5000       1          0    0
5 510000         3      2.00     1680     8080       1          0    0
6 1230000        4      4.50     5420    101930       1          0    0

  condition grade sqft_above sqft_basement yr_built yr_renovated
1           3     7            1180             0     1955              0
2           3     7            2170             400    1951            1991
3           3     6            770              0     1933              0
4           5     7            1050             910    1965              0
5           3     8            1680             0     1987              0
6           3    11            3890            1530    2001              0
```

Checking presence of missing observations

We use the function `is.na()` to detect whether there is any missing observation in the given dataset or not as :-

```
sum(is.na(Data_House))

[1] 0
```

since there is no such missing observation, hence, we don't need to apply any data imputation techniques here and it makes the analysis little simpler.

Data type of each predictors

First we check the names of each and every variable present in the dataset as :-

```
vec_names = names(Data_House)
vec_names

[1] "price"          "bedrooms"        "bathrooms"       "sqft_living"
[5] "sqft_lot"        "floors"          "waterfront"      "view"
[9] "condition"       "grade"           "sqft_above"      "sqft_basement"
[13] "yr_built"        "yr_renovated"
```

To get an idea about each of them, we use the `summary()` function :-

```

data_summary = summary(Data_House)
data_summary

      price      bedrooms      bathrooms      sqft_living
Min.    : 75000   Min.    : 0.000   Min.    :0.000   Min.    : 290
1st Qu.: 321950  1st Qu.: 3.000   1st Qu.:1.750   1st Qu.: 1427
Median  : 450000  Median  : 3.000   Median  :2.250   Median  : 1910
Mean    : 540182  Mean    : 3.371   Mean    :2.115   Mean    : 2080
3rd Qu.: 645000  3rd Qu.: 4.000   3rd Qu.:2.500   3rd Qu.: 2550
Max.    :7700000  Max.    :33.000   Max.    :8.000   Max.    :13540

      sqft_lot      floors      waterfront      view
Min.    : 520     Min.    :1.000   Min.    :0.000000   Min.    :0.0000
1st Qu.: 5040    1st Qu.:1.000   1st Qu.:0.000000   1st Qu.:0.0000
Median  : 7618    Median  :1.500   Median  :0.000000   Median  :0.0000
Mean    : 15107   Mean    :1.494   Mean    :0.007542   Mean    :0.2343
3rd Qu.: 10688   3rd Qu.:2.000   3rd Qu.:0.000000   3rd Qu.:0.0000
Max.    :1651359  Max.    :3.500   Max.    :1.000000   Max.    :4.0000

      condition      grade      sqft_above      sqft_basement
Min.    :1.000     Min.    : 1.000   Min.    : 290     Min.    : 0.0
1st Qu.:3.000     1st Qu.: 7.000   1st Qu.:1190    1st Qu.: 0.0
Median  :3.000     Median  : 7.000   Median  :1560    Median  : 0.0
Mean    :3.409     Mean    : 7.657   Mean    :1788    Mean    : 291.5
3rd Qu.:4.000     3rd Qu.: 8.000   3rd Qu.:2210    3rd Qu.: 560.0
Max.    :5.000     Max.    :13.000   Max.    :9410    Max.    :4820.0

      yr_built      yr_renovated
Min.    :1900      Min.    : 0.0
1st Qu.:1951      1st Qu.: 0.0
Median  :1975      Median  : 0.0
Mean    :1971      Mean    : 84.4
3rd Qu.:1997      3rd Qu.: 0.0
Max.    :2015      Max.    :2015.0

```

`str()` function also gives us some idea about the nature of each variable.

```

str(Data_House)

'data.frame': 21613 obs. of  14 variables:
 $ price       : num  221900 538000 180000 604000 510000 ...
 $ bedrooms    : int  3 3 2 4 3 4 3 3 3 3 ...
 $ bathrooms   : num  1 2.25 1 3 2 4.5 2.25 1.5 1 2.5 ...
 $ sqft_living : int  1180 2570 770 1960 1680 5420 1715 1060 1780 1890 ...
 $ sqft_lot    : int  5650 7242 10000 5000 8080 101930 6819 9711 7470 6560 ...
 $ floors      : num  1 2 1 1 1 1 2 1 1 2 ...
 $ waterfront  : int  0 0 0 0 0 0 0 0 0 ...
 $ view         : int  0 0 0 0 0 0 0 0 0 ...
 $ condition   : int  3 3 3 5 3 3 3 3 3 3 ...
 $ grade        : int  7 7 6 7 8 11 7 7 7 7 ...

```

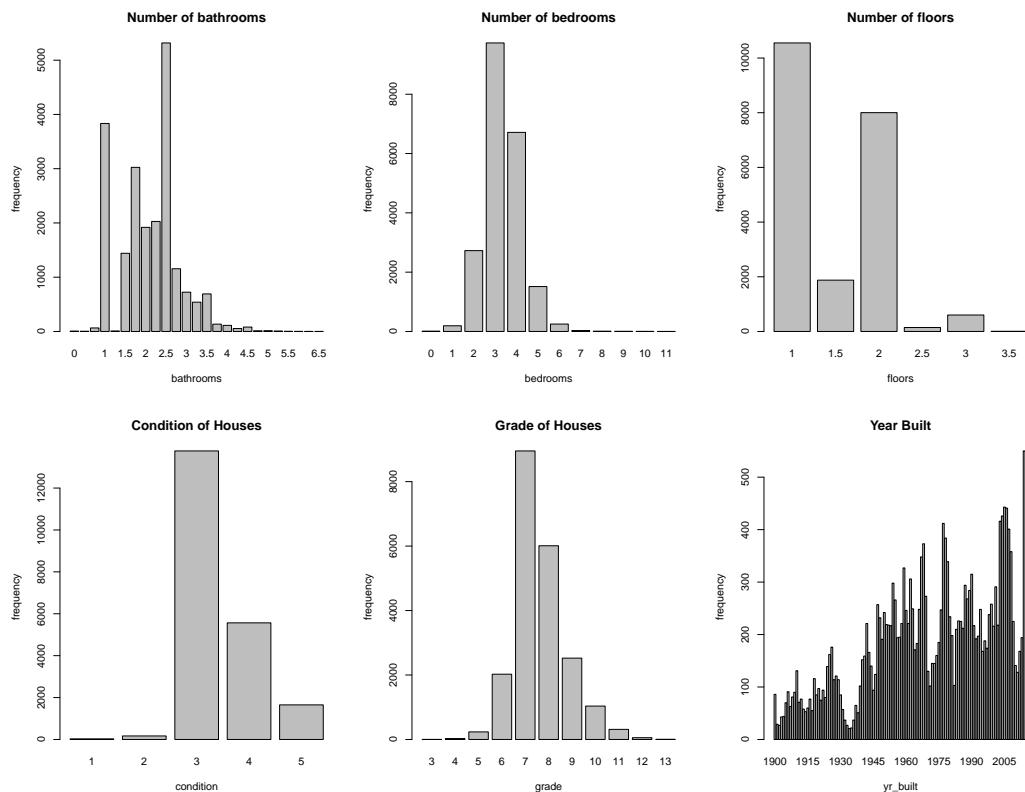
```
$ sqft_above : int 1180 2170 770 1050 1680 3890 1715 1060 1050 1890 ...
$ sqft_basement: int 0 400 0 910 0 1530 0 0 730 0 ...
$ yr_built : int 1955 1951 1933 1965 1987 2001 1995 1963 1960 2003 ...
$ yr_renovated : int 0 1991 0 0 0 0 0 0 0 0 ...
```

Observation :-

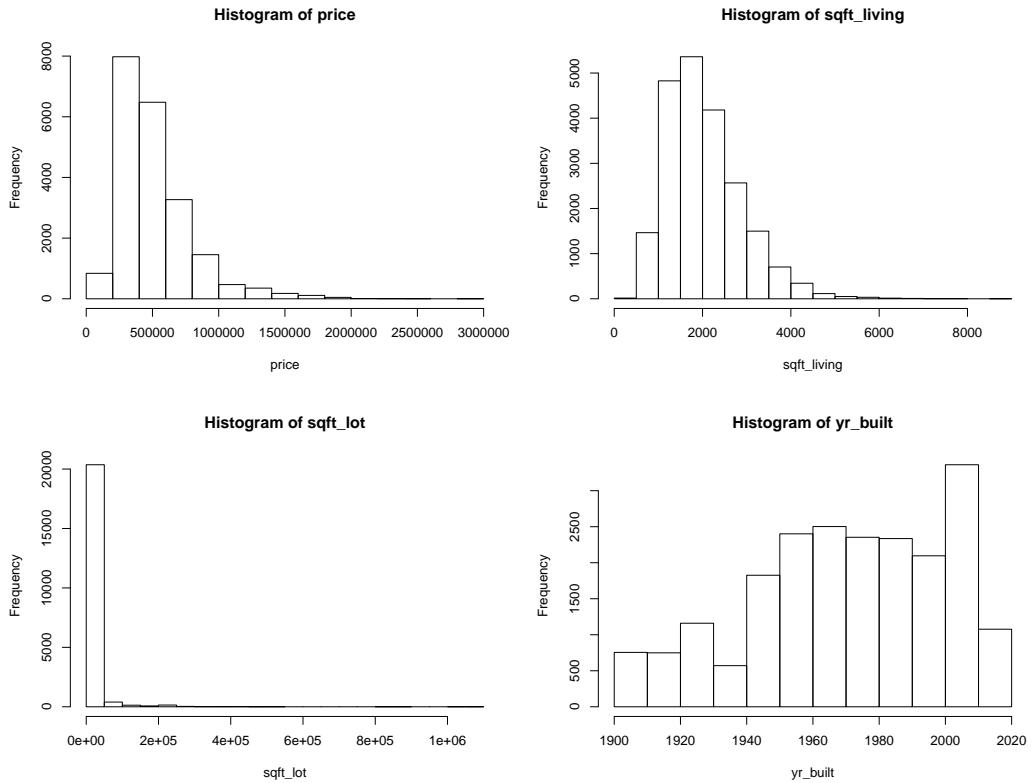
From the summary outputs for different variable, we can see that there might be potential outliers in the data as the ranges for some of the variables vary tremendously so checking presence of any outliers will be of great help in this case.

Barplot for different integer valued variables

We plot the bar diagrams for different count datas :-

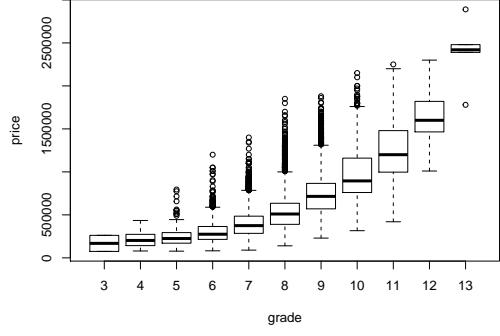
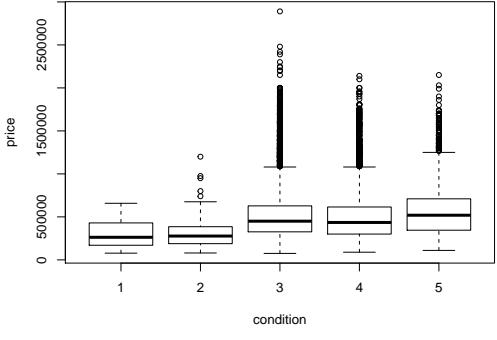
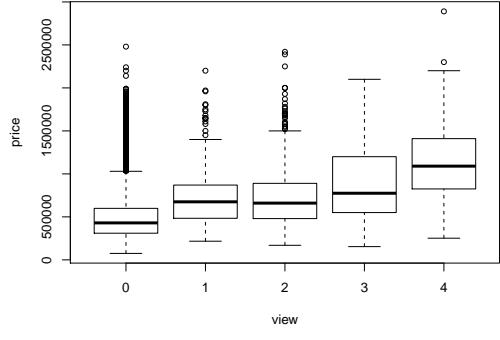
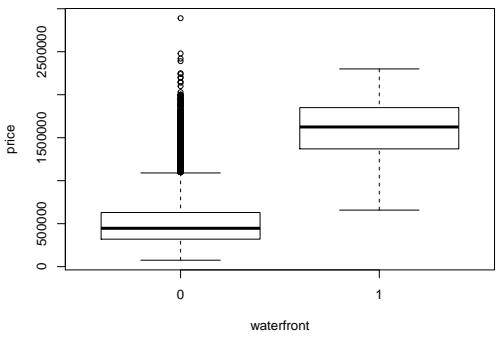
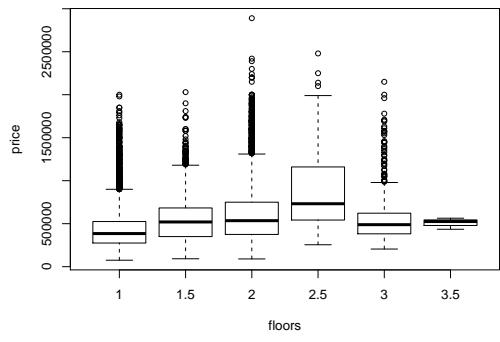
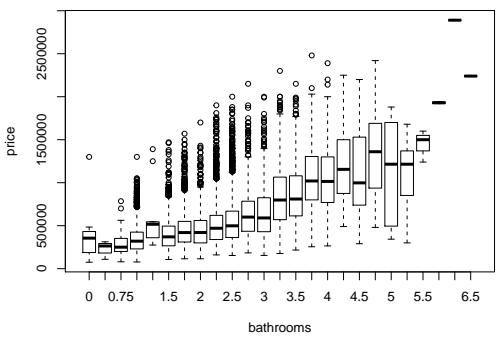
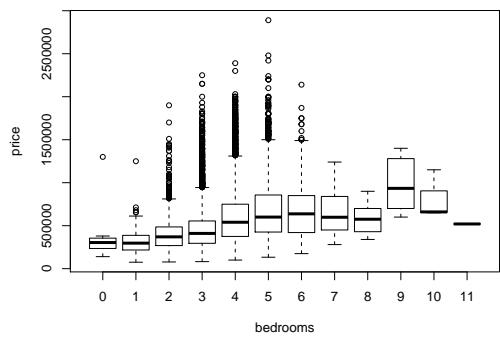


Histograms Plots



Boxplots

We plot the boxplots of price for different integer valued variables :-



Observation :-

From the boxplots, we can see that in some of the plots, the price increases with increasing value of the other covariate like number of bedrooms, bathrooms, waterfront, view, condition, grade etc. We would perform ANOVA test to detect significant differences in price for different values of these covariates.

Confirmatory Data Analysis

Loading all the packages required for analysis.

```
set.seed(1000)
library(readr)

Warning: package 'readr' was built under R version 3.6.3

library(splines)
library(glmnet)

Warning: package 'glmnet' was built under R version 3.6.3
Loading required package: Matrix
Loaded glmnet 4.1-1

library(boot)
library(leaps)

Warning: package 'leaps' was built under R version 3.6.3

library(tidyverse)

Warning: package 'tidyverse' was built under R version 3.6.3
- Attaching packages -----
  v ggplot2 3.3.5    v dplyr  1.0.6
  v tibble   3.1.0    v stringr 1.4.0
  v tidyrr   1.1.3    vforcats 0.5.1
  v purrrr  0.3.4
Warning: package 'tibble' was built under R version 3.6.3
Warning: package 'tidyrr' was built under R version 3.6.3
Warning: package 'purrr' was built under R version 3.6.3
Warning: package 'dplyr' was built under R version 3.6.3
Warning: package 'forcats' was built under R version 3.6.3
- Conflicts -----
  x tidyrr::expand() masks Matrix::expand()
  x dplyr::filter() masks stats::filter()
  x dplyr::lag()    masks stats::lag()
  x tidyrr::pack()  masks Matrix::pack()
  x tidyrr::unpack() masks Matrix::unpack()

library(caret)
```

```

Warning: package 'caret' was built under R version 3.6.3
Loading required package: lattice

Attaching package: 'lattice'
The following object is masked from 'package:boot':
  melanoma

Attaching package: 'caret'
The following object is masked from 'package:purrr':
  lift

library(ggplot2)
library(ggpubr)

Warning: package 'ggpubr' was built under R version 3.6.2
Loading required package: magrittr
Warning: package 'magrittr' was built under R version 3.6.3

Attaching package: 'magrittr'
The following object is masked from 'package:purrr':
  set_names
The following object is masked from 'package:tidyverse':
  extract

library(magrittr)

```

Outlier Treatment

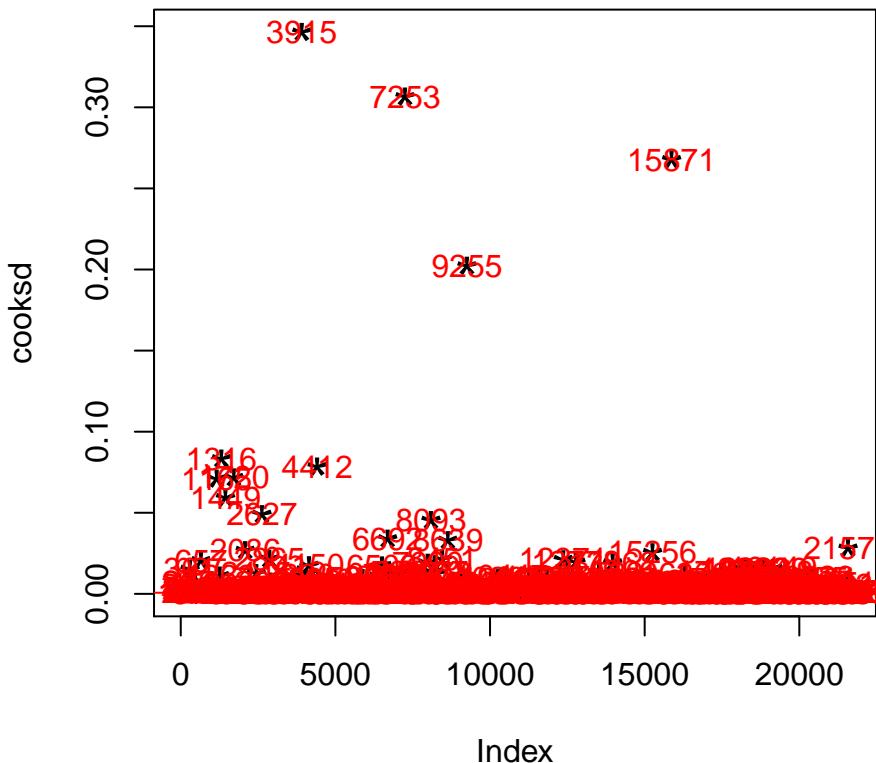
We use cook's distance for detecting any potential outlier in the data. We plot the cook's distance for each observation in a scatter plot w.r.t their indexes.

