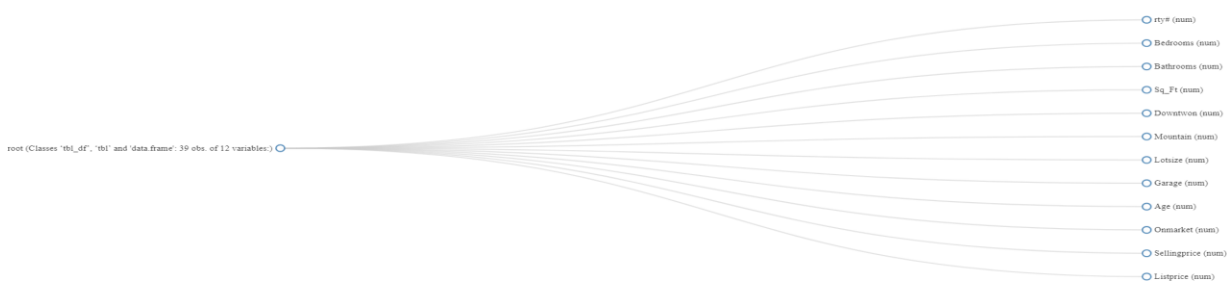
Appendix:

EDA Analysis:

```r
# EDA analysis of the data (to summarize all the data)

#Read ski.xlxs data into skiData
install.packages("readxl")
library(readxl)
ski=read_excel("ski.xlsx")
skiData=ski[-1]
```

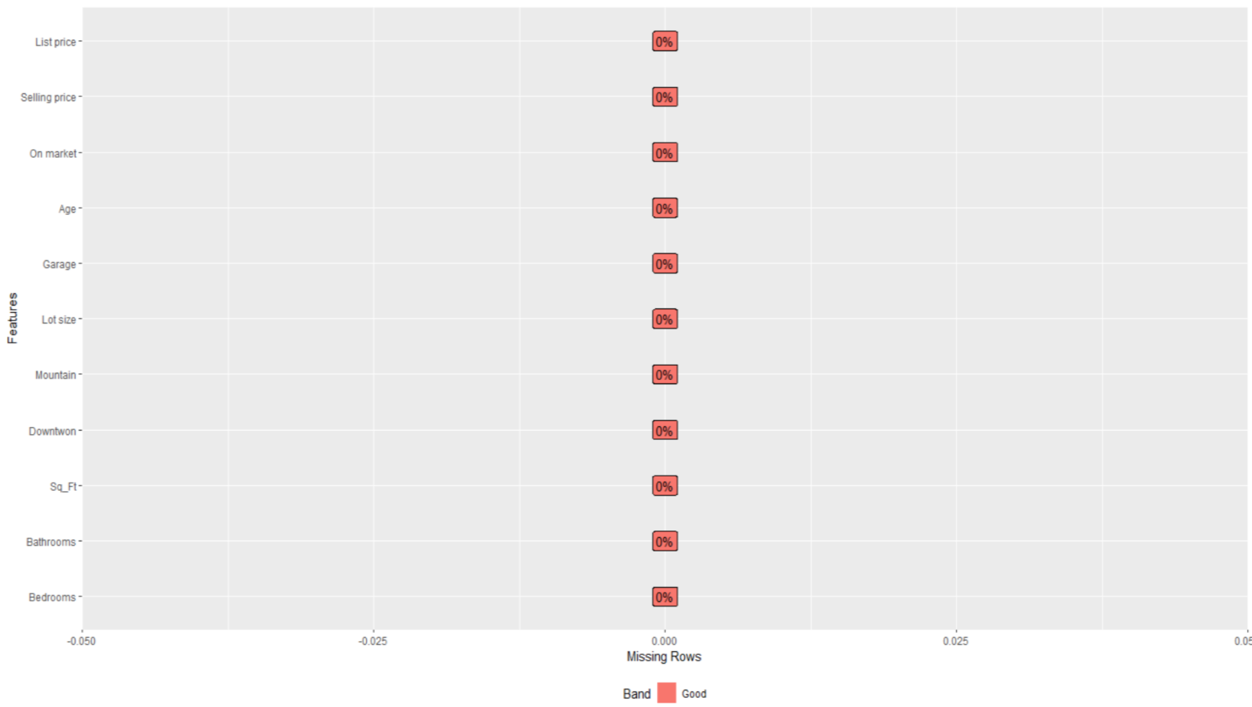
```r
#install required packages and libraries

install.packages("funModeling")
install.packages("tidyverse")
install.packages("Hmisc")
install.packages("DataExplorer")
library(funModeling)
library(tidyverse)
library(Hmisc)
library(DataExplorer)
```

```r
plot_str(ski)    #show data structure
```




```
> glimpse(ski)     #show data overview
Observations: 39
Variables: 12
$ `rty #`          <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18,...
$ Bedrooms         <dbl> 3, 3, 3, 3, 4, 3, 2, 3, 3, 2, 3, 3, 4, 3, 4, 2, 4, 3, 3, 3, 3,...