```

mod <- lm(price ~ ., data=Data_House)
cooksd <- cooks.distance(mod)
plot(cooksd, pch="*", cex=2, main="Influential Obs by Cooks distance") # plot cook's
abline(h = 4*mean(cooksd, na.rm=T), col="red") # add cutoff line
text(x=1:length(cooksd)+1, y=cooksd, labels=ifelse(cooksd>4*mean(cooksd, na.rm=T), nam

```

Influential Obs by Cooks distance



We detect potential by choosing those observations which are greater than 4 times average cook's distance. (some of the indexes of those outliers are given below)

```
ind_outlier = (cooksdist > 4 * mean(cooksdist, na.rm=T))
ind_out = which(ind_outlier == TRUE)
head(ind_out)

22 116 154 231 240 247
22 116 154 231 240 247
```

```
length(ind_out)

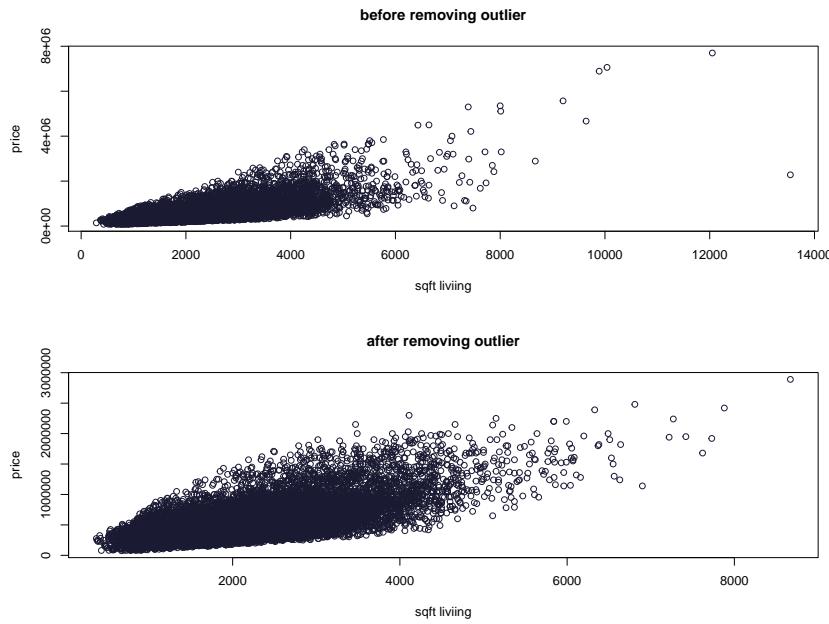
[1] 428

Data_House_1 = Data_House[-ind_out,]
dim(Data_House_1)

[1] 21185      14

attach(Data_House_1)
```

For comparison, we plot the scatter plots of price vs sqft living as shown :-



We should find the impact of the quantitative variables on the price of the houses separately.

Price vs Area of Basement

We obtain the impact of area of basement on the price of the houses.

```
lr=lm(price~sqft_basement)
summary(lr)

Call:
lm(formula = price ~ sqft_basement)

Residuals:
    Min      1Q      Median      3Q      Max  
-474073 -181361   -62612   111493  1958639 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 4.614e+05  2.213e+03 208.47 <2e-16 ***
sqft_basement 1.877e+02  4.342e+00  43.22 <2e-16 ***  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 268700 on 21183 degrees of freedom
Multiple R-squared:  0.08105, Adjusted R-squared:  0.081 
F-statistic: 1868 on 1 and 21183 DF,  p-value: < 2.2e-16
```

The linear model is,

$$y_i = a + bx_i + e_i$$

where, y_i = Price of i th house and x = area of basement of the i th house, $i = 1, 2, \dots, 21613$.

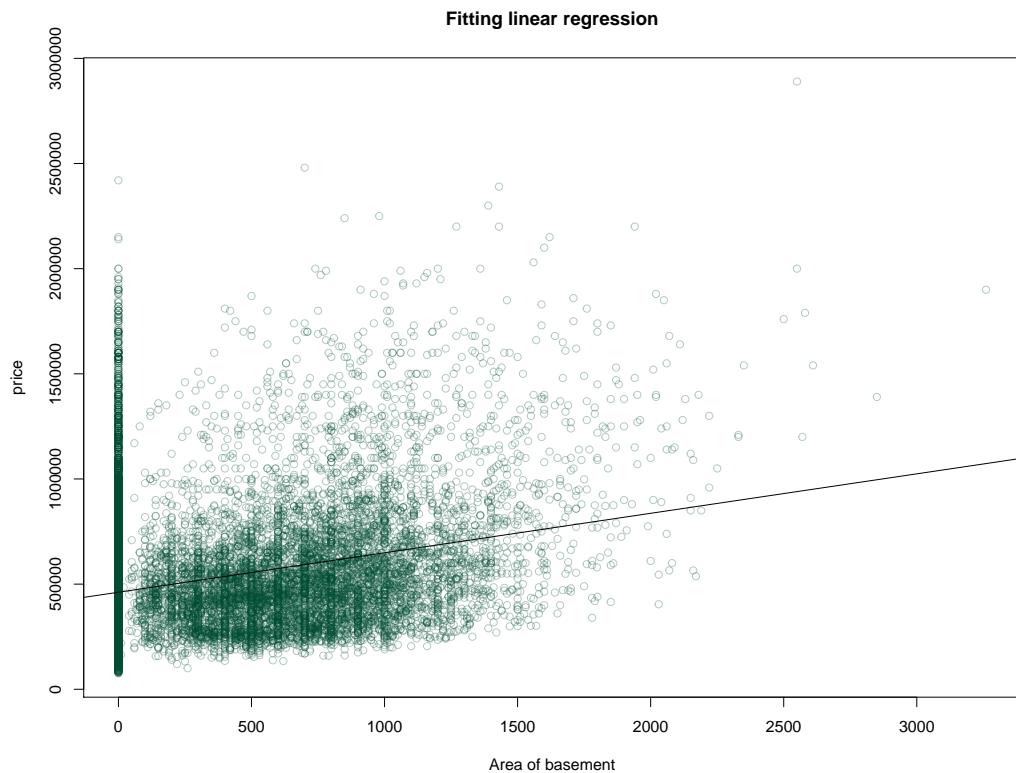
We set the null hypothesis,

$$H_0 : b = 0 \text{ vs } H_1 : b \neq 0$$

The p value of the hypothesis is less than 0.05.

Then we can conclude that **the area of basement of the houses have significant effect of the linear regression of price .**

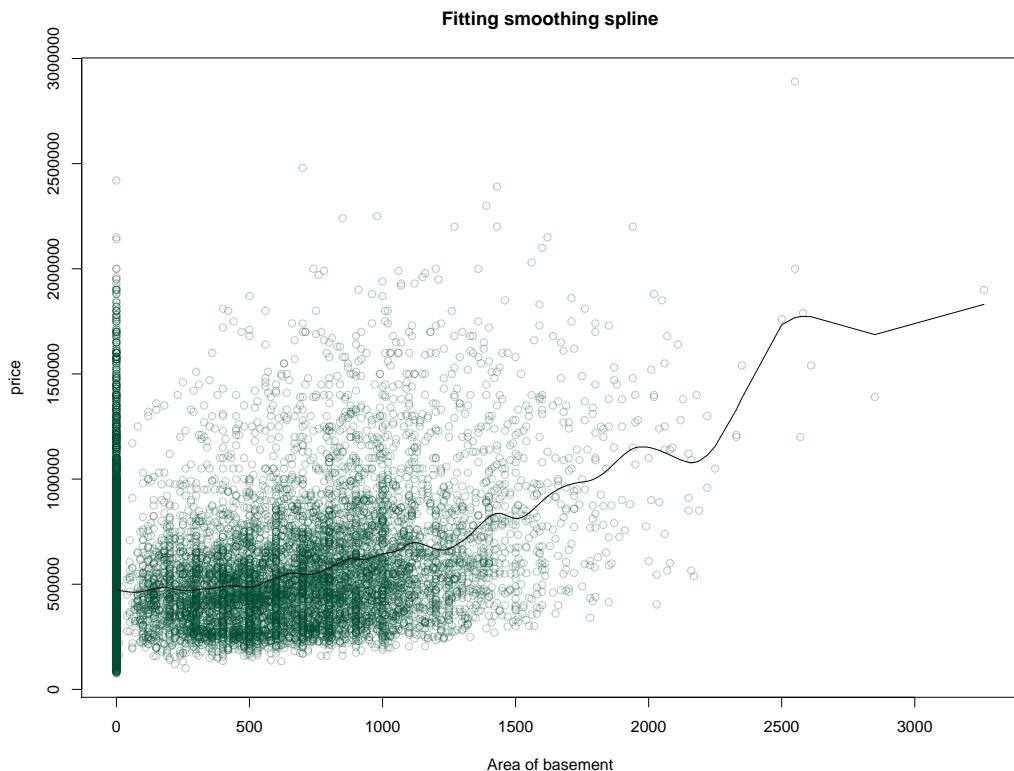
The fitted linear regression is as follows,



To get more precise relation between price and the area of the basements,we fitted both **Box kernel regression** and **Gaussian kernel regression**.



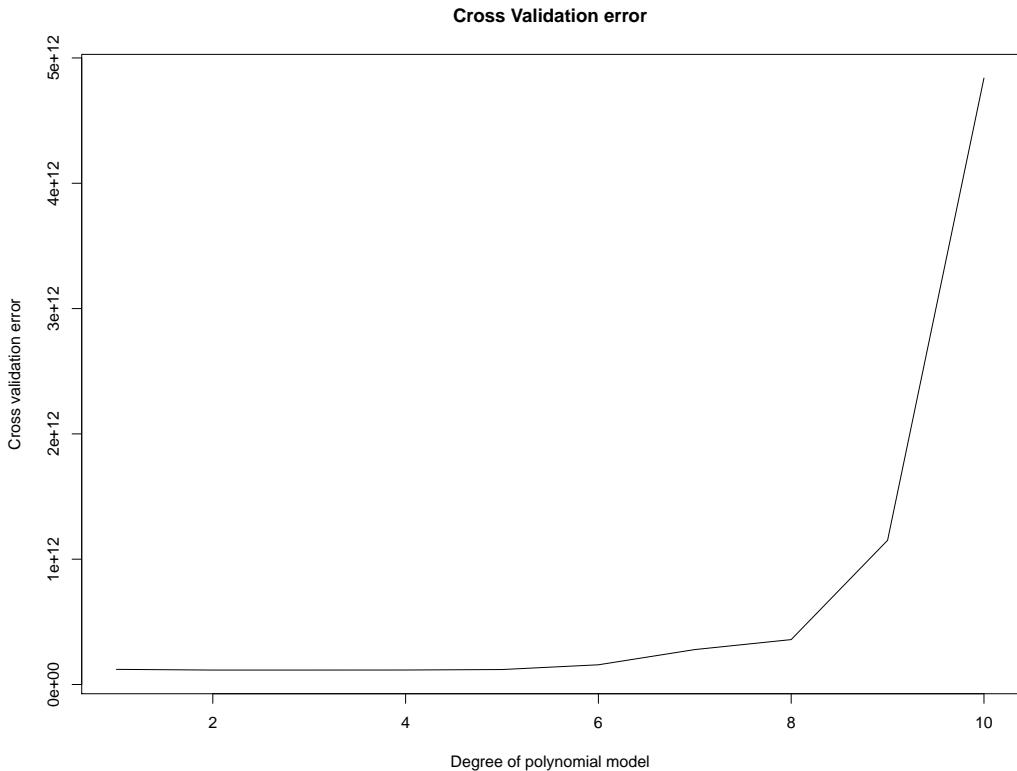
The fitted **spline regression** of the price on the area of basement using **smoothing spline**, is as follows,



We have the linear regression of the price on the area of basement. Further, we are interested in fitting a **polynomial regression** with appropriate degree.

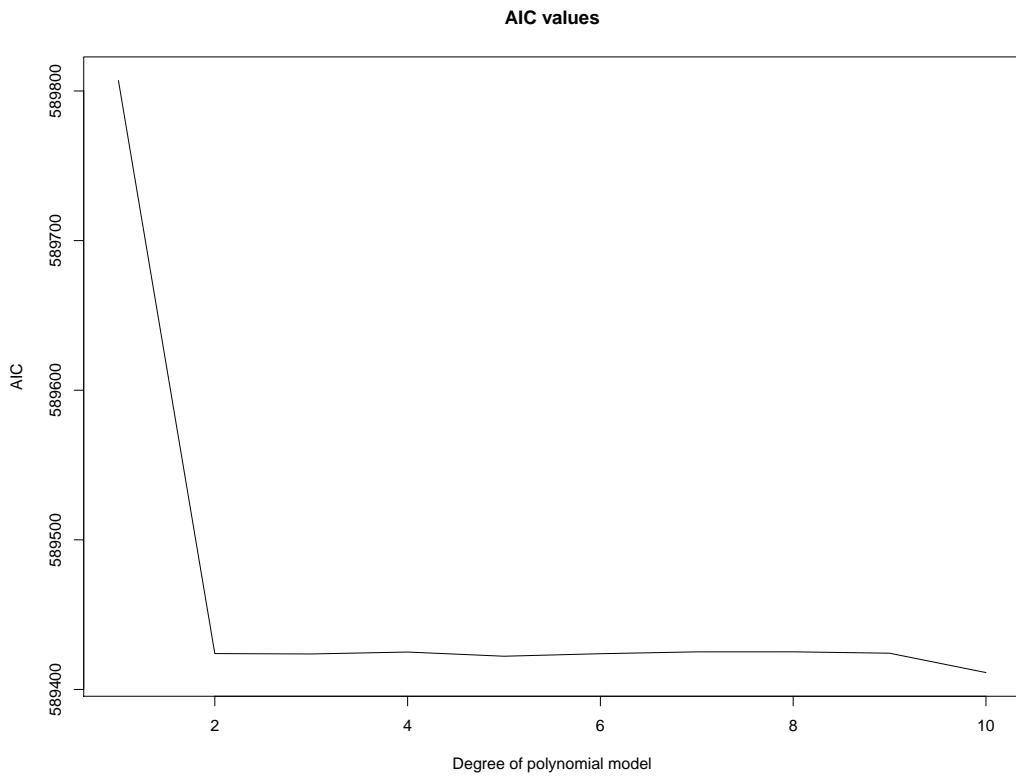
In order to determine the appropriate degree of the polynomial regression, we can fit the polynomial regression model with different degrees and then minimize the measure of optimism. We can obtain the measure of optimism using **Akaike information criterion(AIC)**, **Bayesian information criterion(BIC)** and **Cross-validation** method.

The value of **Cross validation error** of the different degrees of polynomial model is as follows,



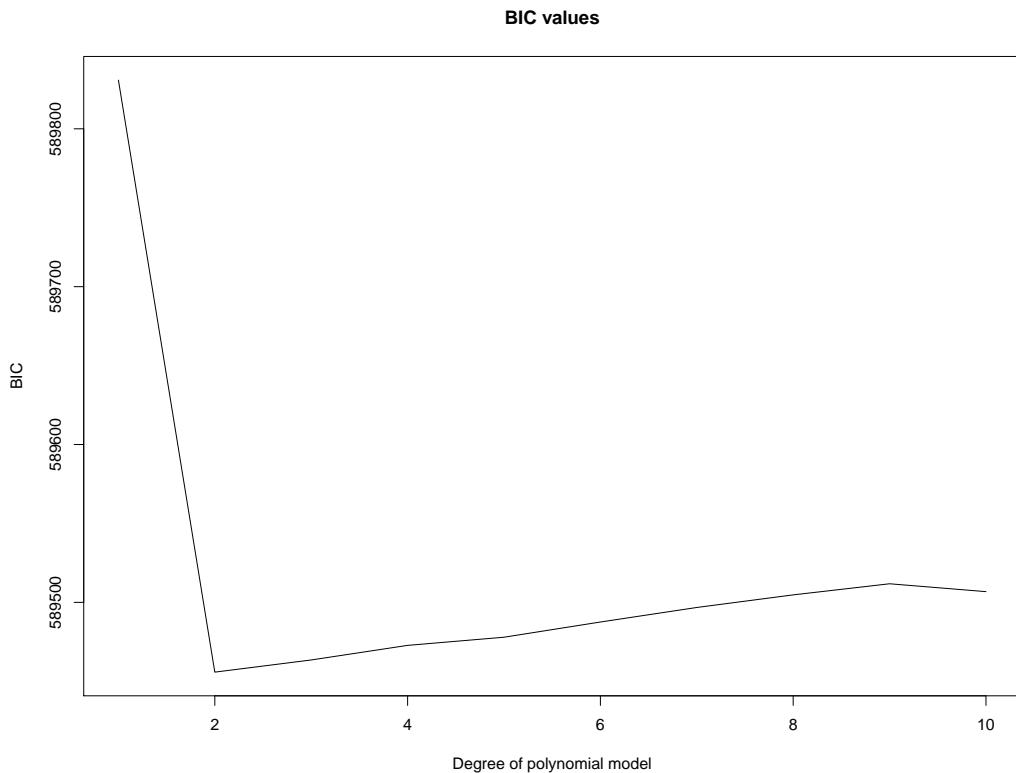
Cross Validation method suggest that we should fit a second degree **polynomial regression** of price on area of basement

The value of AIC of the different degrees of polynomial model is as follows,



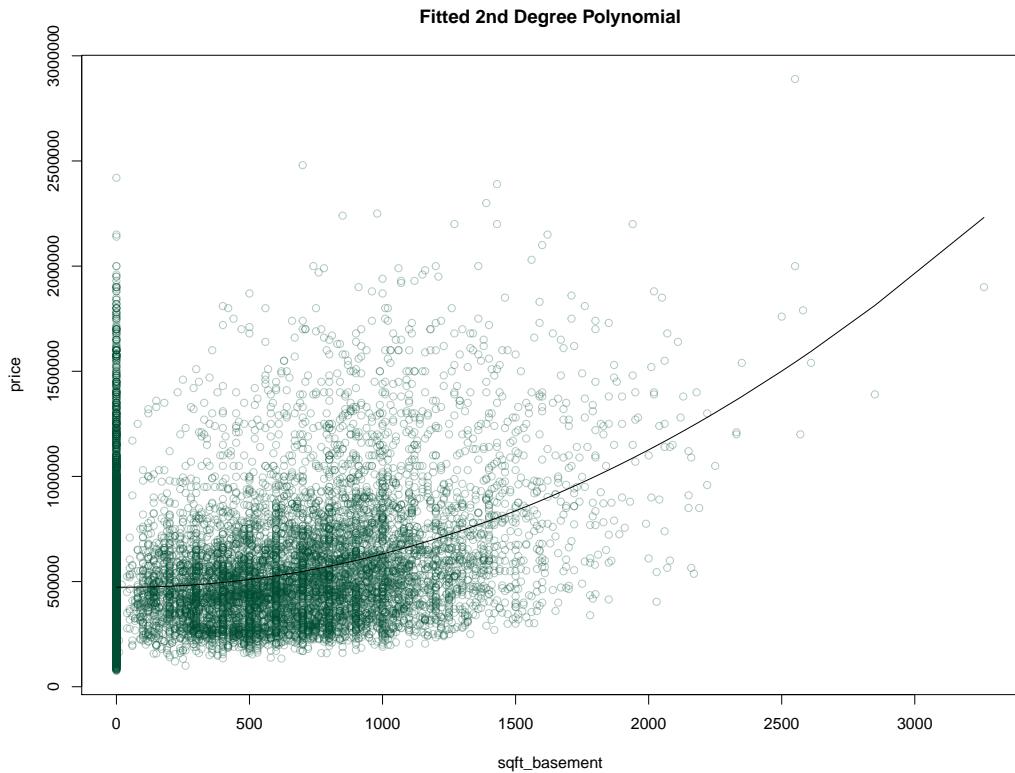
AIC method suggest that we should fit a ten degree **polynomial regression** of price on area of basement.

The value of **BIC** of the different degrees of polynomial model is as follows,



BIC method suggest that we should fit a second degree **polynomial regression** of price on area of basement.

Cross Validation method is the most accurate method among the given methods. Since, two of the methods suggest to fit a second degree polynomial hence, we should fit a second degree **polynomial regression** of price on area of basement.



Price vs Area Above

We obtain the impact of area of above on the price of the houses.