$ Bathrooms        <dbl> 2.00, 2.00, 2.50, 2.00, 2.50, 2.25, 1.00, 2.00, 2.00, 2.00, 2....
$ Sq_Ft            <dbl> 1771, 1213, 1884, 1922, 1858, 1948, 1200, 2002, 1160, 1300, 18...
$ Downtwon         <dbl> 15, 5, 2, 1, 0, 1, 52, 2, 15, 3, 0, 7, 3, 2, 31, 5, 0, 20, 20,...
$ Mountain         <dbl> 20, 1, 7, 6, 5, 6, 48, 7, 20, 2, 5, 2, 8, 7, 36, 10, 5, 15, 15...
$ `Lot size`       <dbl> 0.230, 0.170, 0.180, 0.290, 0.520, 21.910, 40.000, 0.180, 2.52...
$ Garage           <dbl> 2, 1, 2, 1, 2, 2, 2, 1, 0, 2, 2, 2, 2, 1, 2, 3, 0, 2, 0, 0, 2,...
$ Age              <dbl> 4, 5, 16, 80, 9, 20, 21, 18, 17, 67, 18, 3, 19, 14, 19, 26, 23...
$ `On market`      <dbl> 127, 98, 105, 103, 39, 403, 211, 126, 296, 158, 191, 150, 308,...
$ `Selling price`  <dbl> 365.0, 400.0, 408.0, 410.0, 448.0, 545.0, 315.0, 424.0, 326.0,...
$ `List price`     <dbl> 368.0, 406.0, 418.0, 417.0, 457.5, 550.0, 335.0, 429.0, 365.0,...
>
```

```
plot_missing(skiData)  #show missing data
```

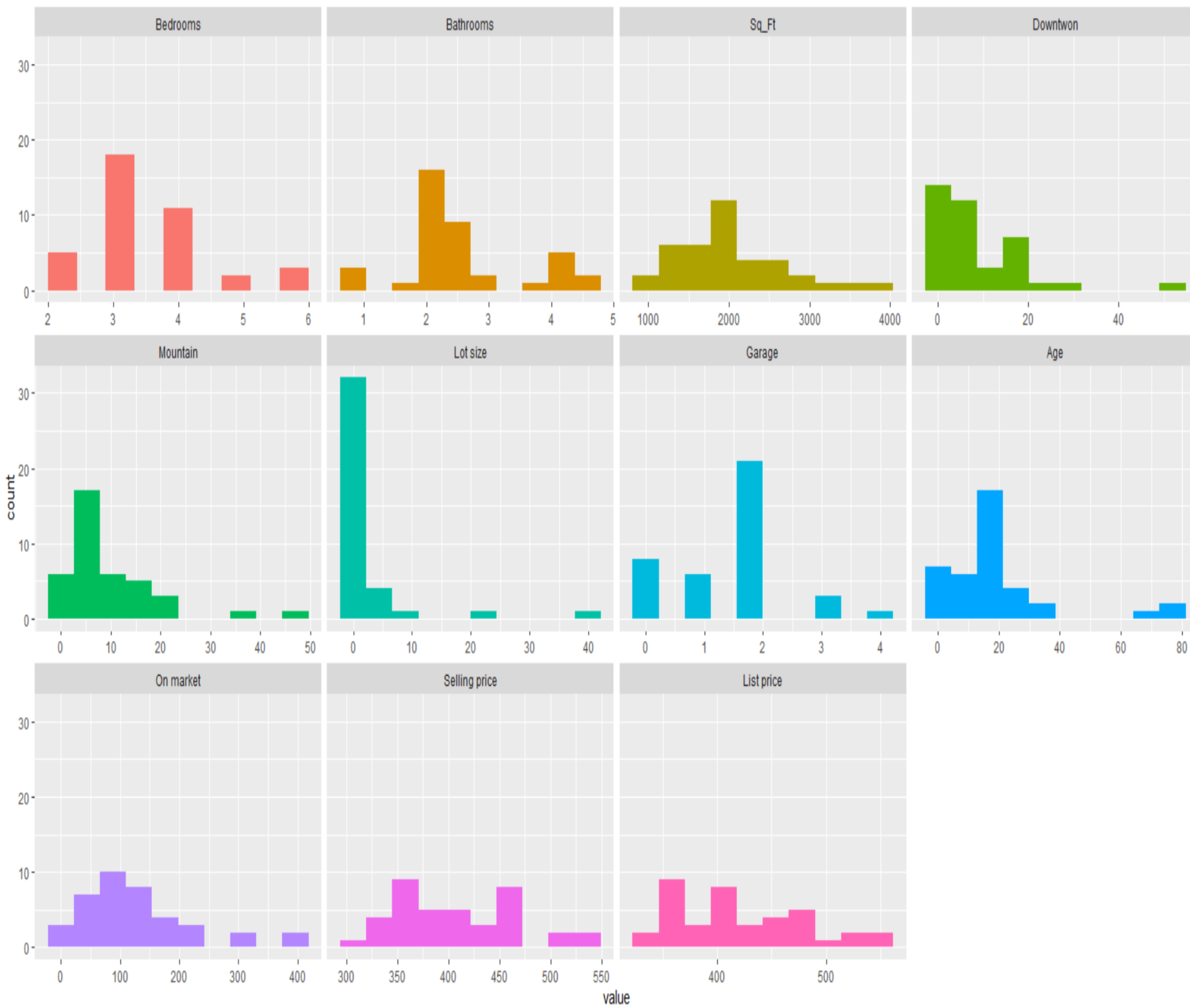



```
> profiling_num(skiData)
        variable        mean     std_dev variation_coef     p_01      p_05    p_25    p_50     p_75
1       Bedrooms    3.487179   1.0481009      0.3005583   2.0000    2.0000    3.00    3.00    4.000
2      Bathrooms    2.512821   0.9335640      0.3715204   1.0000    1.0000    2.00    2.25    2.625
3          Sq_Ft 2003.794872 664.4719383      0.3316068 995.3600 1148.0000 1550.00 1922.00 2290.000
4       Downtwon    8.692308  10.6774574      1.2283801   0.0000    0.0000    2.00    5.00   13.500
5       Mountain    9.666667   9.3761549      0.9699471   1.3800    2.0000    5.00    7.00   12.500
6       Lot size    2.596667   7.2502668      2.7921438   0.1038    0.1235    0.23    0.34    1.100
7         Garage    1.564103   0.9945872      0.6358836   0.0000    0.0000    1.00    2.00    2.000
8            Age   19.461538  18.0712714      0.9285634   3.0000    3.9000   10.00   16.00   20.500
9      On market  131.000000  95.2067003      0.7267687  17.5200   20.9000   69.50  105.00  165.000
10 Selling price  409.943590  58.7972242      0.1434276 319.1800  337.7000  362.75  400.00  458.250
11    List price  420.443590  58.5369869      0.1392267 336.7100  348.8600  367.75  409.00  464.250
       p_95     p_99   skewness  kurtosis     iqr                range_98                      range_80
1     6.000   6.0000  0.8678691  3.498607   1.000                  [2, 6]                        [2, 5]
2     4.050   4.6550  0.8040347  3.006134   0.625              [1, 4.655]                     [1.95, 4]
3  3282.000 3759.1000  0.8633321  3.652949 740.000        [995.36, 3759.1]                [1210.4, 2794]
4    25.600  44.0200  2.0895276  8.243992  11.500 [0, 44.0199999999999]                       [0, 20]
5    21.600  43.4400  2.3954358  9.470721   7.500           [1.38, 43.44]                       [2, 20]
6    11.641  33.1258  4.1781239 20.537496   0.870       [0.1038, 33.1258] [0.161, 3.46400000000001]
7     3.000   3.6200 -0.1792244  2.682989   1.000               [0, 3.62]                      [0, 2.2]
8    68.100  79.2400  2.2512647  7.825822  10.500              [3, 79.24]                       [4, 31]
9   317.500 408.5800  1.3976895  4.771464  95.500         [17.52, 408.58]                  [35.8, 241.6]
10  521.000 539.3000  0.5247684  2.450196  95.500         [319.18, 539.3]                    [345, 477]
11  528.800 548.1000  0.5827667  2.407303  96.500         [336.71, 548.1]                 [359.58, 496.4]
>
```