```
lr=lm(price~sqft_above)
summary(lr)

##
## Call:
## lm(formula = price ~ sqft_above)
##
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -573911 -151298 -32904  104577 1582194 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 1.406e+05  3.791e+03   37.08   <2e-16 ***
## sqft_above  2.122e+02  1.967e+00   107.91   <2e-16 ***
```

```

## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 225200 on 21183 degrees of freedom
## Multiple R-squared: 0.3547, Adjusted R-squared: 0.3547
## F-statistic: 1.164e+04 on 1 and 21183 DF, p-value: < 2.2e-16

```

The linear model is,

$$y_i = a + bx_i + e_i$$

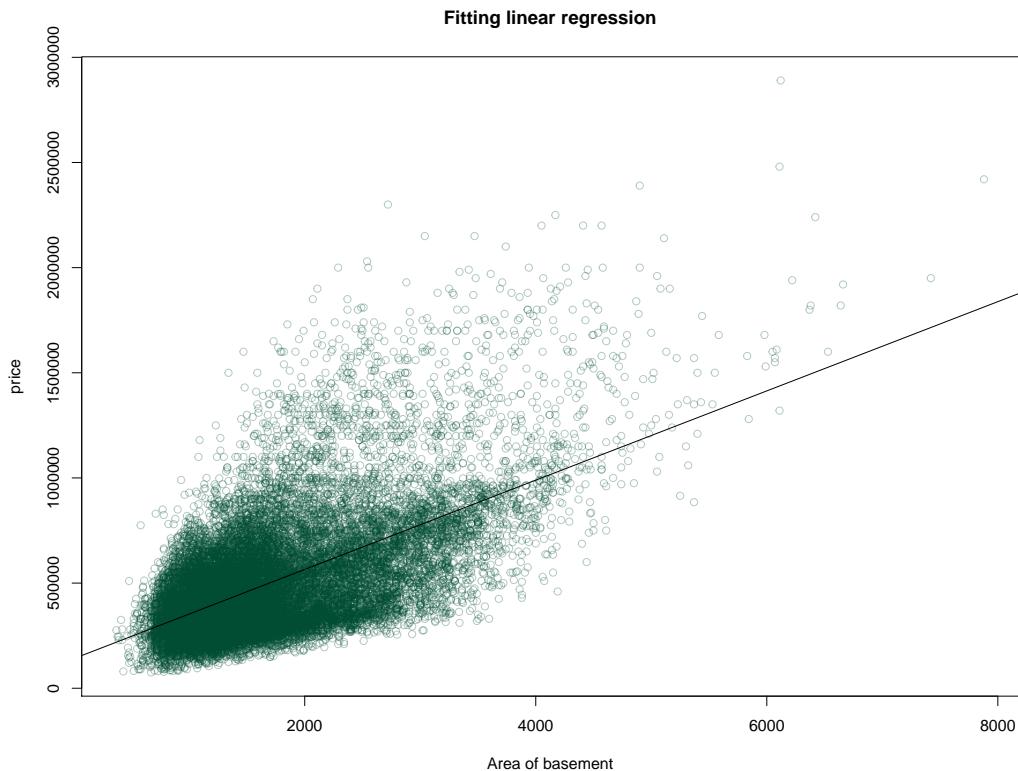
where, y_i = Price of i th house and x =area of above of the i th house, $i = 1, 2, \dots, 21613$. We set the null hypothesis,

$$H_0 : b = 0 \text{ vs } H_1 : b \neq 0$$

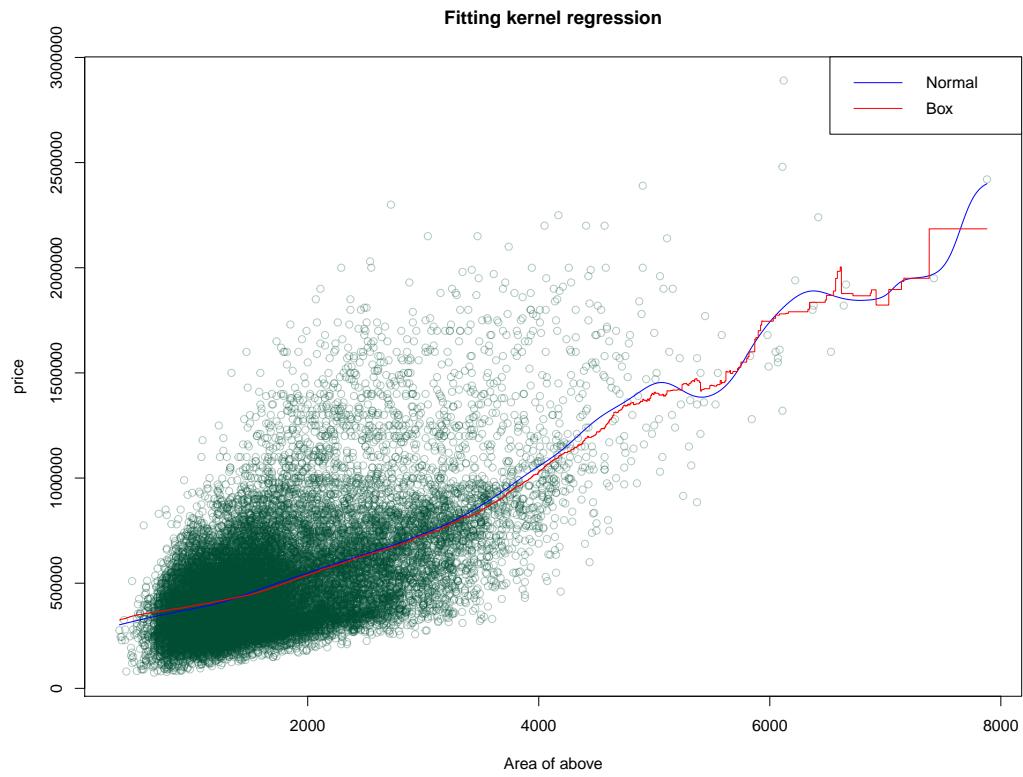
The p value of the hypothesis is less than 0.05.

Then we can conclude that **the area of above of the houses have significant effect of the linear regression of price**.

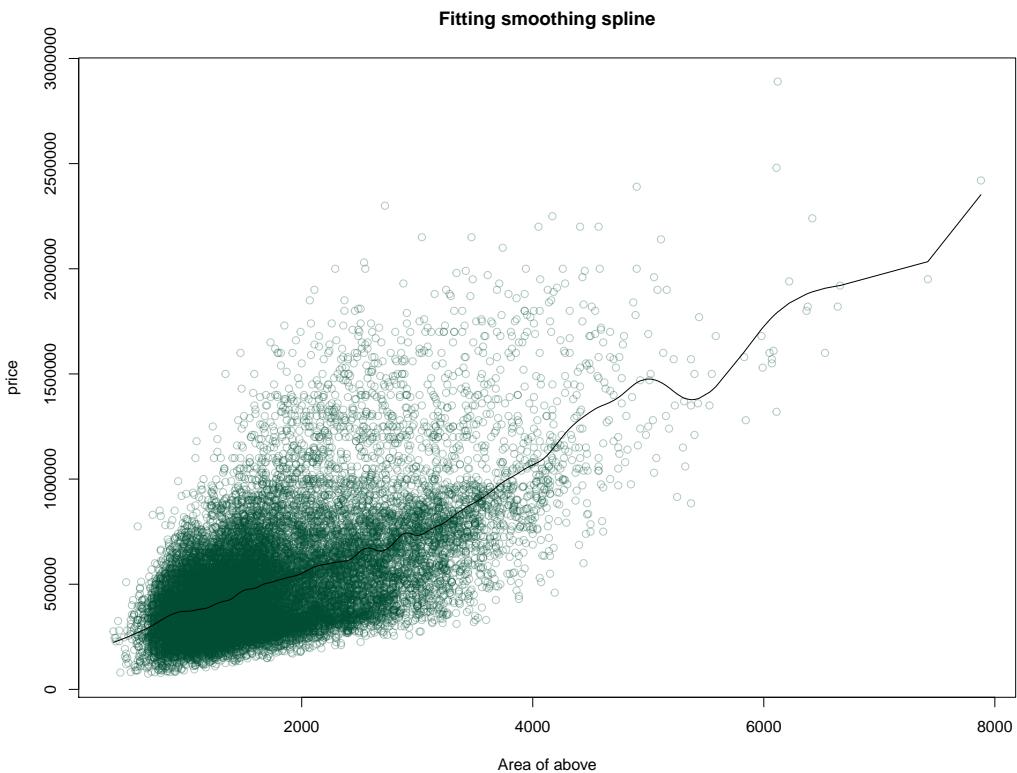
The fitted linear regression is as follows,



To get more precise relation between price and the area of the above,we fitted both **Box kernel regression** and **Gaussian kernel regression**.



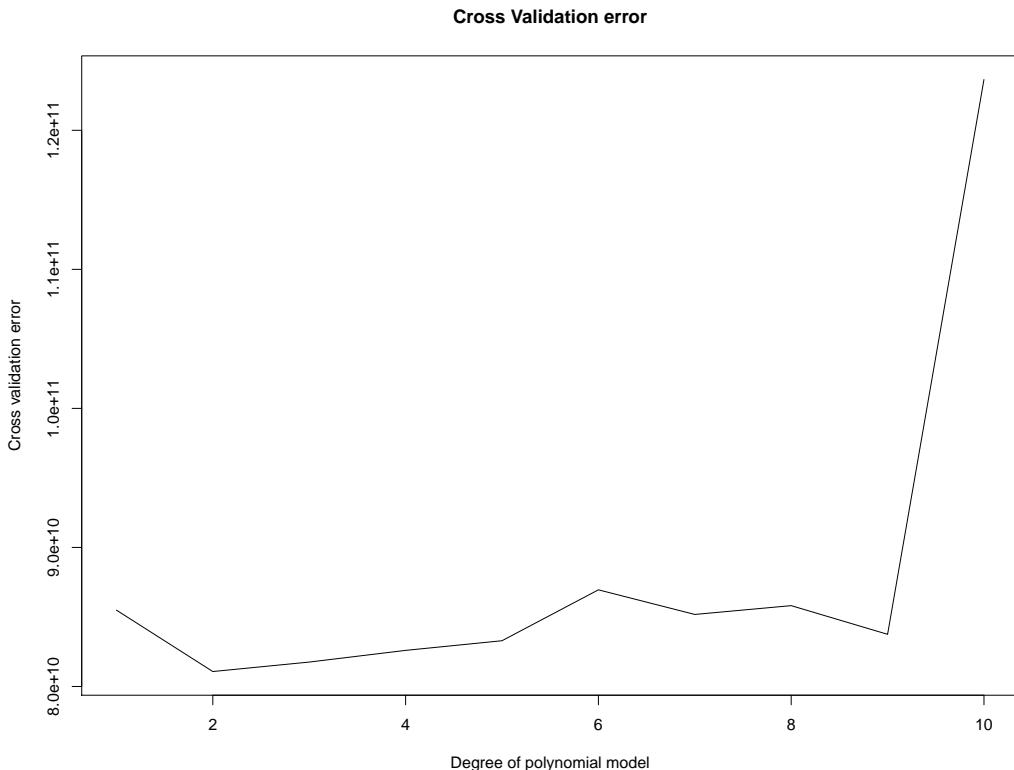
The fitted **spline regression** of the price on the area of the above using **smoothing spline**, is as follows,



We have the linear regression of the price on the area of above. Further, we are interested in fitting a **polynomial regression** with appropriate degree.

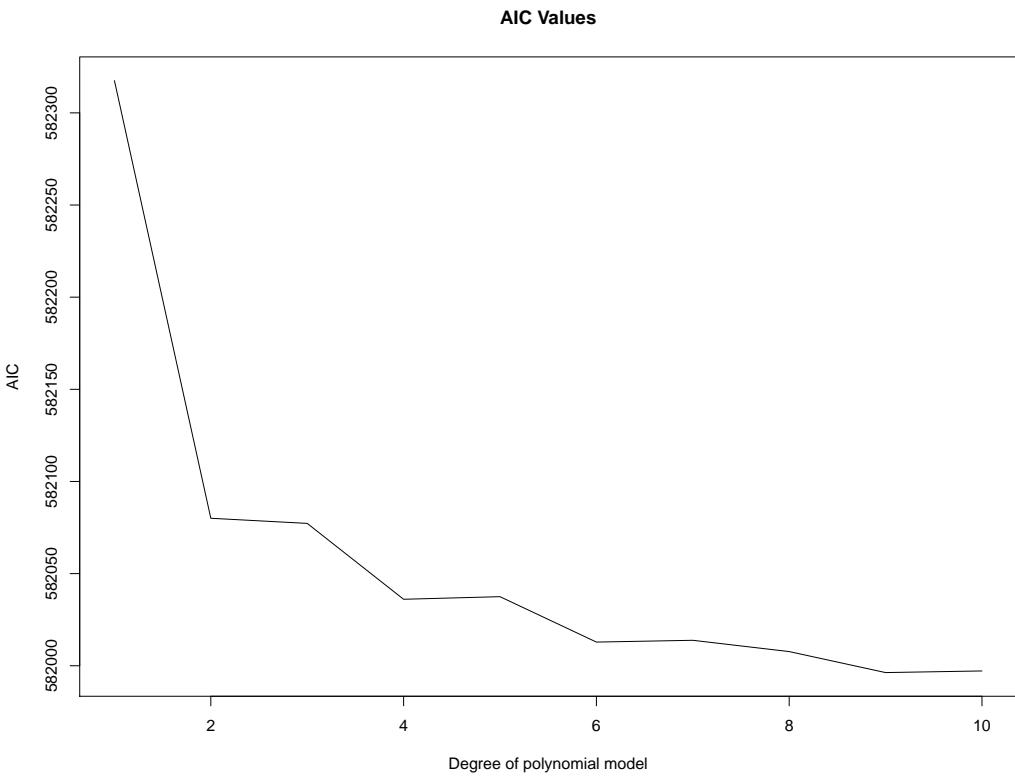
In order to determine the appropriate degree of the polynomial regression, we can fit the polynomial regression model with different degrees and then minimize the measure of optimism. We can obtain the measure of optimism using **Akaike information criterion(AIC)**, **Bayesian information criterion(BIC)** and **Cross-validation** method.

The value of **Cross validation error** of the different degrees of polynomial model is as follows,



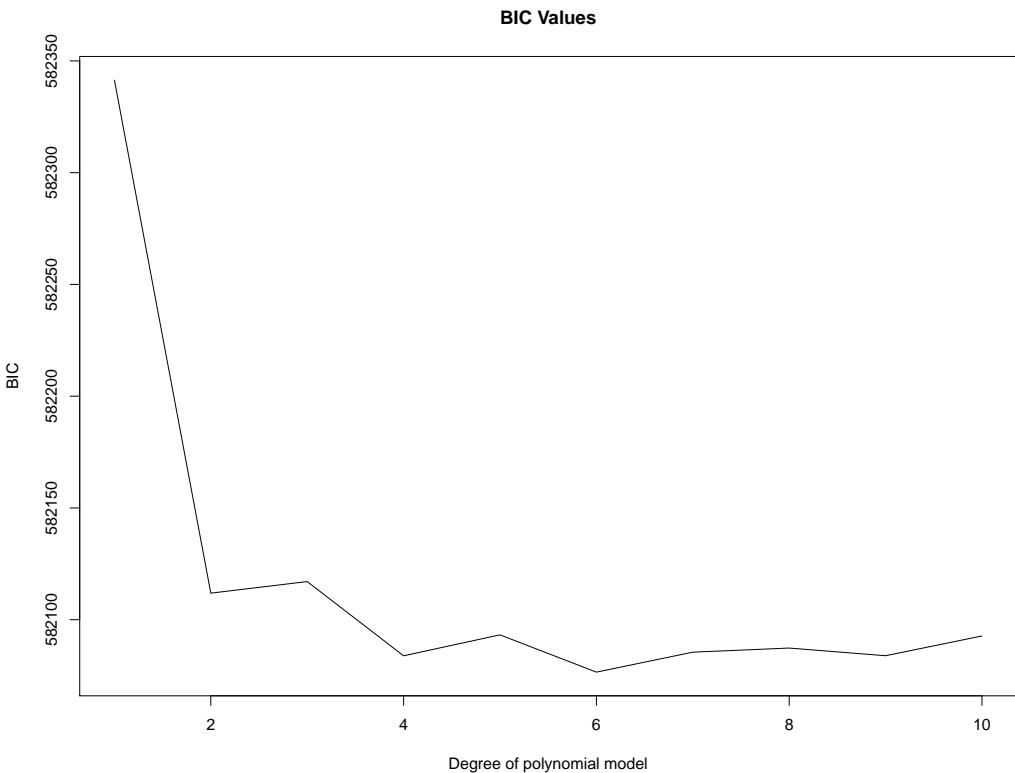
Cross Validation method suggest that we should fit a second degree **polynomial regression** of price on area of basement

The value of AIC of the different degrees of polynomial model is as follows,



AIC method suggest that we should fit a ten degree **polynomial regression** of price on area of basement.

The value of **BIC** of the different degrees of polynomial model is as follows,



BIC method suggest that we should fit a six degree **polynomial regression** of price on area of basement.

Cross Validation method is the most accurate method among the given methods. Thus, we should fit a two degree **polynomial regression** of price on area of basement.

Price vs Sqft Lot

We obtain the impact of area of lot on the price of the houses.