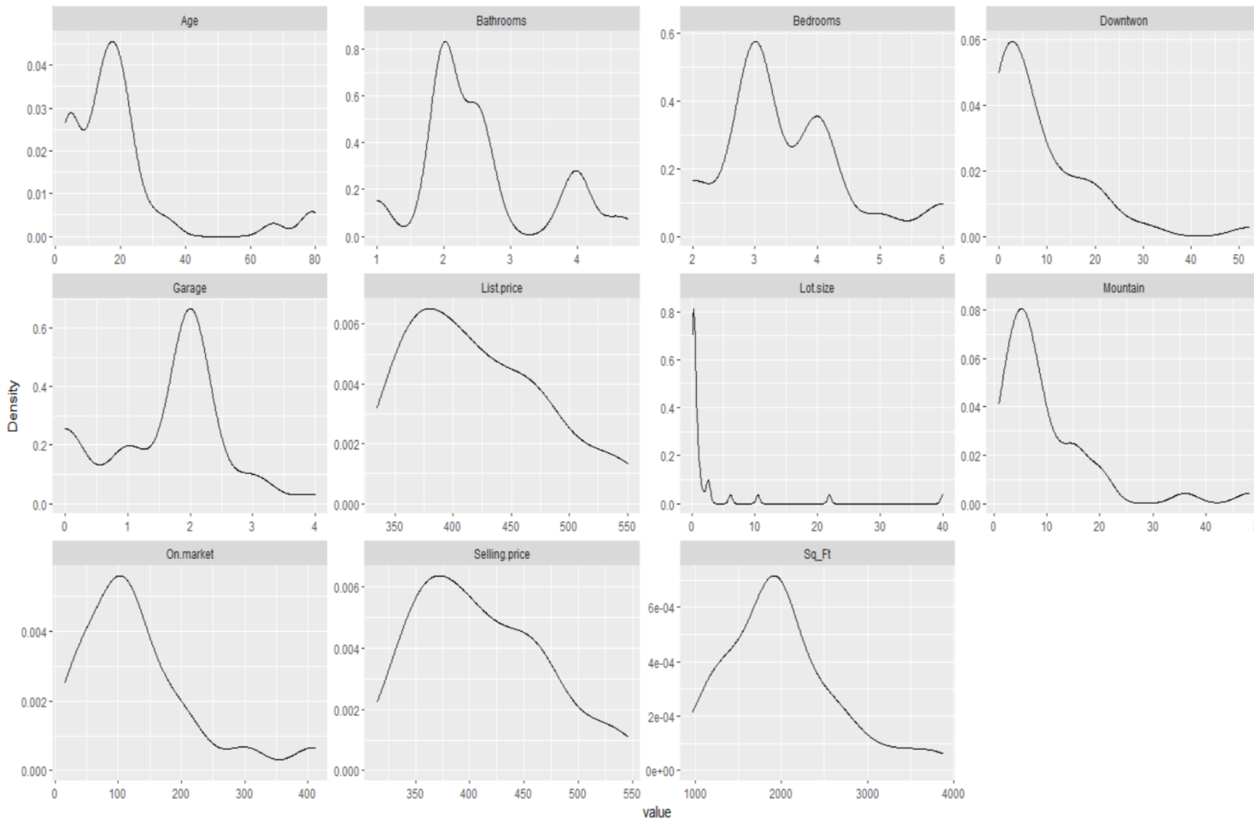
```
> describe(skiData)
skiData

 11  Variables      39  Observations
--------------------------------------------------------------------------------------------------------------
Bedrooms
       n  missing distinct     Info     Mean      Gmd
      39        0        5    0.877    3.487    1.101

lowest : 2 3 4 5 6, highest: 2 3 4 5 6

Value          2     3     4     5     6
Frequency      5    18    11     2     3
Proportion 0.128 0.462 0.282 0.051 0.077
--------------------------------------------------------------------------------------------------------------
Bathrooms
       n  missing distinct     Info     Mean      Gmd      .05      .10      .25      .50      .75      .90
      39        0       10    0.929    2.513   0.9838    1.000    1.950    2.000    2.250    2.625    4.000
     .95
   4.050

lowest : 1.00 1.75 2.00 2.25 2.50, highest: 2.75 3.75 4.00 4.50 4.75

Value       1.00  1.75  2.00  2.25  2.50  2.75  3.75  4.00  4.50  4.75
Frequency      3     1    15     1     9     2     1     5     1     1
Proportion 0.077 0.026 0.385 0.026 0.231 0.051 0.026 0.128 0.026 0.026
--------------------------------------------------------------------------------------------------------------
Age
       n  missing distinct     Info     Mean      Gmd      .05      .10      .25      .50      .75      .90
      39        0       22    0.995    19.46    16.61      3.9      4.0     10.0     16.0     20.5     31.0
     .95
    68.1

lowest :   3   4   5   9 11, highest: 30 35 67 78 80
--------------------------------------------------------------------------------------------------------------
On market
       n  missing distinct     Info     Mean      Gmd      .05      .10      .25      .50      .75      .90