```
lr=lm(price~sqft_lot)
summary(lr)

##
## Call:
## lm(formula = price ~ sqft_lot)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -826007 -192869  -66823  116118 2335677
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 5.027e+05 2.078e+03 241.88 <2e-16 ***
## sqft_lot    8.060e-01 5.680e-02 14.19 <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 279000 on 21183 degrees of freedom
## Multiple R-squared:  0.009417, Adjusted R-squared:  0.00937
## F-statistic: 201.4 on 1 and 21183 DF, p-value: < 2.2e-16
```

The linear model is,

$$y_i = a + bx_i + e_i$$

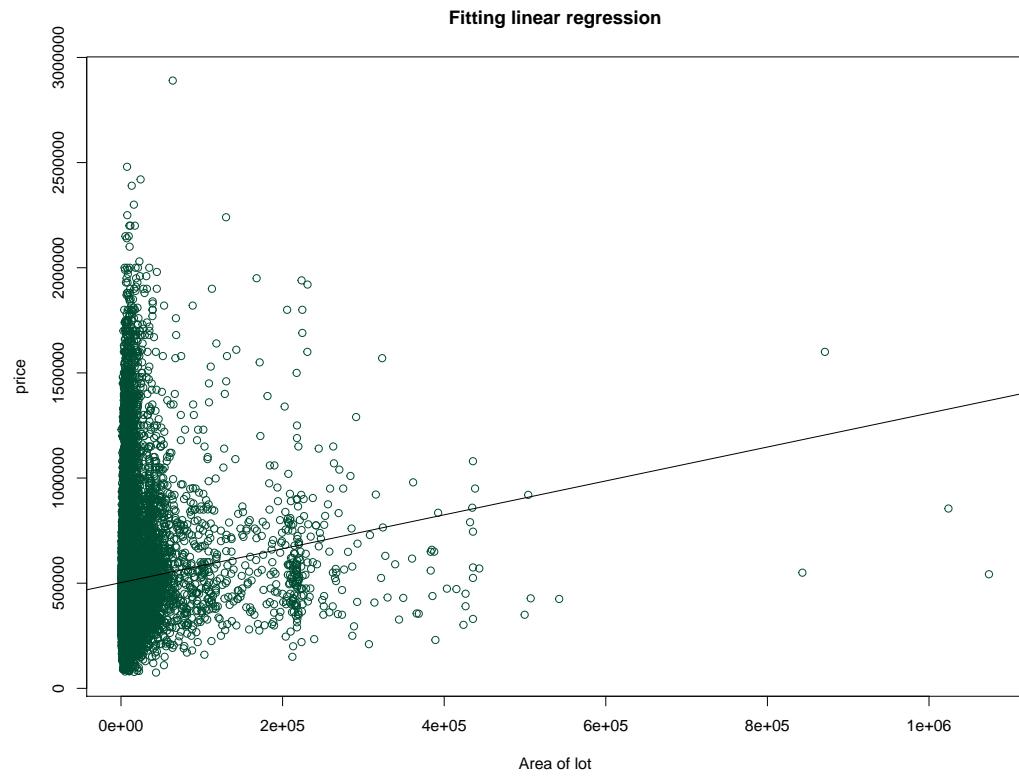
where, y_i = Price of i th house and x = area of lot of the i th house, $i = 1, 2, \dots, 21613$. We set the null hypothesis,

$$H_0 : b = 0 \text{ vs } H_1 : b \neq 0$$

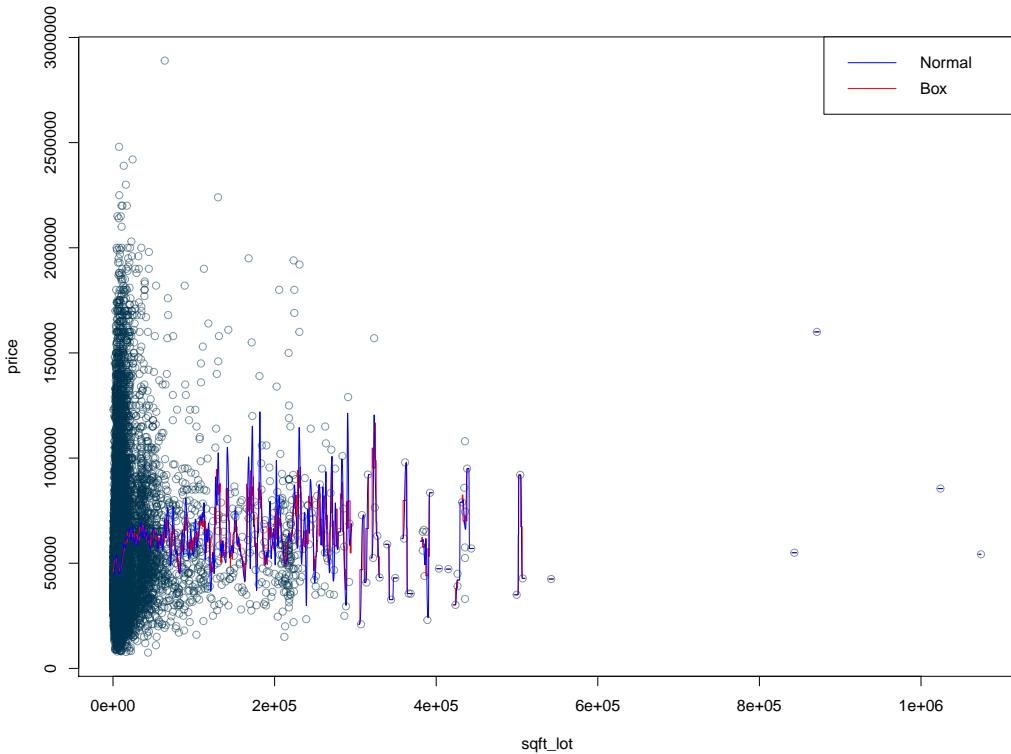
The p value of the hypothesis is less than 0.05.

Then we can conclude that **the area of lot of the houses have significant effect of the linear regression of price**.

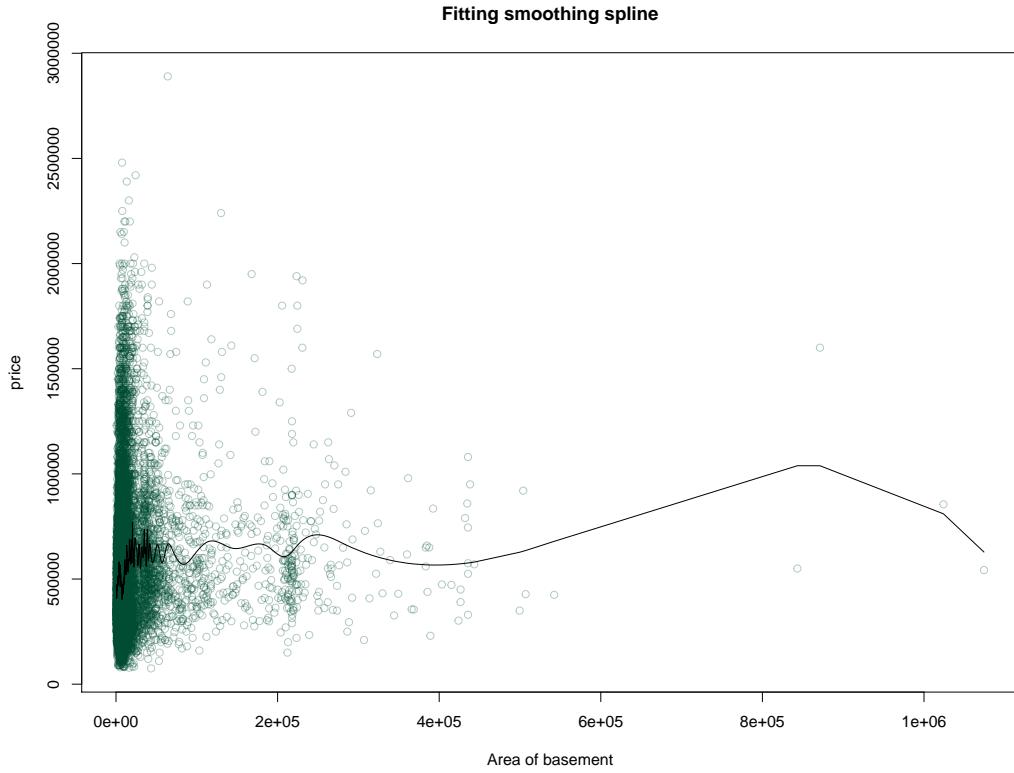
The fitted linear regression is as follows,



To get more precise relation between price and the area of the basements, we fitted both **Box kernel regression** and **Gaussian kernel regression**.



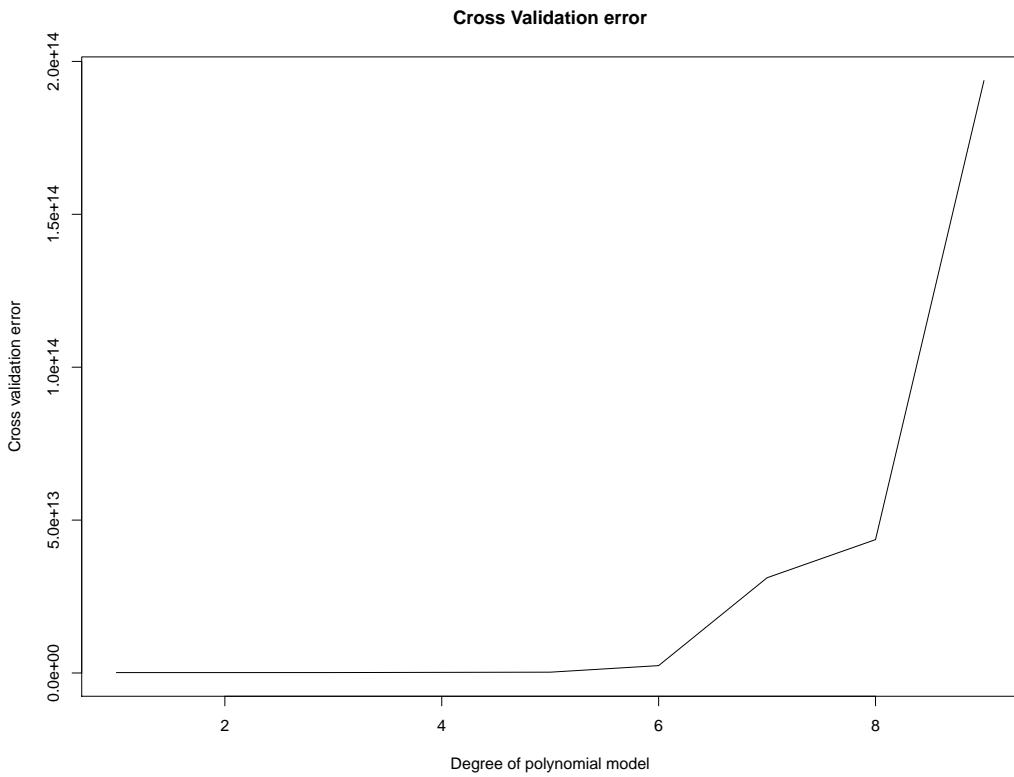
The fitted **spline regression** of the price on the area of basement using **smoothing spline**, is as follows,



We have the linear regression of the price on the area of basement.Further,we are interested in fitting a **polynomial regression** with appropriate degree.

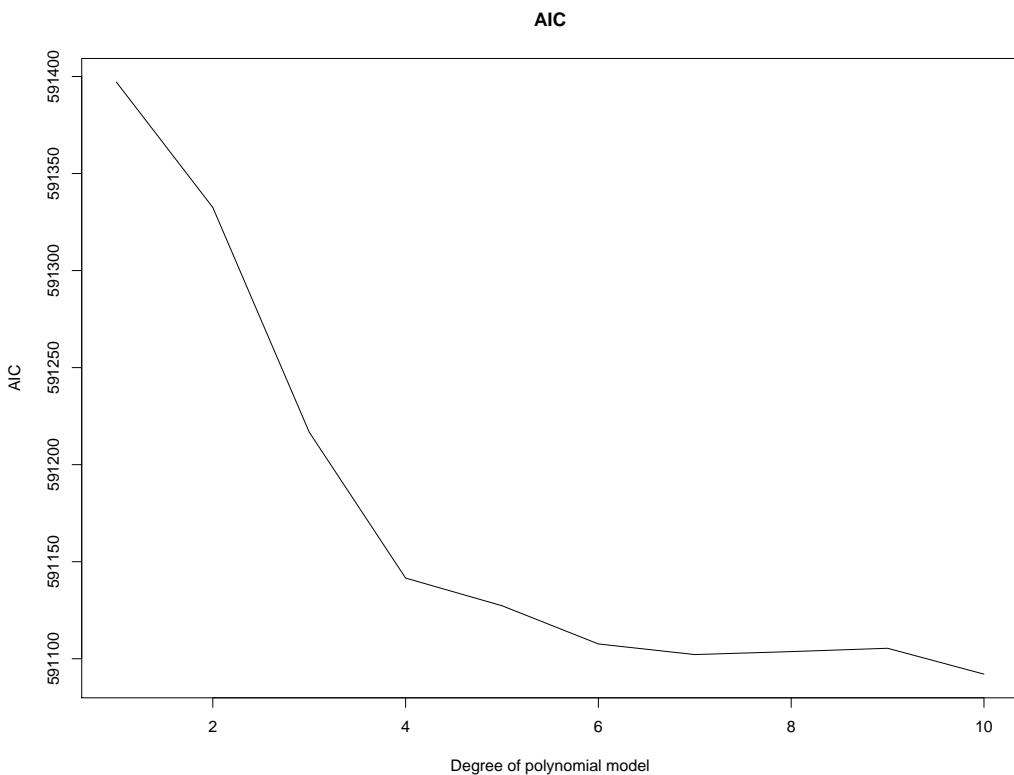
In order to determine the appropriate degree of the polynomial regression,we can fit the polynomial regression model with different degrees and then minimize the measure of optimism.We can obtain the measure of optimism using **Akaike information criterion(AIC)**,**Bayesian information criterion(BIC)** and **Cross-validation** method.

The value of **Cross validation error** of the different degrees of polynomial model is as follows,



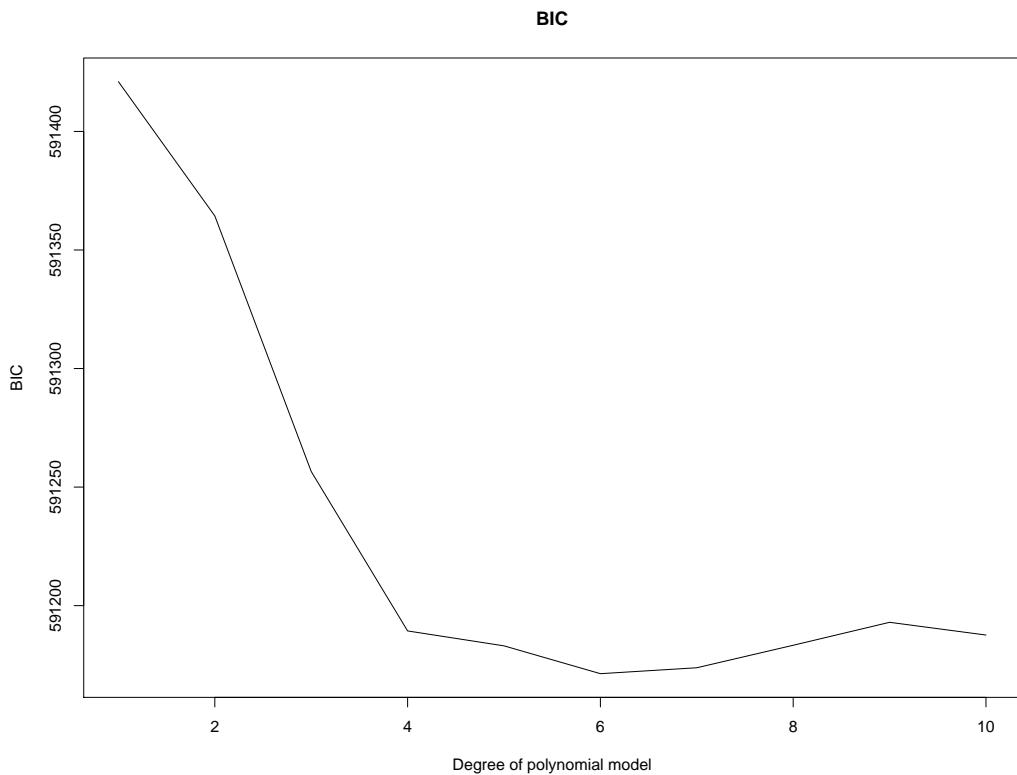
Cross Validation method suggest that we should fit a second degree **polynomial regression** of price on area of basement

The value of AIC of the different degrees of polynomial model is as follows,



AIC method suggest that we should fit a ten degree **polynomial regression** of price on area of basement.

The value of **BIC** of the different degrees of polynomial model is as follows,



BIC method suggest that we should fit a six degree **polynomial regression** of price on area of basement.

Cross Validation method is the most accurate method among the given methods. Thus, we should fit a six degree **polynomial regression** of price on area of basement.

Now, we should find the impact of the qualitative variables on the price of the house using **ANOVA** method.

We obtain the effect of number of bedrooms on the price of the house.

The **ANOVA** model is,

$$y_{ij} = \alpha + \mu_i + e_{ij}, i = 1, 2, \dots, n_i, j = 1, 2, \dots, 12$$

Here, y_{ij} = price of j th house with i number of bedrooms.

μ_i = additional effect of number of bedrooms on the price.

n_i = number of houses with i number of bedrooms

We set the null hypothesis,

$$H_0 : \mu_1 = \mu_2 = \mu_3 = \dots = \mu_{12} = 0 \text{ vs } H_1 : H_0 \text{ is not true}$$

The **ANOVA** table is given by,

```

summary(aov(price~as.factor(bedrooms)))

          Df   Sum Sq  Mean Sq F value Pr(>F)
as.factor(bedrooms)    11 1.932e+14 1.757e+13   252.8 <2e-16 ***
Residuals             21173 1.471e+15 6.948e+10
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

The p value is less than 0.05.

Thus,we can conclude that **number of bedrooms has significant effect on the price of the house.**

We obtain the effect of number of bathrooms on the price of the house.

The **ANOVA** table is given by,

```

summary(aov(price~as.factor(bathrooms)))

          Df   Sum Sq  Mean Sq F value Pr(>F)
as.factor(bathrooms)    24 5.050e+14 2.104e+13   384 <2e-16 ***
Residuals              21160 1.159e+15 5.479e+10
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

The p value is less than 0.05.

Thus,we can conclude that **number of bathrooms has significant effect on the price of the house.**

We obtain the effect of number of floors on the price of the house.

The **ANOVA** table is given by,

```

summary(aov(price~as.factor(floors)))

          Df   Sum Sq  Mean Sq F value Pr(>F)
as.factor(floors)      5 1.671e+14 3.342e+13   472.8 <2e-16 ***
Residuals              21179 1.497e+15 7.069e+10
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

The p value is less than 0.05.

Thus,we can conclude that **number of floors has significant effect on the price of the house.**

We obtain the effect of waterfront on the price of the house.

The **ANOVA** table is given by,

```

summary(aov(price~as.factor(waterfront)))

              Df      Sum Sq   Mean Sq F value Pr(>F)
as.factor(waterfront)    1 4.549e+13 4.549e+13   595.3 <2e-16 ***
Residuals                 21183 1.619e+15 7.642e+10
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

The p value is less than 0.05.

Thus,we can conclude that **waterfront has significant effect on the price of the house.**

We obtain the effect of view on the price of the house.

The **ANOVA** table is given by,

```

summary(aov(price~as.factor(view)))

              Df      Sum Sq   Mean Sq F value Pr(>F)
as.factor(view)      4 2.003e+14 5.009e+13   724.6 <2e-16 ***
Residuals                 21180 1.464e+15 6.912e+10
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

The p value is less than 0.05.

Thus,we can conclude that **view has significant effect on the price of the house.**

We obtain the effect of condition on the price of the house.

The **ANOVA** table is given by,

```

summary(aov(price~as.factor(condition)))

              Df      Sum Sq   Mean Sq F value Pr(>F)
as.factor(condition)    4 1.581e+13 3.953e+12   50.79 <2e-16 ***
Residuals                  21180 1.649e+15 7.783e+10
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

The p value is less than 0.05.

Thus,we can conclude that **condition has significant effect on the price of the house.**

We obtain the effect of grade on the price of the house.

The **ANOVA** table is given by,

```
summary(aov(price~as.factor(grade)))  
  
          Df   Sum Sq   Mean Sq F value Pr(>F)  
as.factor(grade)    10 8.507e+14 8.507e+13    2214 <2e-16 ***  
Residuals         21174 8.137e+14 3.843e+10  
---  
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The p value is less than 0.05.

Thus, we can conclude that **grade has significant effect on the price of the house.**

We obtain the effect of renovation indicator on the price of the house.

The **ANOVA** table is given by,

```
renovation_indicator=as.integer(yr_renovated>0)  
summary(aov(price~as.factor(renovation_indicator)))  
  
          Df   Sum Sq   Mean Sq F value Pr(>F)  
as.factor(renovation_indicator)    1 1.554e+13 1.554e+13    199.7 <2e-16 ***  
Residuals                     21183 1.649e+15 7.783e+10  
  
as.factor(renovation_indicator) ***  
Residuals  
---  
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The p value is less than 0.05.

Thus, we can conclude that **renovation indicator has significant effect on the price of the house.**

We obtain the effect of number of bathrooms on the price of the house.

The **ANOVA** table is given by,

```
summary(aov(price~as.factor(bedrooms)))  
  
          Df   Sum Sq   Mean Sq F value Pr(>F)  
as.factor(bedrooms)    11 1.932e+14 1.757e+13    252.8 <2e-16 ***  
Residuals           21173 1.471e+15 6.948e+10  
---  
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The p value is less than 0.05.

Thus, we can conclude that **number of bathrooms has significant effect on the price of the house.**

Multivariate Models and Selection of Optimum Models

Before fitting any model, we check whether some of the covariates are highly correlated or not i.e. if there is any multicollinearity present.

```
model <- lm(price~., data=Data_House_1)
summary(model)

Call:
lm(formula = price ~ ., data = Data_House_1)

Residuals:
    Min      1Q  Median      3Q     Max 
-597678 -99996  -9065   82911 1037952 

Coefficients: (1 not defined because of singularities)
              Estimate Std. Error t value Pr(>|t|)    
(Intercept) 5.480e+06 1.059e+05 51.756 < 2e-16 ***
bedrooms   -2.638e+04 1.627e+03 -16.213 < 2e-16 ***
bathrooms   3.512e+04 2.711e+03 12.956 < 2e-16 ***
sqft_living 1.339e+02 3.650e+00 36.689 < 2e-16 ***
sqft_lot    -1.651e-01 3.429e-02 -4.816 1.48e-06 ***
floors      4.027e+04 2.861e+03 14.074 < 2e-16 ***
waterfront  6.016e+05 2.708e+04 22.219 < 2e-16 ***
view        3.972e+04 1.775e+03 22.380 < 2e-16 ***
condition   1.905e+04 1.898e+03 10.037 < 2e-16 ***
grade       1.203e+05 1.669e+03 72.100 < 2e-16 ***
sqft_above   -1.310e+01 3.468e+00 -3.779 0.000158 *** 
sqft_basement NA         NA         NA        
yr_builtin  -3.171e+03 5.431e+01 -58.381 < 2e-16 ***
yr_renovated 3.446e+00 3.078e+00  1.119 0.263010  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 162300 on 21172 degrees of freedom
Multiple R-squared:  0.6649, Adjusted R-squared:  0.6647 
F-statistic: 3500 on 12 and 21172 DF,  p-value: < 2.2e-16
```

Then we use vif() function to calculate the vif values for different predictors

```

library("car")

Warning: package 'car' was built under R version 3.6.3
Loading required package: carData
Warning: package 'carData' was built under R version 3.6.3

Attaching package: 'car'
The following object is masked from 'package:dplyr':
  recode
The following object is masked from 'package:purrr':
  some
The following object is masked from 'package:boot':
  logit

vif(model)

Error in vif.default(model): there are aliased coefficients in the model

```

the output means that there are some variables present which have perfect linear relationship i.e. perfect multicollinearity. To detect this we use alias() function.

```

alias(model)

Model :
price ~ bedrooms + bathrooms + sqft_living + sqft_lot + floors +
  waterfront + view + condition + grade + sqft_above + sqft_basement +
  yr_built + yr_renovated

Complete :
            (Intercept) bedrooms bathrooms sqft_living sqft_lot floors
sqft_basement  0          0          0          1          0          0
                  waterfront view condition grade sqft_above yr_built
sqft_basement  0          0          0          0         -1          0
                  yr_renovated
sqft_basement  0

```

since the values are 1,-1 for the predictors sqft_living and sqft_above hence, we examine them carefully and notice that sqft_living is actually equal to sqft_above+sqft_base. So we remove the column sqft_living as it is a redundant information. Then we fit the model again.

```

Data_House_2 = Data_House_1[,-4]
model1 <- lm(price~., data=Data_House_2)
summary(model1)

```

Call:

```

lm(formula = price ~ ., data = Data_House_2)

Residuals:
    Min      1Q  Median      3Q     Max 
-597678 -99996 -9065   82911 1037952 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept) 5.480e+06 1.059e+05 51.756 < 2e-16 ***
bedrooms    -2.638e+04 1.627e+03 -16.213 < 2e-16 ***
bathrooms   3.512e+04 2.711e+03 12.956 < 2e-16 ***
sqft_lot    -1.651e-01 3.429e-02 -4.816 1.48e-06 ***
floors       4.027e+04 2.861e+03 14.074 < 2e-16 ***
waterfront  6.016e+05 2.708e+04 22.219 < 2e-16 ***
view         3.972e+04 1.775e+03 22.380 < 2e-16 ***
condition   1.905e+04 1.898e+03 10.037 < 2e-16 ***
grade        1.203e+05 1.669e+03 72.100 < 2e-16 ***
sqft_above   1.208e+02 2.791e+00 43.287 < 2e-16 ***
sqft_basement 1.339e+02 3.650e+00 36.689 < 2e-16 ***
yr_builtin  -3.171e+03 5.431e+01 -58.381 < 2e-16 ***
yr_renovated 3.446e+00 3.078e+00  1.119   0.263  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 162300 on 21172 degrees of freedom
Multiple R-squared:  0.6649, Adjusted R-squared:  0.6647 
F-statistic:  3500 on 12 and 21172 DF,  p-value: < 2.2e-16

```

This gives the same value of R^2 .

Forward Subset Selection

Now, we use forward selection method to select optimum models with different number of covariates as follows :-

```

regfit.fwd = regsubsets(price~., data = Data_House_2, method = "forward", nvmax = 12)
reg_summary_fwd = summary(regfit.fwd)
reg_summary_fwd$outmat

      bedrooms bathrooms sqft_lot floors waterfront view condition
1 ( 1 )    " "      " "      " "      " "      " "      " "
2 ( 1 )    " "      " "      " "      " "      " "      " "
3 ( 1 )    " "      "*"      " "      " "      " "      " "
4 ( 1 )    " "      "*"      " "      " "      " "      "*" 
5 ( 1 )    " "      "*"      " "      " "      " "      "*" 
6 ( 1 )    " "      "*"      " "      " "      " "      "*" 
7 ( 1 )    " "      "*"      " "      " "      "*"      "*" 

```

```

8  ( 1 )   "*"      "*"      " "      " "      "*"      "*"      " "
9  ( 1 )   "*"      "*"      " "      " "      "*"      "*"      "*"      " "
10 ( 1 )  "*"      "*"      " "      " "      "*"      "*"      "*"      "*"
11 ( 1 )  "*"      "*"      "*"      "*"      "*"      "*"      "*"      "*"
12 ( 1 )  "*"      "*"      "*"      "*"      "*"      "*"      "*"      "*"

            grade sqft_above sqft_basement yr_built yr_renovated
1  ( 1 )   "*"      " "      " "      " "      " "
2  ( 1 )   "*"      " "      " "      "*"      " "      " "
3  ( 1 )   "*"      " "      " "      "*"      " "      " "
4  ( 1 )   "*"      " "      " "      "*"      " "      " "
5  ( 1 )   "*"      "*"      " "      "*"      " "      " "
6  ( 1 )   "*"      "*"      "*"      "*"      " "      " "
7  ( 1 )   "*"      "*"      "*"      "*"      " "      " "
8  ( 1 )   "*"      "*"      "*"      "*"      " "      " "
9  ( 1 )   "*"      "*"      "*"      "*"      " "      " "
10 ( 1 )  "*"      "*"      "*"      "*"      " "      " "
11 ( 1 )  "*"      "*"      "*"      "*"      " "      " "
12 ( 1 )  "*"      "*"      "*"      "*"      "*"      "*"

ind_num = list()

```

Now, to select the “best” model among all the optimum models with different number of covariates, we use different criterions like R^2 , RSS , R_{adj}^2 , Mallow's– C_p , BIC and $CV(10-fold)$.

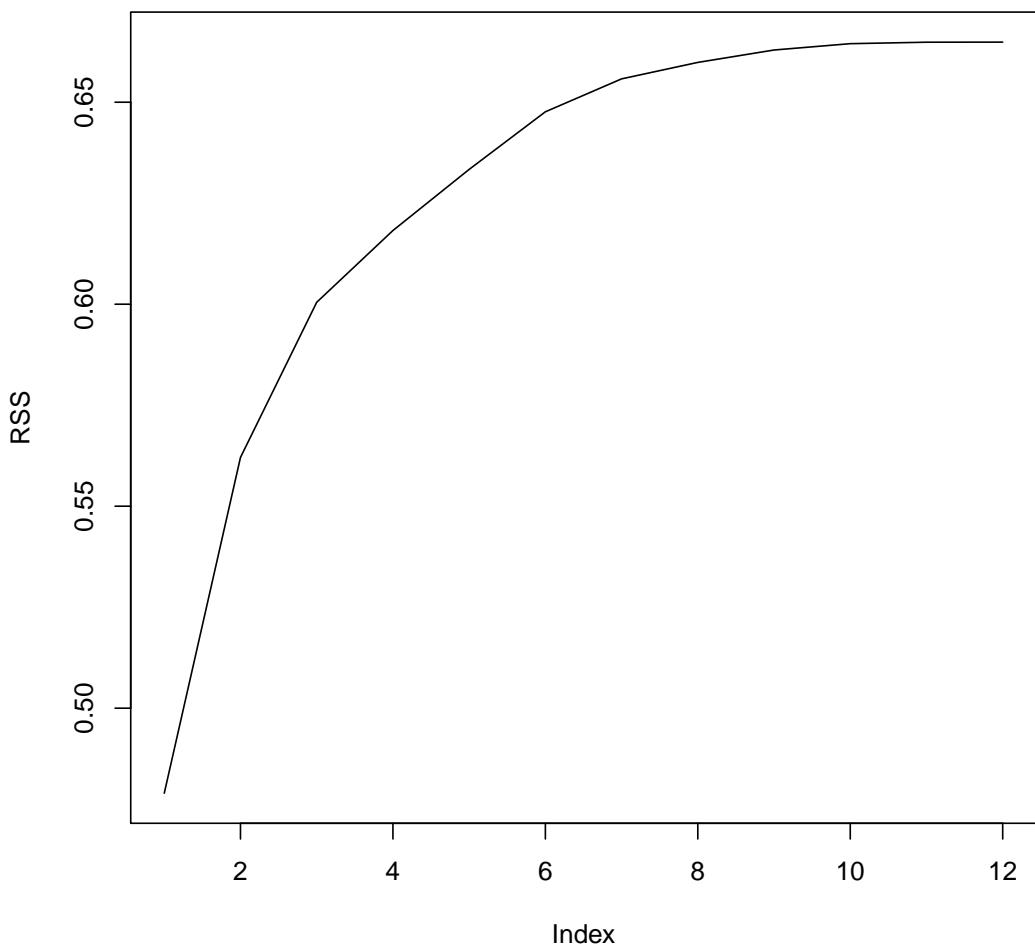
```

ind_num = list()
cv.error.12 = rep(0,12)
for(i in 1:12)
{
  ind = as.integer(reg_summary_fwd$outmat[i,] == "*")
  ind_num[[i]] = which(ind == 1) + 1
  glm.fit = glm(price~. ,data=Data_House_2[,c(1,ind_num[[i]])])
  cv.error.12[i] = cv.glm(Data_House_1 ,glm.fit,K = 10)$delta[1]
}

```

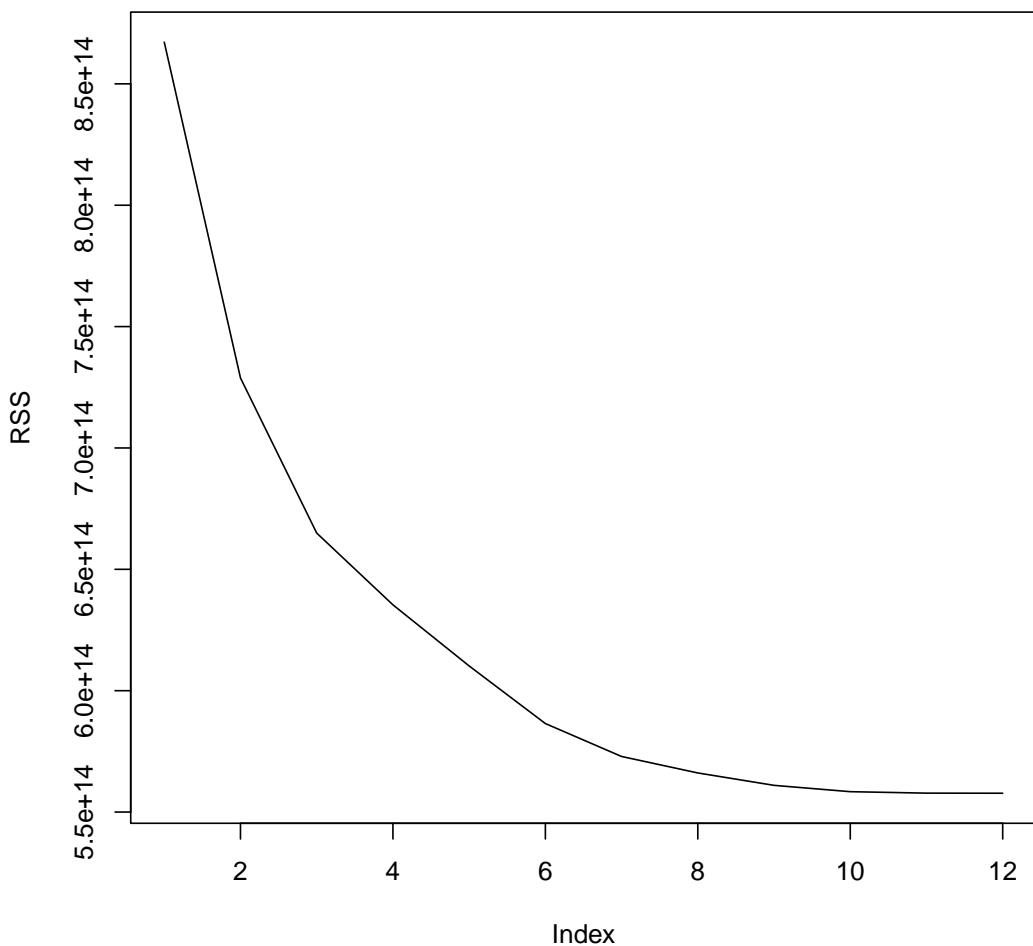
1. Values of R^2 for different models :-

```
plot(reg_summary_fwd$rsq,type = "l",ylab = expression(RSS))
```



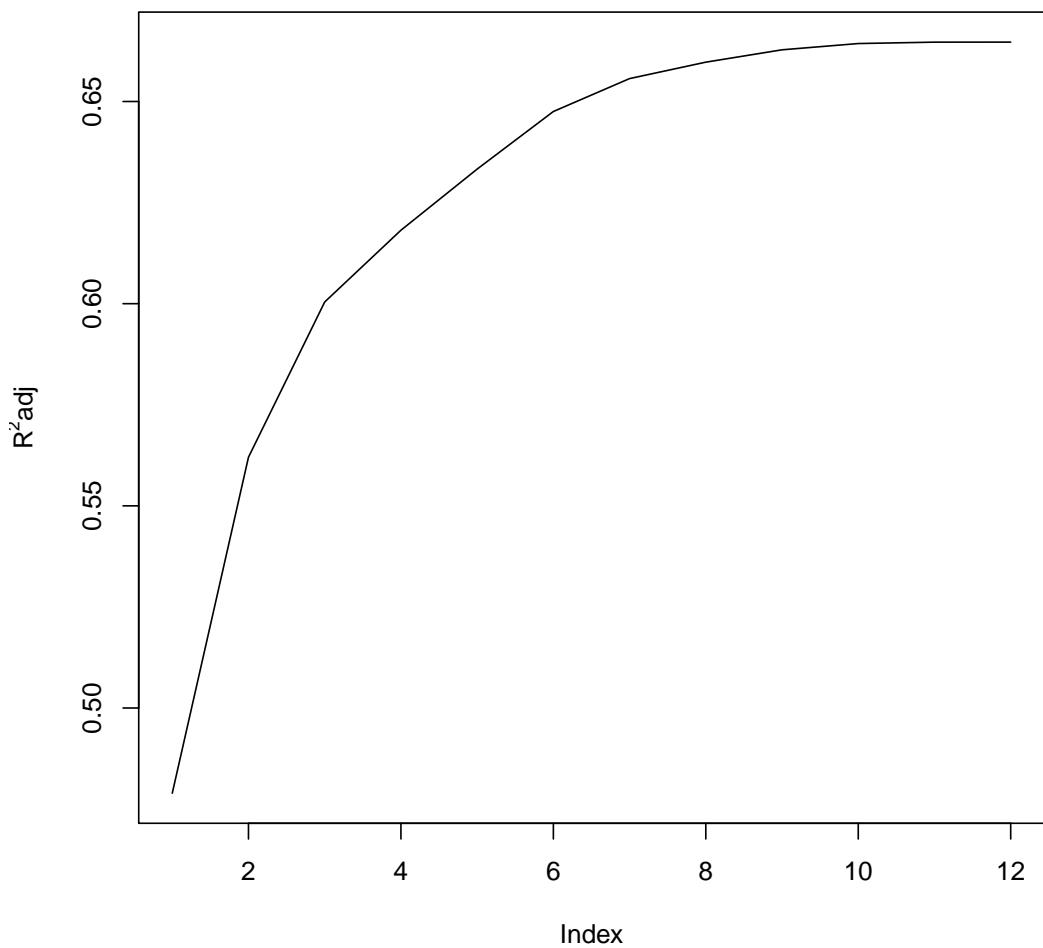
2. Value of Residual Sum of squares(RSS) :-

```
plot(reg_summary_fwd$rss,type = "l",ylab = expression(RSS))
```