      39        0       37        1      131    100.2     20.9     35.8     69.5    105.0    165.0    241.6
     .95
   317.5

lowest :  16  20  21  23  39, highest: 228 296 308 403 412
--------------------------------------------------------------------------------------------------------------
Selling price
       n  missing distinct     Info     Mean      Gmd      .05      .10      .25      .50      .75      .90
      39        0       33    0.999    409.9    67.17    337.7    345.0    362.8    400.0    458.2    477.0
     .95
   521.0

lowest : 315 326 339 345 350, highest: 470 505 520 530 545
--------------------------------------------------------------------------------------------------------------
List price
       n  missing distinct     Info     Mean      Gmd      .05      .10      .25      .50      .75      .90
      39        0       35        1    420.4    66.73    348.9    359.6    367.8    409.0    464.2    496.4
     .95
   528.8

lowest : 335.0 339.5 349.9 357.9 360.0, highest: 490.0 522.0 527.0 545.0 550.0
--------------------------------------------------------------------------------------------------------------
```

```
Sq_Ft
       n  missing distinct     Info     Mean      Gmd      .05      .10      .25      .50      .75      .90
      39        0       38        1     2004    733.9     1148     1210     1550     1922     2290     2794
     .95
    3282

lowest :  968 1040 1160 1200 1213, highest: 2755 2950 3250 3570 3875
------------------------------------------------------------------------------------------------------------
Downtwon
       n  missing distinct     Info     Mean      Gmd      .05      .10      .25      .50      .75      .90
      39        0       13    0.987    8.692    10.45      0.0      0.0      2.0      5.0     13.5     20.0
     .95
    25.6

lowest :  0  1  2  3  5, highest: 15 20 25 31 52

Value          0     1     2     3     5     7    10    12    15    20    25    31    52
Frequency      7     2     5     3     5     4     2     1     3     4     1     1     1
Proportion 0.179 0.051 0.128 0.077 0.128 0.103 0.051 0.026 0.077 0.103 0.026 0.026 0.026
------------------------------------------------------------------------------------------------------------
Mountain
       n  missing distinct     Info     Mean      Gmd      .05      .10      .25      .50      .75      .90
      39        0       12    0.981    9.667    8.764      2.0      2.0      5.0      7.0     12.5     20.0
     .95
    21.6

lowest :  1  2  3  5  6, highest: 10 15 20 36 48

Value          1     2     3     5     6     7     8    10    15    20    36    48
Frequency      1     5     2     9     2     4     2     4     5     3     1     1
Proportion 0.026 0.128 0.051 0.231 0.051 0.103 0.051 0.103 0.128 0.077 0.026 0.026
------------------------------------------------------------------------------------------------------------
Lot size
       n  missing distinct     Info     Mean      Gmd      .05      .10      .25      .50      .75      .90
      39        0       29    0.998    2.597    4.295   0.1235   0.1610   0.2300   0.3400   1.1000   3.4640
     .95
 11.6410

lowest :  0.100  0.110  0.125  0.170  0.180, highest:  2.800  6.120 10.500 21.910 40.000

Value        0.1   0.2   0.3   0.4   0.5   0.7   0.8   1.0   1.2   1.3   1.9   2.5   2.8   6.1  10.5  21.9
Frequency      4    13     3     1     2     3     1     2     1     1     1     2     1     1     1     1