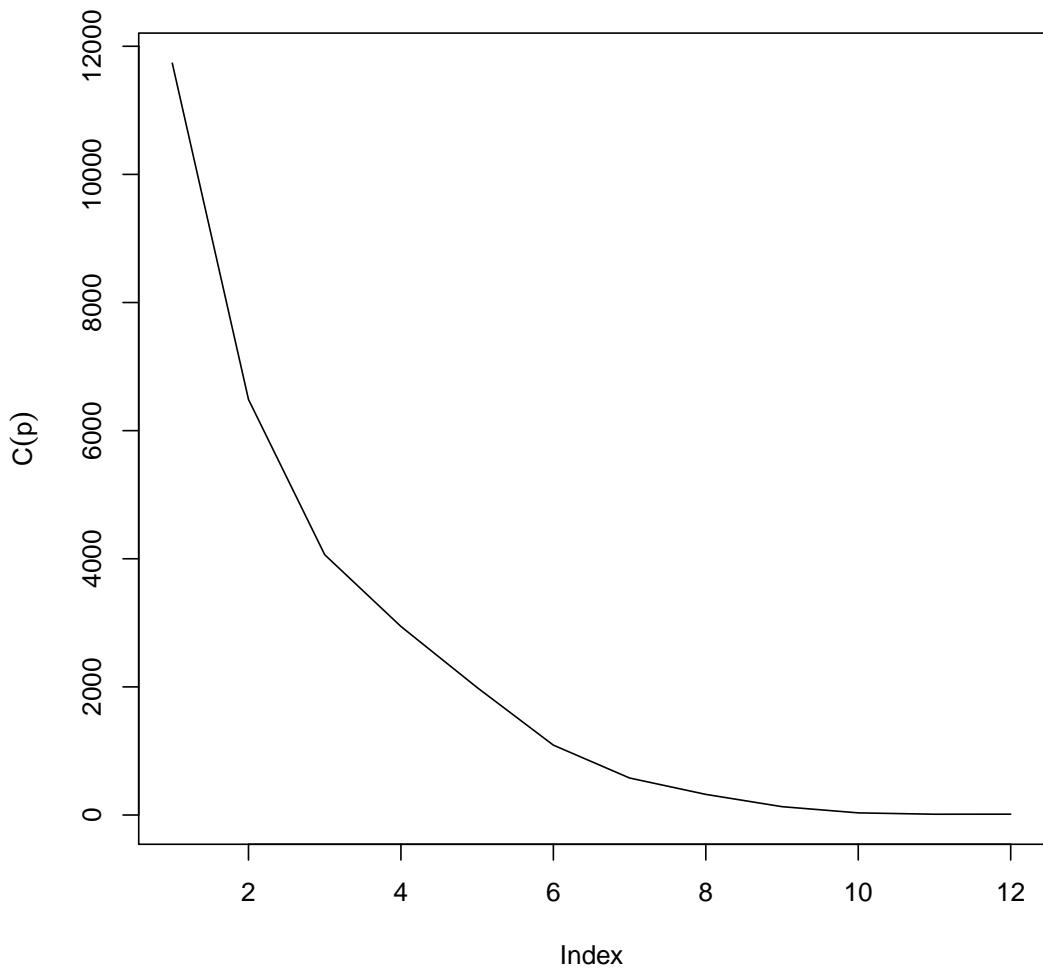
3. That of the adjusted- R^2 :-

```
plot(reg_summary_fwd$adjr2, type = "l", ylab = expression(paste(R^2, "adj")))
```



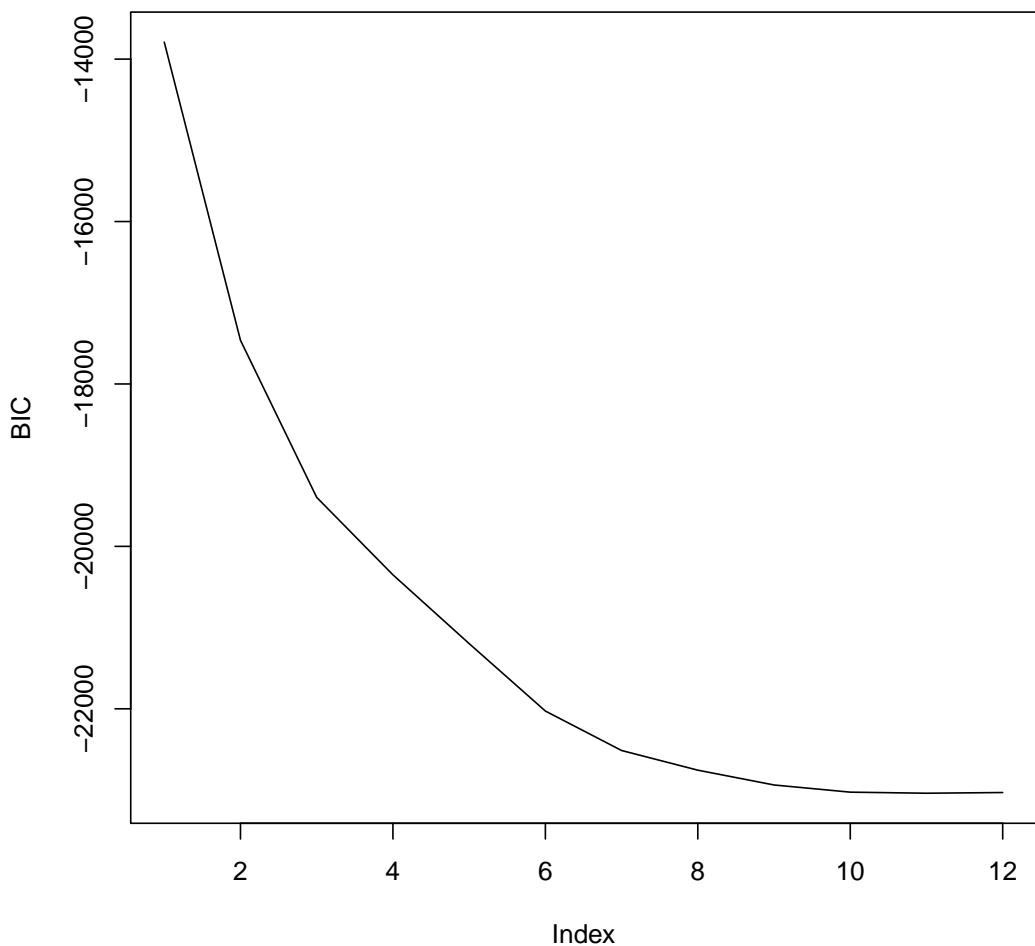
4. Mallow's $- C_p$:-

```
plot(reg_summary_fwd$cp,type = "l",ylab = expression(C(p)))
```



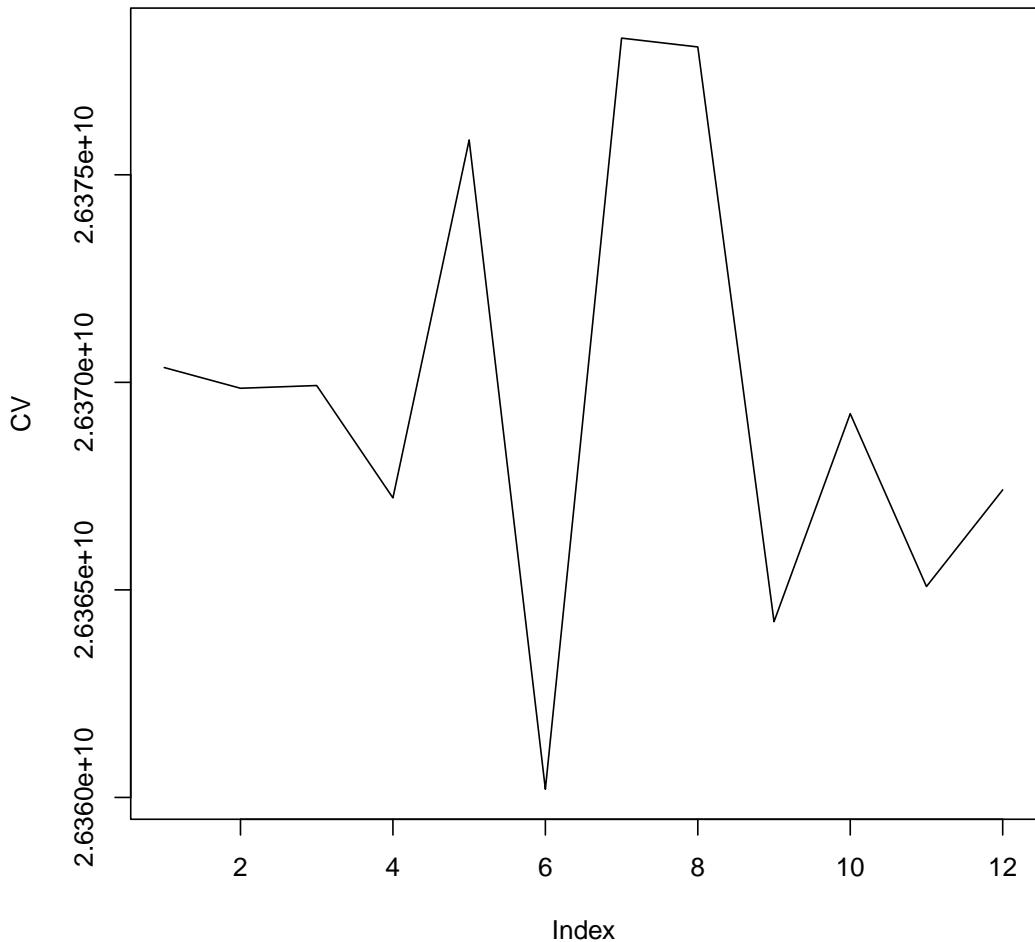
5. *BIC* values :-

```
plot(reg_summary_fwd$bic,type = "l",ylab = "BIC")
```



6. 10 – fold CV values :-

```
plot(cv.error.12,type = "l",ylab = "CV")
```



Hence, the forward subset selection selects the model :-

```

opt_ind = which.min(cv.error.12)
opt_ind

[1] 6

glm.fit.opt = glm(price~. ,data=Data_House_2[,c(1,ind_num[[opt_ind]])])
glm.fit.opt

Call: glm(formula = price ~ ., data = Data_House_2[, c(1, ind_num[[opt_ind]])])

Coefficients:
(Intercept)      bathrooms        view       grade    sqft_above
      5495527.69       40527.11      50938.55     127912.04      103.19
sqft_basement    yr_builtin
          97.23      -3175.92

```

```
Degrees of Freedom: 21184 Total (i.e. Null); 21178 Residual
Null Deviance: 1.664e+15
Residual Deviance: 5.865e+14 AIC: 569500
```

Backward Subset Selection

Now, we use backward selection method to select optimum models with different number of covariates as follows :-

```
regfit.bwd = regsubsets(price~., data = Data_House_2, method = "backward", nvmax = 12)
reg_summary_bwd = summary(regfit.bwd)
reg_summary_bwd$outmat

      bedrooms bathrooms sqft_lot floors waterfront view condition
1 ( 1 )   " "     " "     " "     " "     " "     " "     " "
2 ( 1 )   " "     " "     " "     " "     " "     " "     " "
3 ( 1 )   " "     " "     " "     " "     " "     " "     " "
4 ( 1 )   " "     " "     " "     " "     " "     " "     " "
5 ( 1 )   " "     " "     " "     " "     " "     "*"     " "
6 ( 1 )   " "     " "     " "     " "     "*"     "*"     " "
7 ( 1 )   " "     " "     " "     "*"     "*"     "*"     " "
8 ( 1 )   "*"     " "     " "     "*"     "*"     "*"     " "
9 ( 1 )   "*"     "*"     " "     "*"     "*"     "*"     " "
10 ( 1 )  "*"     "*"     " "     "*"     "*"     "*"     "*"
11 ( 1 )  "*"     "*"     "*"     "*"     "*"     "*"     "*"
12 ( 1 )  "*"     "*"     "*"     "*"     "*"     "*"     "*"

      grade sqft_above sqft_basement yr_builtin yr_renovated
1 ( 1 )  "*"     " "     " "     " "     " "
2 ( 1 )  "*"     " "     " "     "*"     " "
3 ( 1 )  "*"     "*"     " "     "*"     " "
4 ( 1 )  "*"     "*"     "*"     "*"     " "
5 ( 1 )  "*"     "*"     "*"     "*"     " "
6 ( 1 )  "*"     "*"     "*"     "*"     " "
7 ( 1 )  "*"     "*"     "*"     "*"     " "
8 ( 1 )  "*"     "*"     "*"     "*"     " "
9 ( 1 )  "*"     "*"     "*"     "*"     " "
10 ( 1 )  "*"     "*"     "*"     "*"     " "
11 ( 1 )  "*"     "*"     "*"     "*"     " "
12 ( 1 )  "*"     "*"     "*"     "*"     "*"

ind_num = list()
```

Now, to select the “best” model among all the optimum models with different number of covariates, we use different criterions like R^2 , RSS , R_{adj}^2 , Mallow's- C_p , BIC and $CV(10-fold)$.

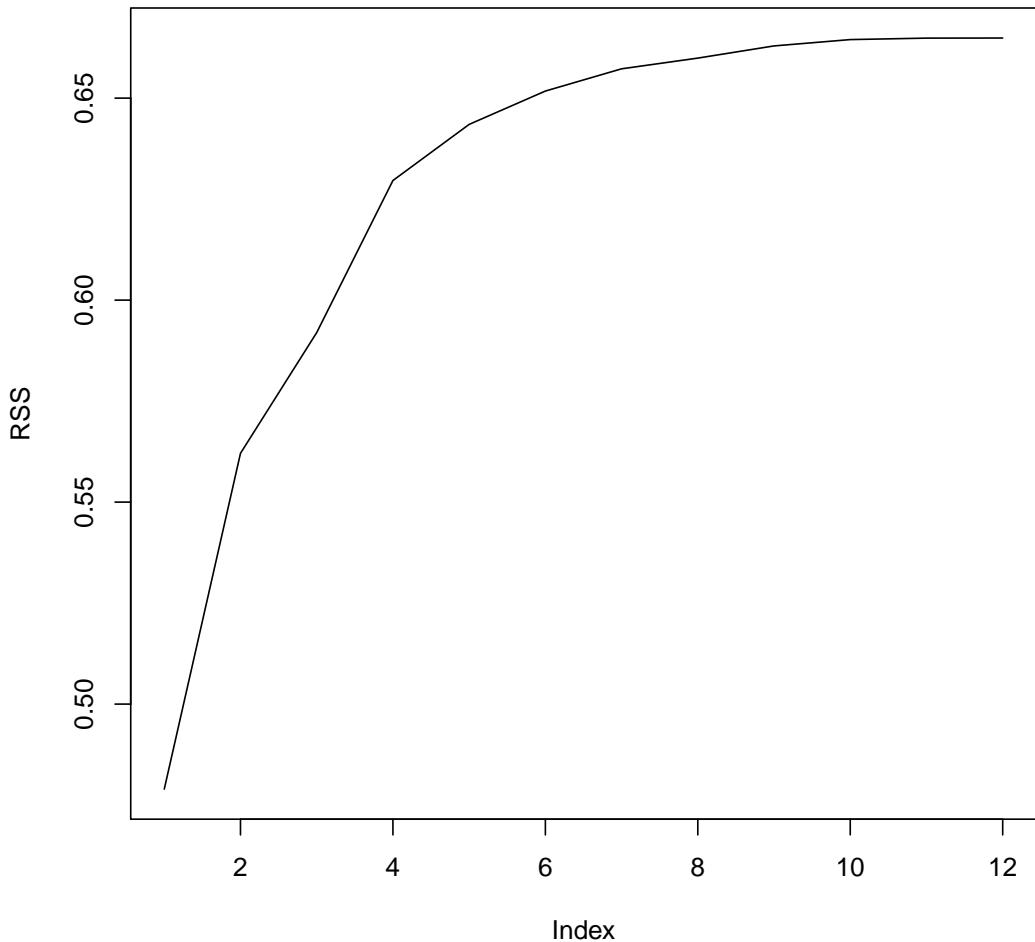
```

ind_num = list()
cv.error.12 = rep(0,12)
for(i in 1:12)
{
  ind = as.integer(reg_summary_bwd$outmat[i,] == "*")
  ind_num[[i]] = which(ind == 1) + 1
  glm.fit = glm(price~. ,data=Data_House_2[,c(1,ind_num[[i]])])
  cv.error.12[i] = cv.glm(Data_House_1 ,glm.fit,K = 10)$delta[1]
}

```

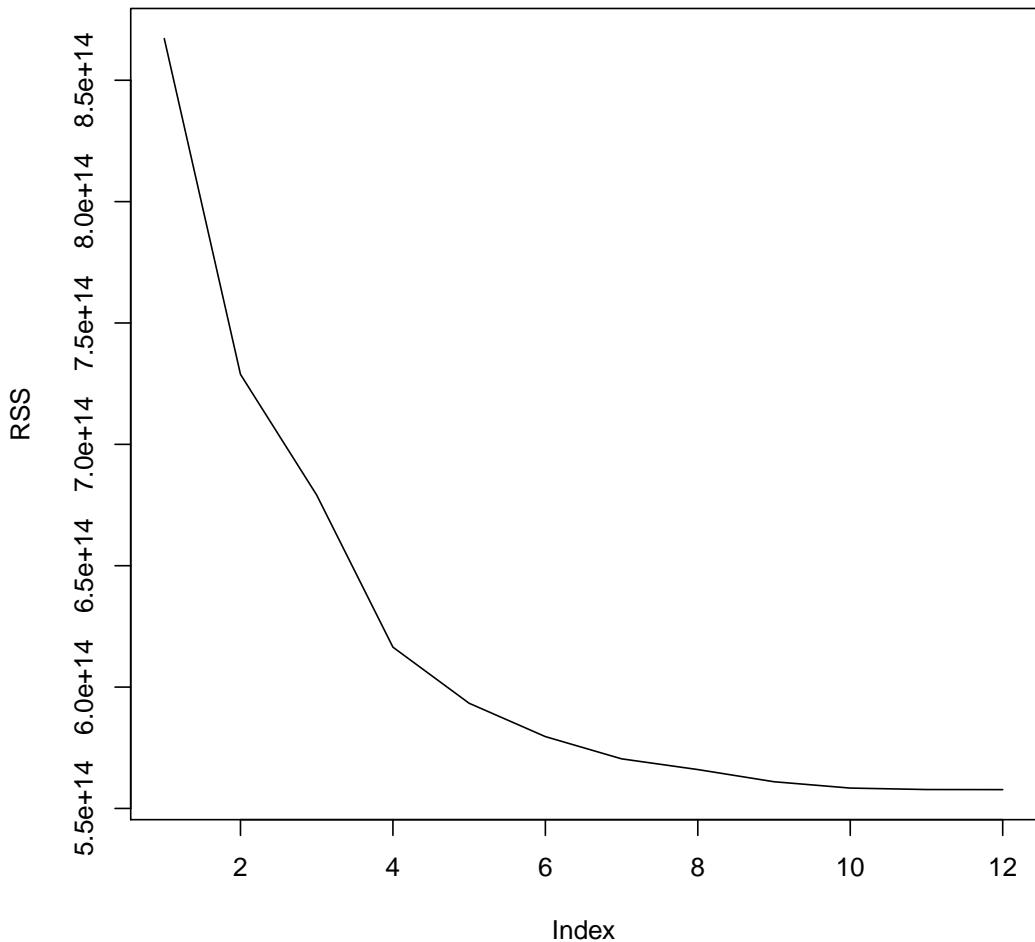
1. Values of R^2 for different models :-

```
plot(reg_summary_bwd$rsq,type = "l",ylab = expression(RSS))
```