Proportion 0.103 0.333 0.077 0.026 0.051 0.077 0.026 0.051 0.026 0.026 0.026 0.051 0.026 0.026 0.026 0.026

Value       40.0
Frequency      1
Proportion 0.026

For the frequency table, variable is rounded to the nearest 0.1
------------------------------------------------------------------------------------------------------------
Garage
       n  missing distinct     Info     Mean      Gmd
      39        0        5    0.832    1.564    1.047

lowest : 0 1 2 3 4, highest: 0 1 2 3 4

Value          0     1     2     3     4
Frequency      8     6    21     3     1
Proportion 0.205 0.154 0.538 0.077 0.026
------------------------------------------------------------------------------------------------------------
```

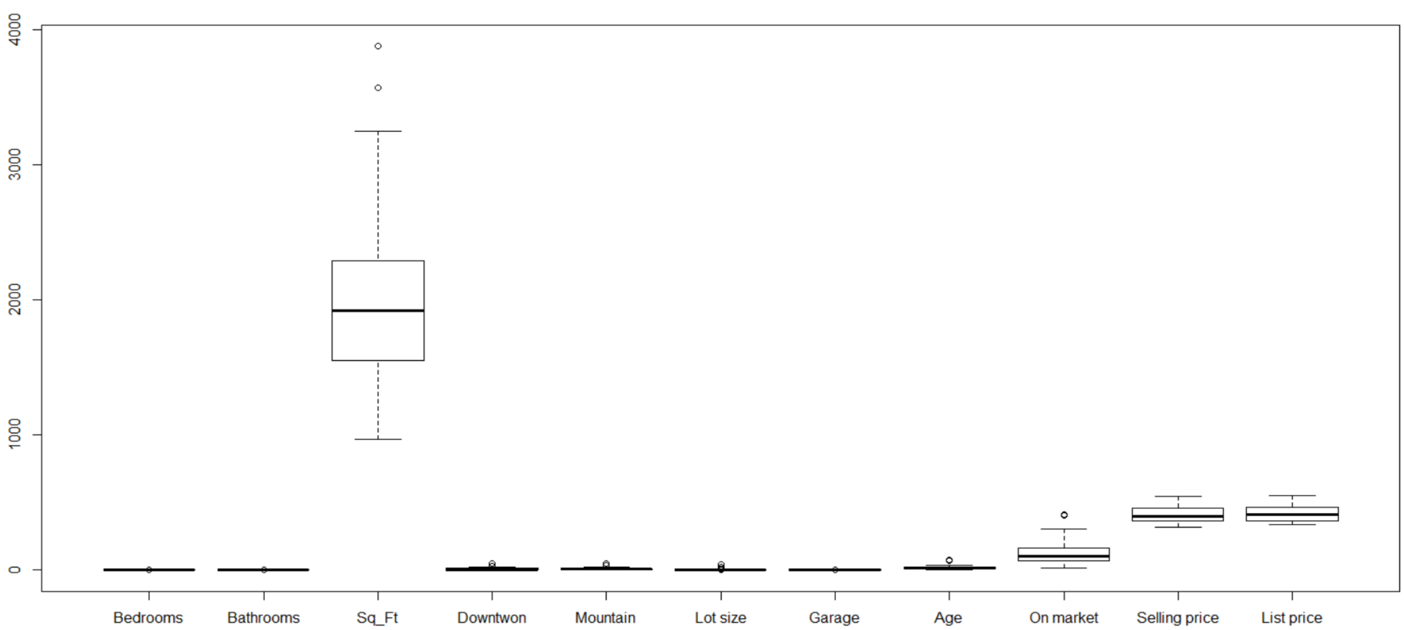
plot_num(skiData)   #plot histogram of all numeric data



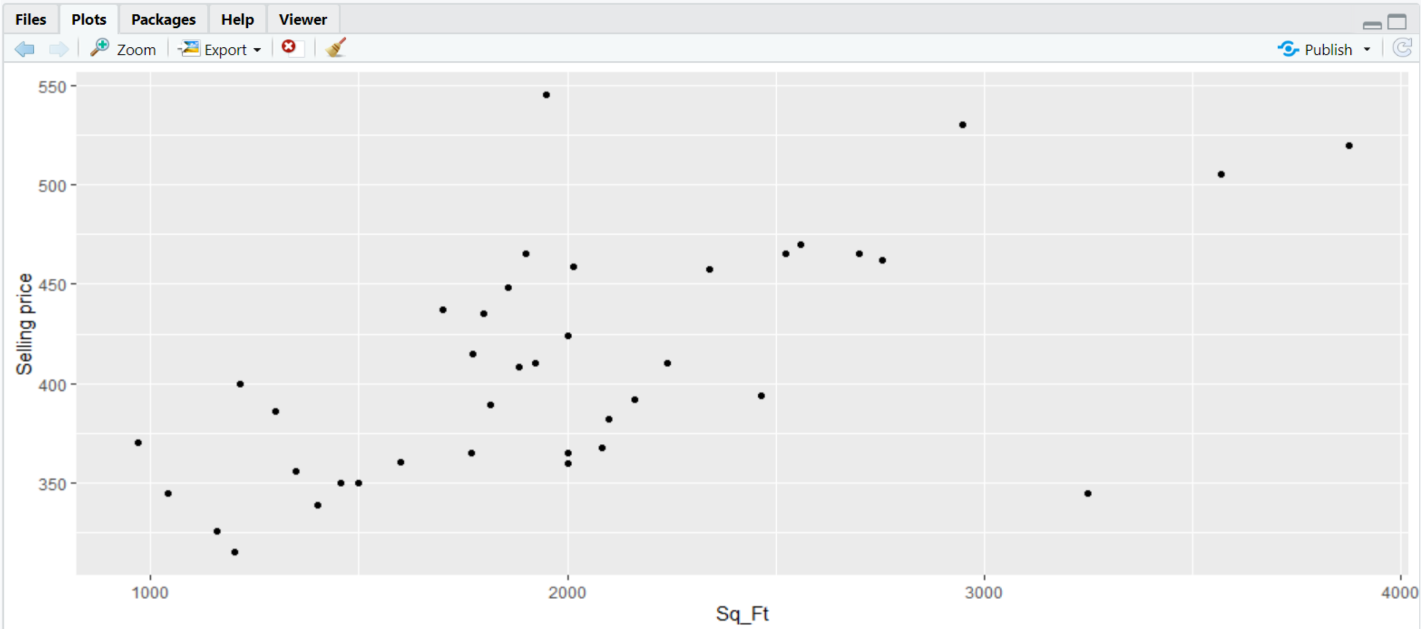
plot_density(skiData) #show density plot of all data



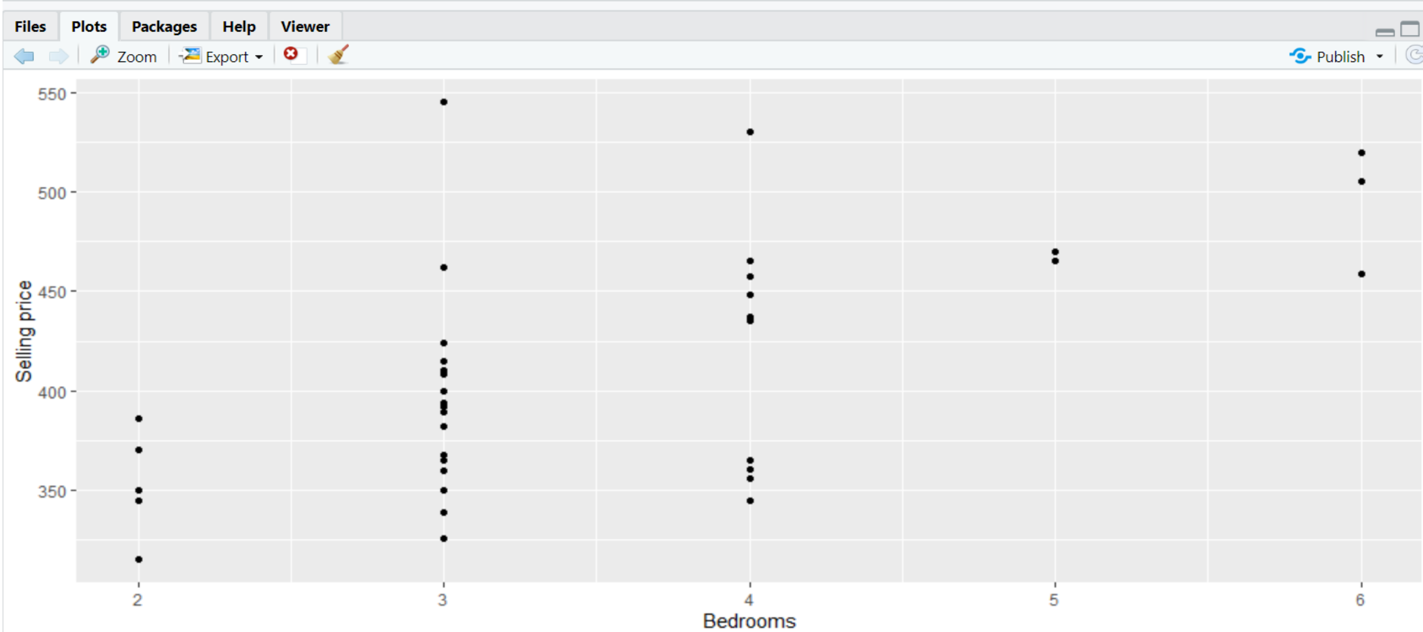

```
#Boxplot
boxplot(skiData)    #boxplot of all data
```



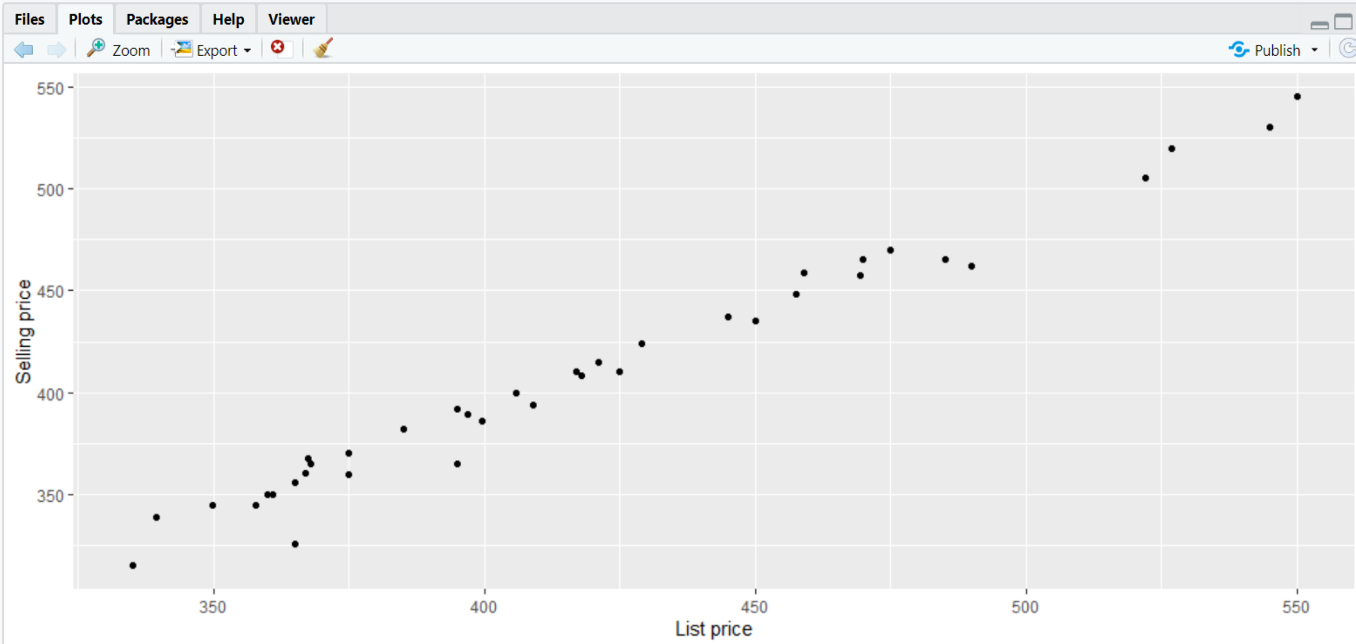
```
> #scatter plots
> ggplot(data = ski) +
+   geom_point(mapping = aes(x = Sq_Ft, y = `Selling price`))
> 
```



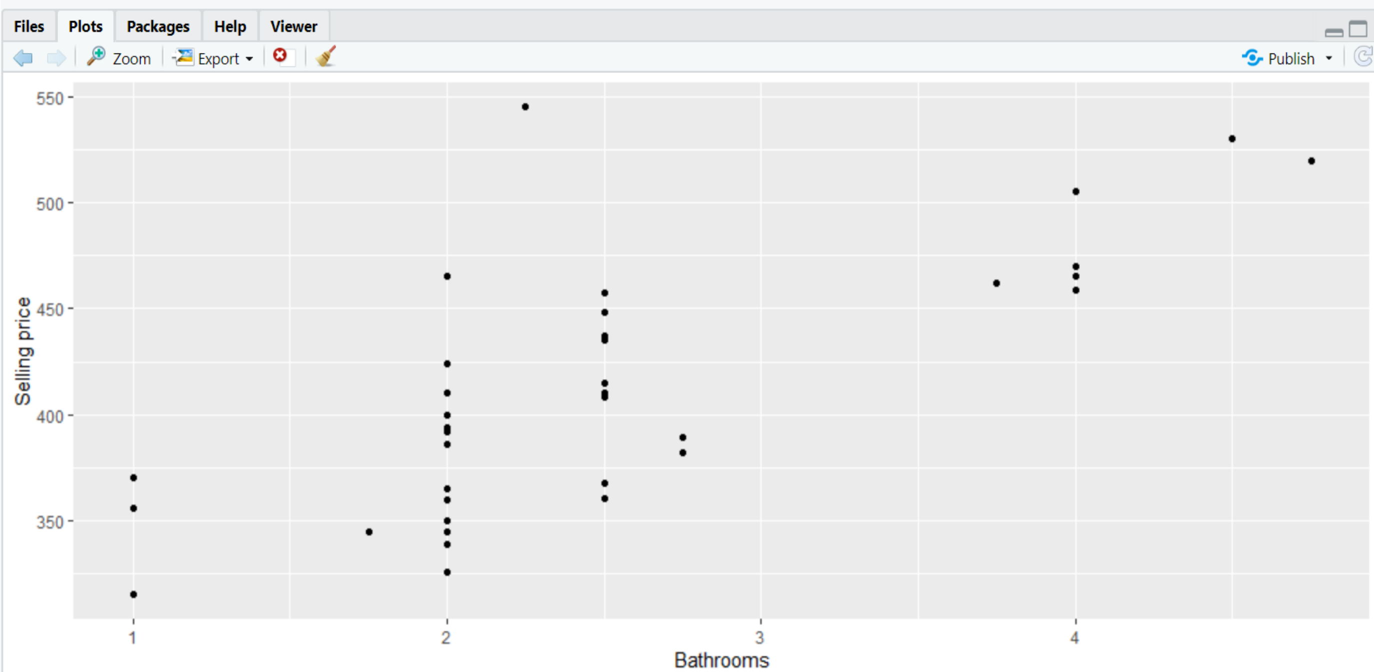

```
> ggplot(data = ski) +
+   geom_point(mapping = aes(x = Bedrooms, y = `Selling price`))
> 
```



```
> ggplot(data = ski) +
+   geom_point(mapping = aes(x =`List price`, y = `Selling price`))
>
```




```
> ggplot(data = ski) +
+   geom_point(mapping = aes(x = Bathrooms, y = `Selling price`))
>
```



pairs(skiData) #scatter plot of all the data

```
> corMatrix
                Bedrooms   Bathrooms      Sq_Ft    Downtwon     Mountain     Lot size      Garage
Bedrooms      1.00000000 0.719609738  0.6770924 -0.01682229 -0.087476894 -0.233502216 -0.11910326