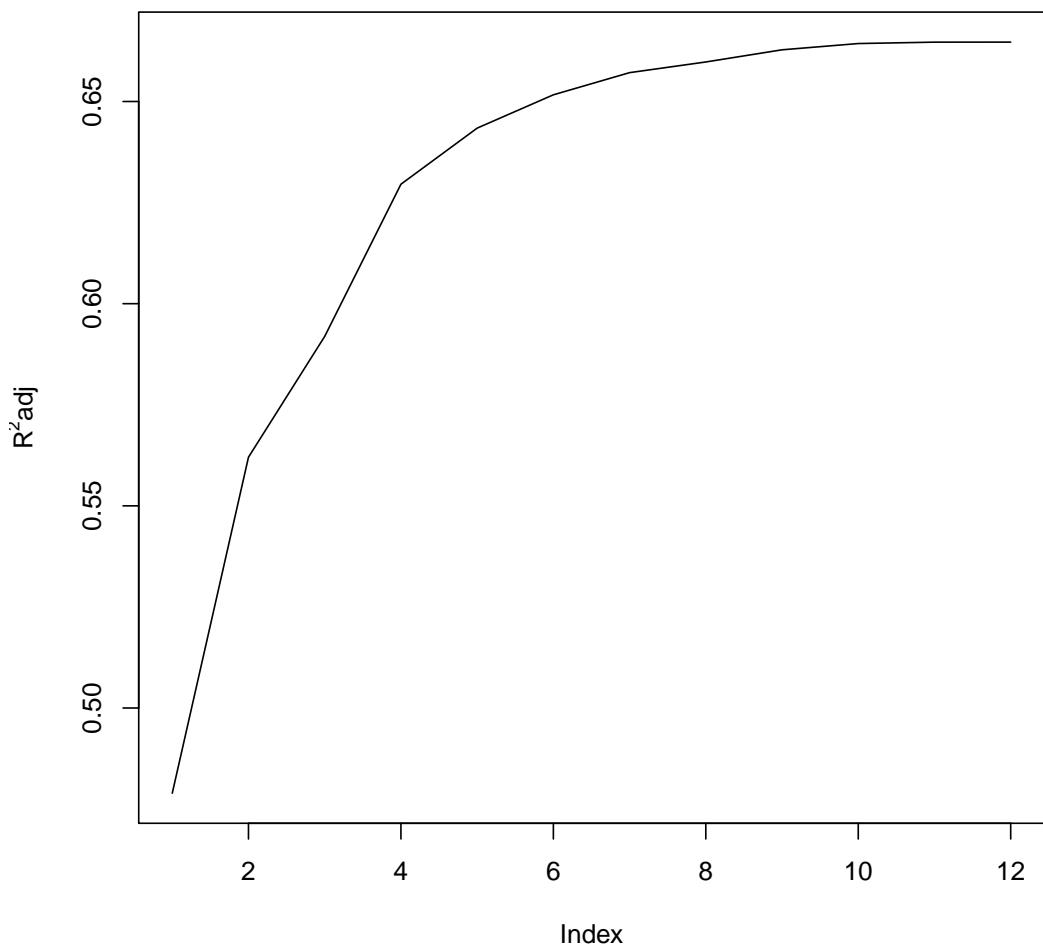
2. Value of Residual Sum of squares(RSS) :-

```
plot(reg_summary_bwd$rss,type = "l",ylab = expression(RSS))
```



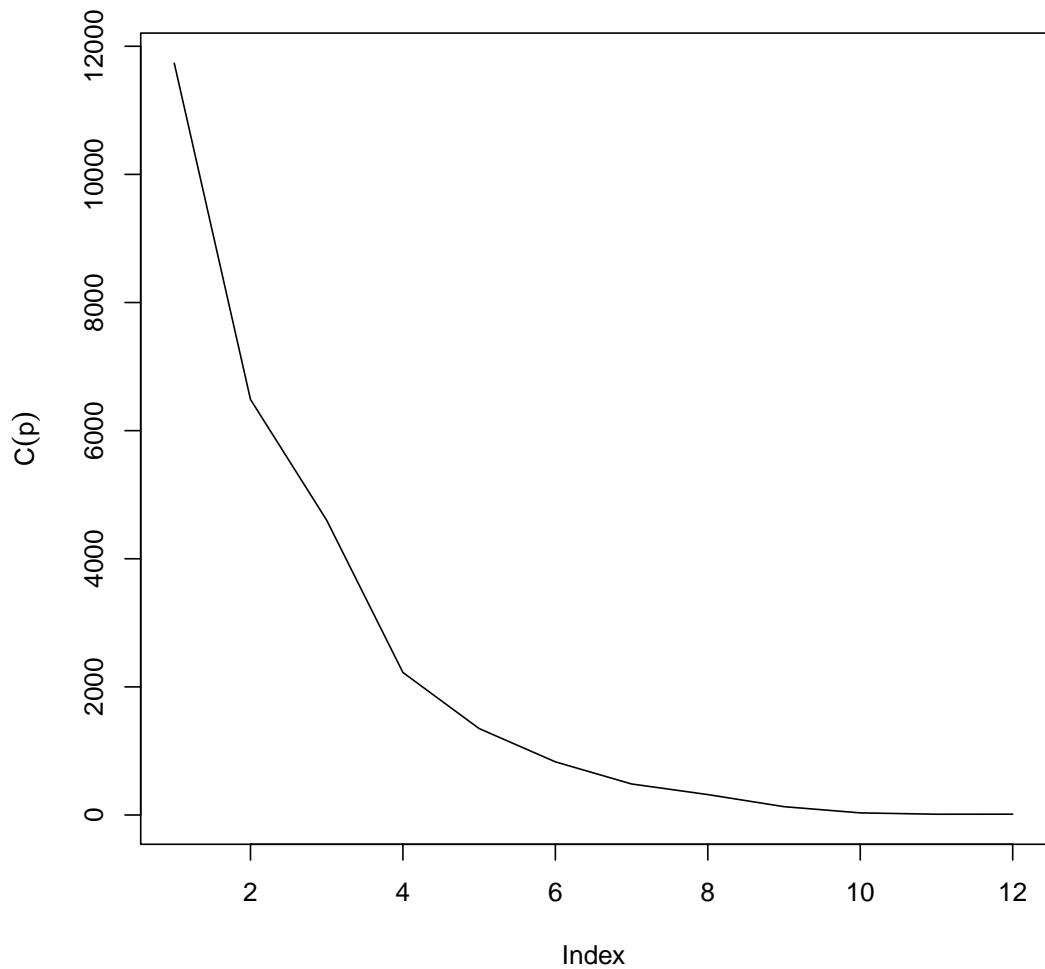
3. That of the adjusted- R^2 :-

```
plot(reg_summary_bwd$adjr2,type = "l",ylab = expression(paste(R^2,"adj")))
```



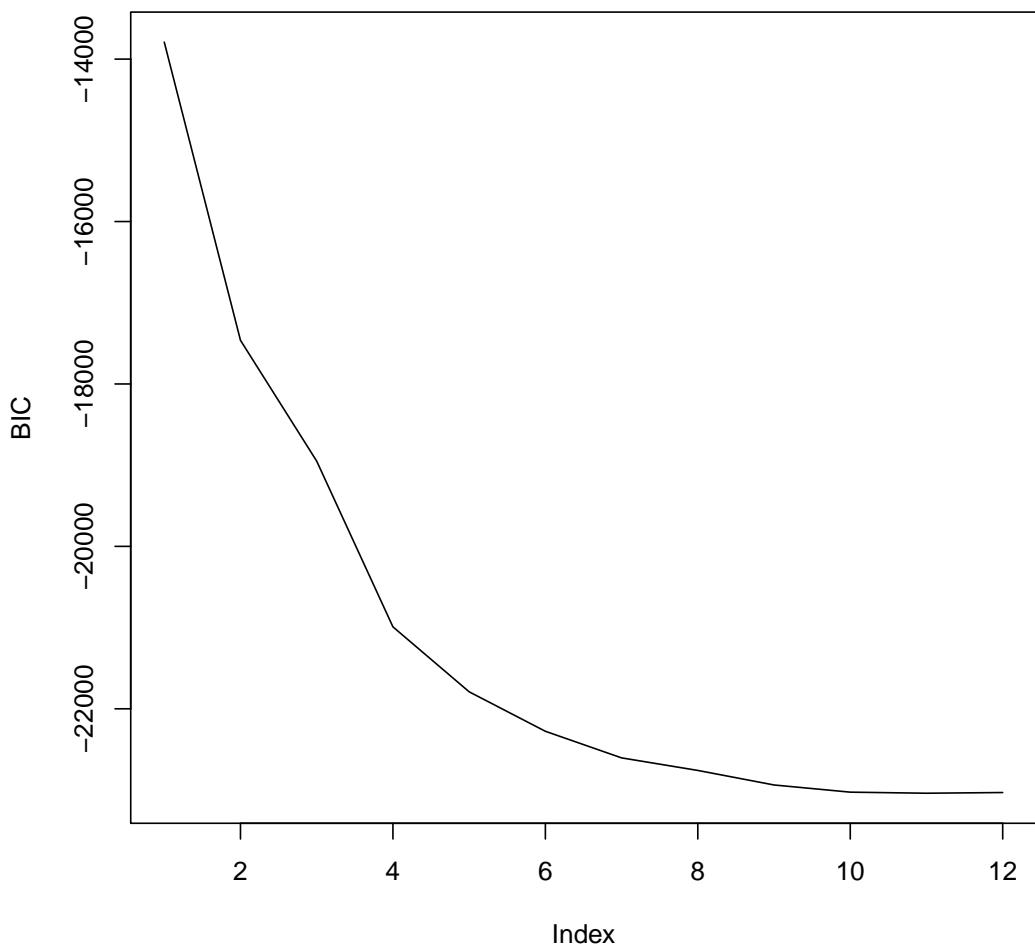
4. Mallow's $- C_p$:-

```
plot(reg_summary_bwd$cp,type = "l",ylab = expression(C(p)))
```



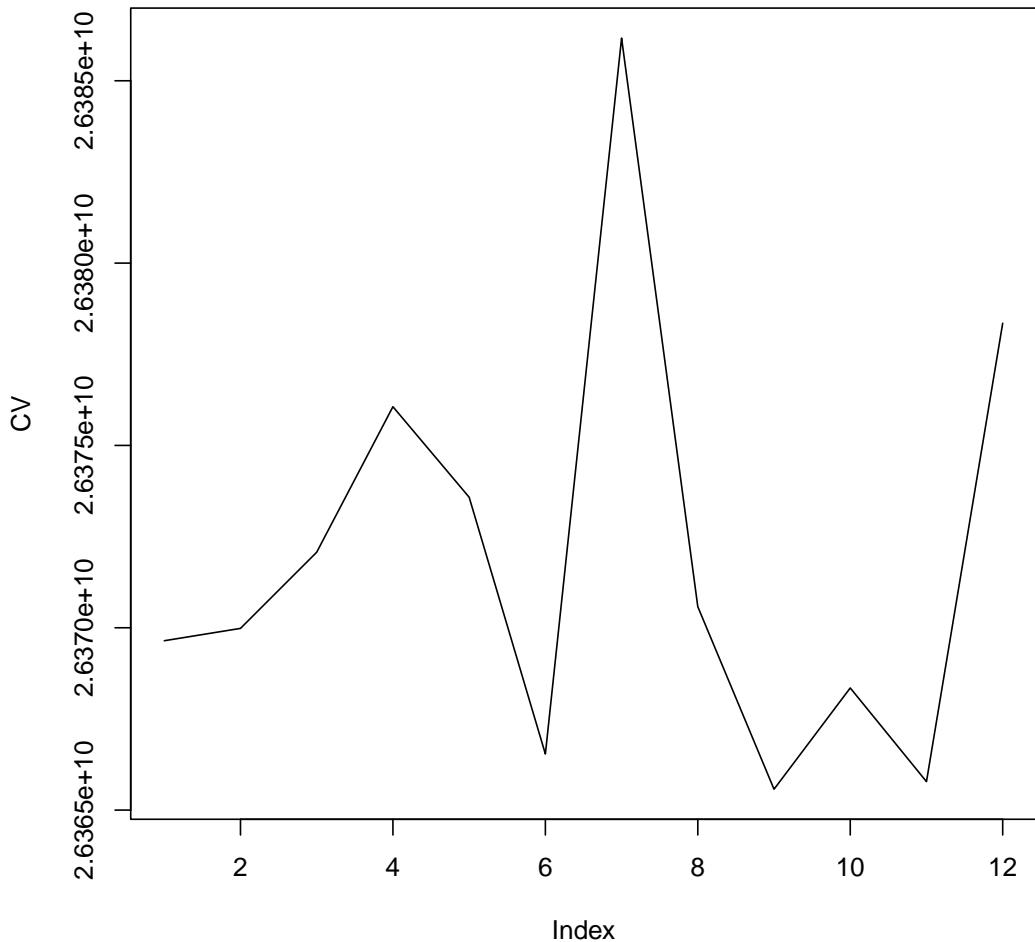
5. *BIC* values :-

```
plot(reg_summary_bwd$bic,type = "l",ylab = "BIC")
```



6. 10 – fold CV values :-

```
plot(cv.error.12,type = "l",ylab = "CV")
```



Hence, the backward subset selection selects the model :-

```

opt_ind = which.min(cv.error.12)
opt_ind

[1] 9

glm.fit.opt = glm(price~. ,data=Data_House_2[,c(1,ind_num[[opt_ind]])])
glm.fit.opt

Call: glm(formula = price ~ ., data = Data_House_2[, c(1, ind_num[[opt_ind]])])

Coefficients:
(Intercept)      bedrooms     bathrooms      floors    waterfront
 5860822.6       -25050.9        37056.3       39317.2       603765.5
      view          grade      sqft_above   sqft_basement    yr_built
  39433.8       120647.5        117.6           135.0        -3334.1

```

```
Degrees of Freedom: 21184 Total (i.e. Null); 21175 Residual
Null Deviance: 1.664e+15
Residual Deviance: 5.61e+14 AIC: 568600
```

Best Subset Selection

Finally, we use best selection method to select optimum models with different number of covariates as follows :-

```
regfit = regsubsets(price~., data = Data_House_2, nvmax = 12)
reg_summary= summary(regfit.bwd)
reg_summary$outmat

      bedrooms bathrooms sqft_lot floors waterfront view condition
1 ( 1 ) " " " " " " " " " "
2 ( 1 ) " " " " " " " " " "
3 ( 1 ) " " " " " " " " " "
4 ( 1 ) " " " " " " " " " "
5 ( 1 ) " " " " " " " " " "
6 ( 1 ) " " " " " " " *" " *"
7 ( 1 ) " " " " " *" " *" " *"
8 ( 1 ) "*" " " " " " *" " *" " *"
9 ( 1 ) "*" " *" " " " *" " *" " *"
10 ( 1 ) "*" " *" " " " *" " *" " *"
11 ( 1 ) "*" " *" " *" " *" " *" " *"
12 ( 1 ) "*" " *" " *" " *" " *" " *"

      grade sqft_above sqft_basement yr_builtin yr_renovated
1 ( 1 ) "*" " " " " " "
2 ( 1 ) "*" " " " " *" " "
3 ( 1 ) "*" " *" " " " *" " "
4 ( 1 ) "*" " *" " *" " *" " "
5 ( 1 ) "*" " *" " *" " *" " "
6 ( 1 ) "*" " *" " *" " *" " "
7 ( 1 ) "*" " *" " *" " *" " "
8 ( 1 ) "*" " *" " *" " *" " "
9 ( 1 ) "*" " *" " *" " *" " "
10 ( 1 ) "*" " *" " *" " *" " "
11 ( 1 ) "*" " *" " *" " *" " "
12 ( 1 ) "*" " *" " *" " *" " *"

ind_num = list()
```

Now, to select the “best” model among all the optimum models with different number of covariates, we use different criterions like R^2 , RSS , R_{adj}^2 , Mallow's- C_p , BIC and $CV(5-fold)$.

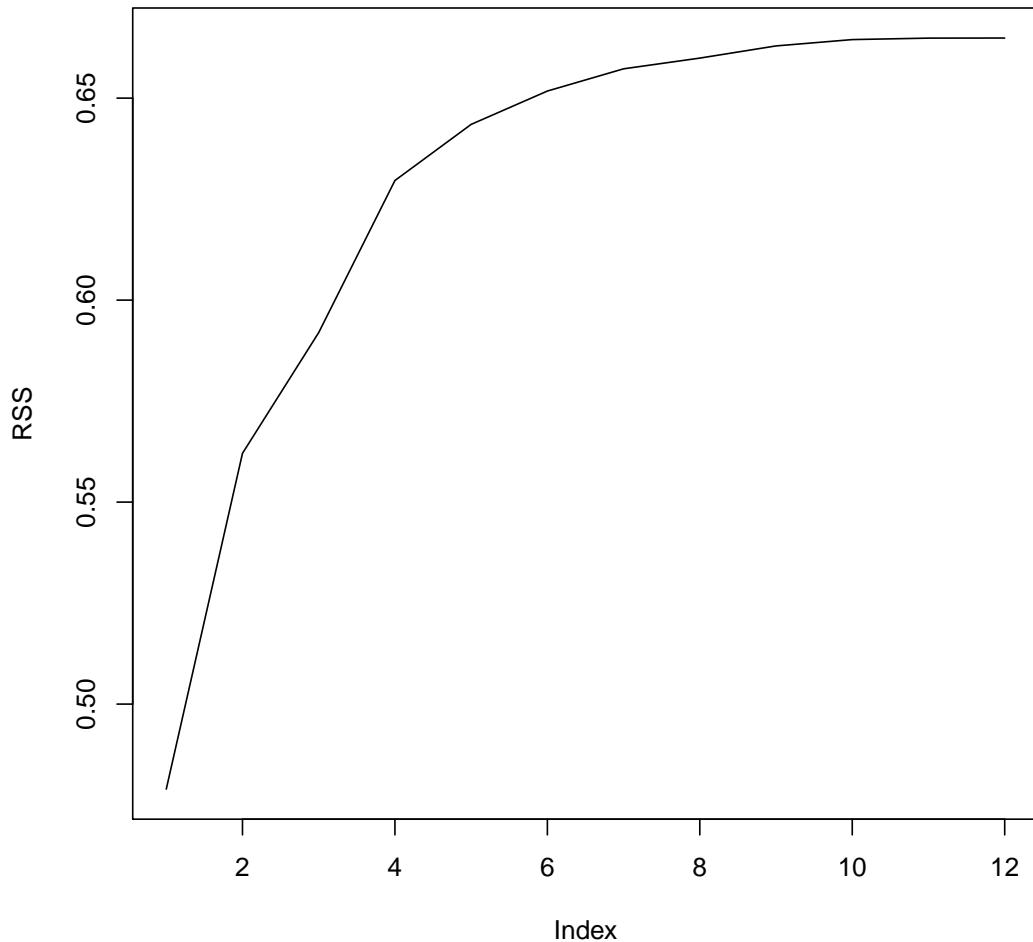
```

ind_num = list()
cv.error.12 = rep(0,12)
for(i in 1:12)
{
  ind = as.integer(reg_summary$outmat[i,] == "*")
  ind_num[[i]] = which(ind == 1) + 1
  glm.fit = glm(price~. ,data=Data_House_2[,c(1,ind_num[[i]])])
  cv.error.12[i] = cv.glm(Data_House_1 ,glm.fit,K = 5)$delta[1]
}

```

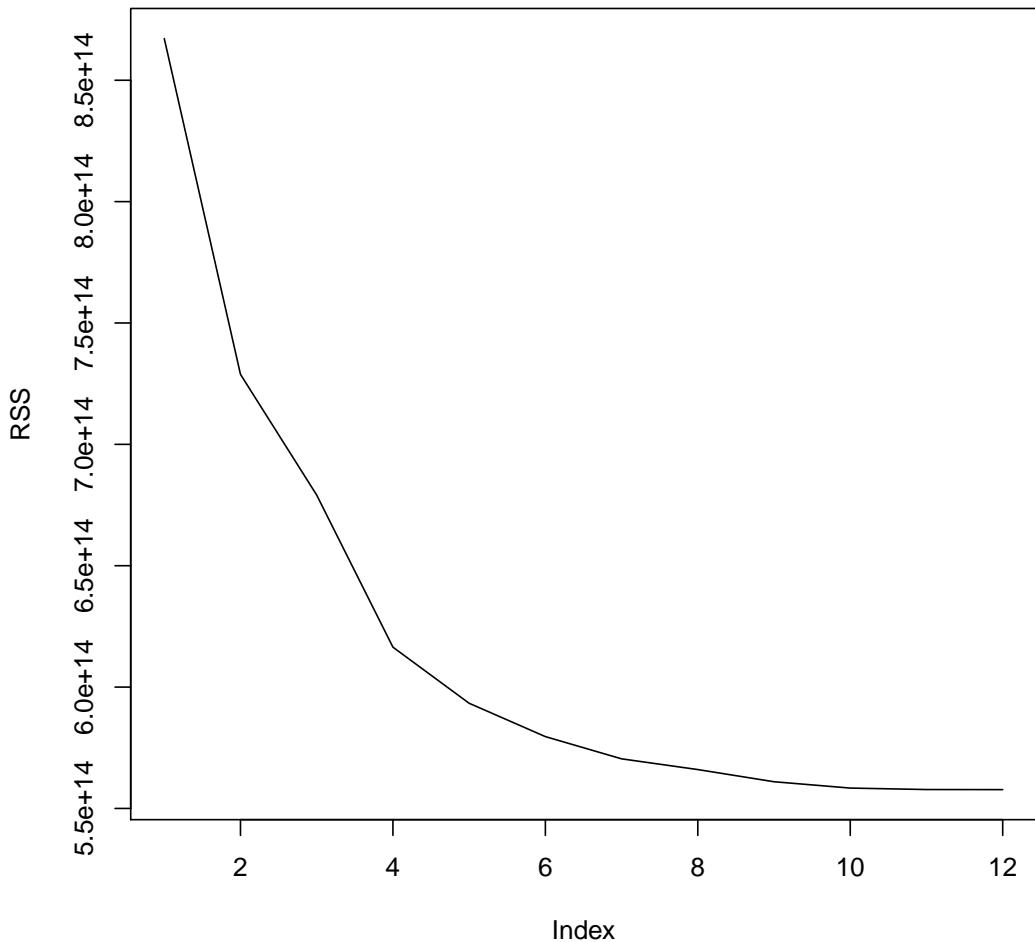
1. Values of R^2 for different models :-

```
plot(reg_summary$rsq,type = "l",ylab = expression(RSS))
```



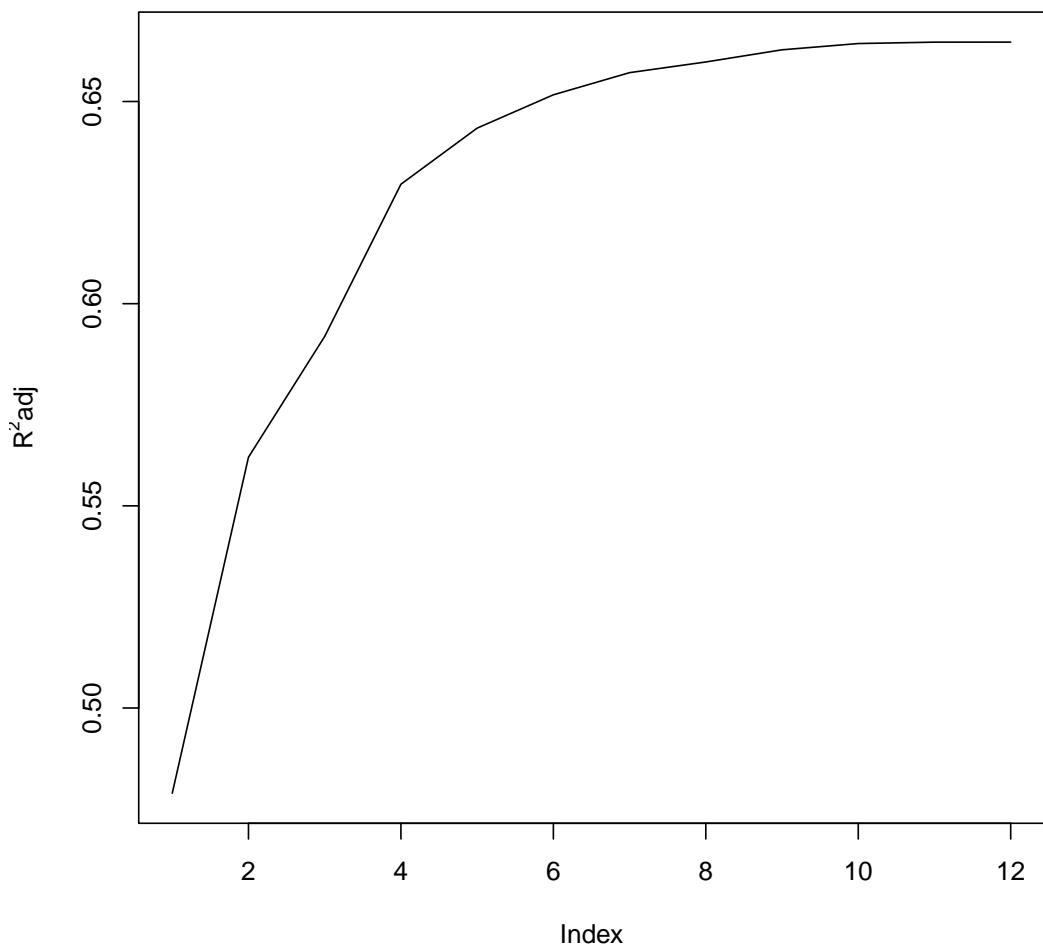
2. Value of Residual Sum of squares(RSS) :-

```
plot(reg_summary$rss,type = "l",ylab = expression(RSS))
```



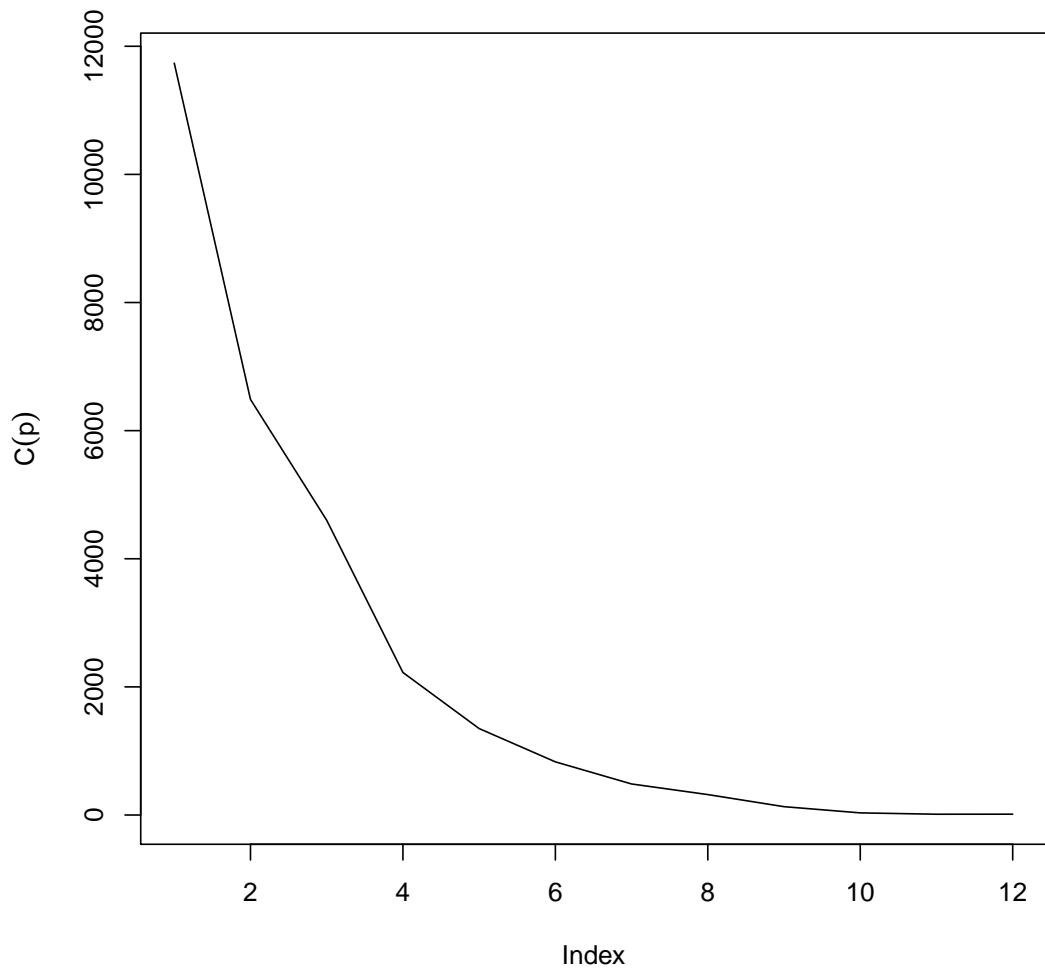
3. That of the adjusted- R^2 :-

```
plot(reg_summary$adjr2,type = "l",ylab = expression(paste(R^2,"adj")))
```



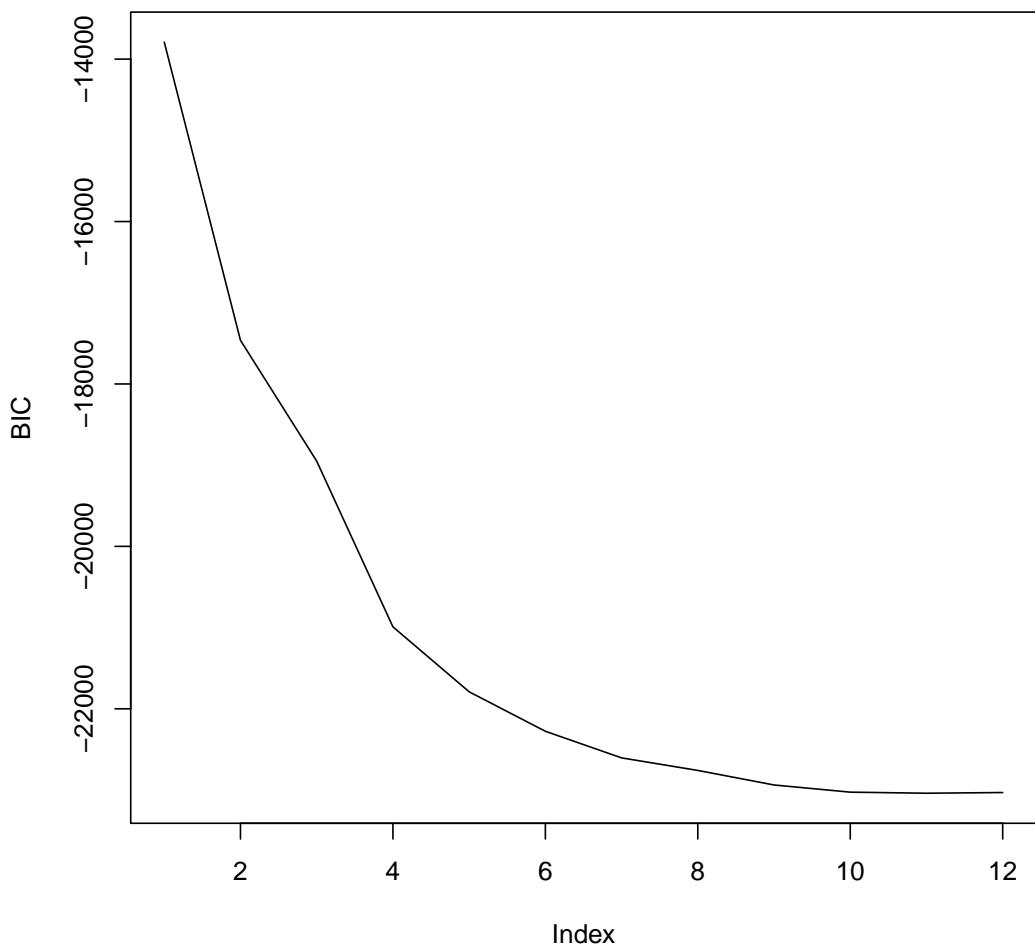
4. Mallow's $- C_p$:-

```
plot(reg_summary$cp,type = "l",ylab = expression(C(p)))
```



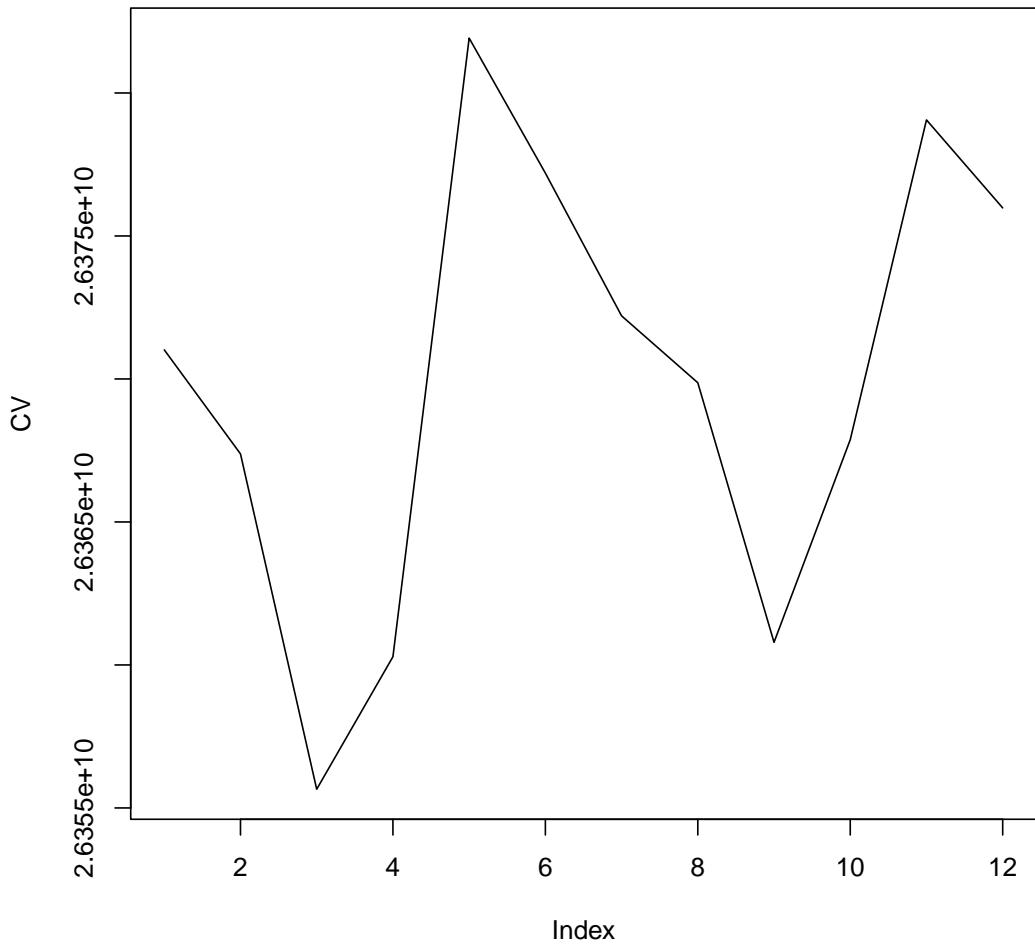
5. *BIC* values :-

```
plot(reg_summary_bwd$bic,type = "l",ylab = "BIC")
```



6. 5 – fold CV values :-

```
plot(cv.error.12,type = "l",ylab = "CV")
```



Hence, the best subset selection selects the model :-

```
opt_ind = which.min(cv.error.12)
opt_ind
[1] 3

glm.fit.opt = glm(price~. ,data=Data_House_2[,c(1,ind_num[[opt_ind]])])
glm.fit.opt

Call: glm(formula = price ~ ., data = Data_House_2[, c(1, ind_num[[opt_ind]])])

Coefficients:
(Intercept)      grade    sqft_above     yr_built
 5866648.9     164421.2       93.5      -3434.7

Degrees of Freedom: 21184 Total (i.e. Null); 21181 Residual
```

Null Deviance: 1.664e+15
Residual Deviance: 6.791e+14 AIC: 572600