Bathrooms     0.71960974 1.000000000  0.7553783 -0.08539395 -0.233998536 -0.281124786  0.30376734
Sq_Ft         0.67709241 0.755378309  1.0000000  0.15466525  0.045590103 -0.209367986  0.24737977
Downtwon     -0.01682229 -0.085393946 0.1546652  1.00000000  0.900294003  0.524735673  0.15306569
Mountain     -0.08747689 -0.233998536 0.0455901  0.90029400  1.000000000  0.569027571  0.04609179
Lot size     -0.23350222 -0.281124786 -0.2093680 0.52473567  0.569027571  1.000000000  0.12226652
Garage       -0.11910326 0.303767338  0.2473798  0.15306569  0.046091790  0.122266519  1.00000000
Age          -0.22615016 -0.194951584 -0.2067820 -0.17122346 -0.034945032  0.008797268 -0.09539484
On market    -0.23893182 0.006143599 -0.1166832 -0.08169924  0.006691908  0.313551699  0.27012966
Selling price 0.61097980 0.742681869  0.6371482 -0.27076285 -0.402171776 -0.044227489  0.32672161
List price    0.59924120 0.747440292  0.6412548 -0.23139986 -0.374625744 -0.026953243  0.34666111
                     Age   On market Selling price  List price
Bedrooms     -0.226150164 -0.238931819   0.61097980  0.59924120
Bathrooms    -0.194951584  0.006143599   0.74268187  0.74744029
Sq_Ft        -0.206782024 -0.116683167   0.63714824  0.64125477
Downtwon     -0.171223456 -0.081699244  -0.27076285 -0.23139986
Mountain     -0.034945032  0.006691908  -0.40217178 -0.37462574
Lot size      0.008797268  0.313551699  -0.04422749 -0.02695324
Garage       -0.095394837  0.270129660   0.32672161  0.34666111
Age           1.000000000  0.113767013  -0.19714627 -0.19318664
On market     0.113767013  1.000000000   0.11046973  0.14138228
Selling price -0.197146266 0.110469731   1.00000000  0.98992618
List price   -0.193186636  0.141382280   0.98992618  1.00000000
> |
```

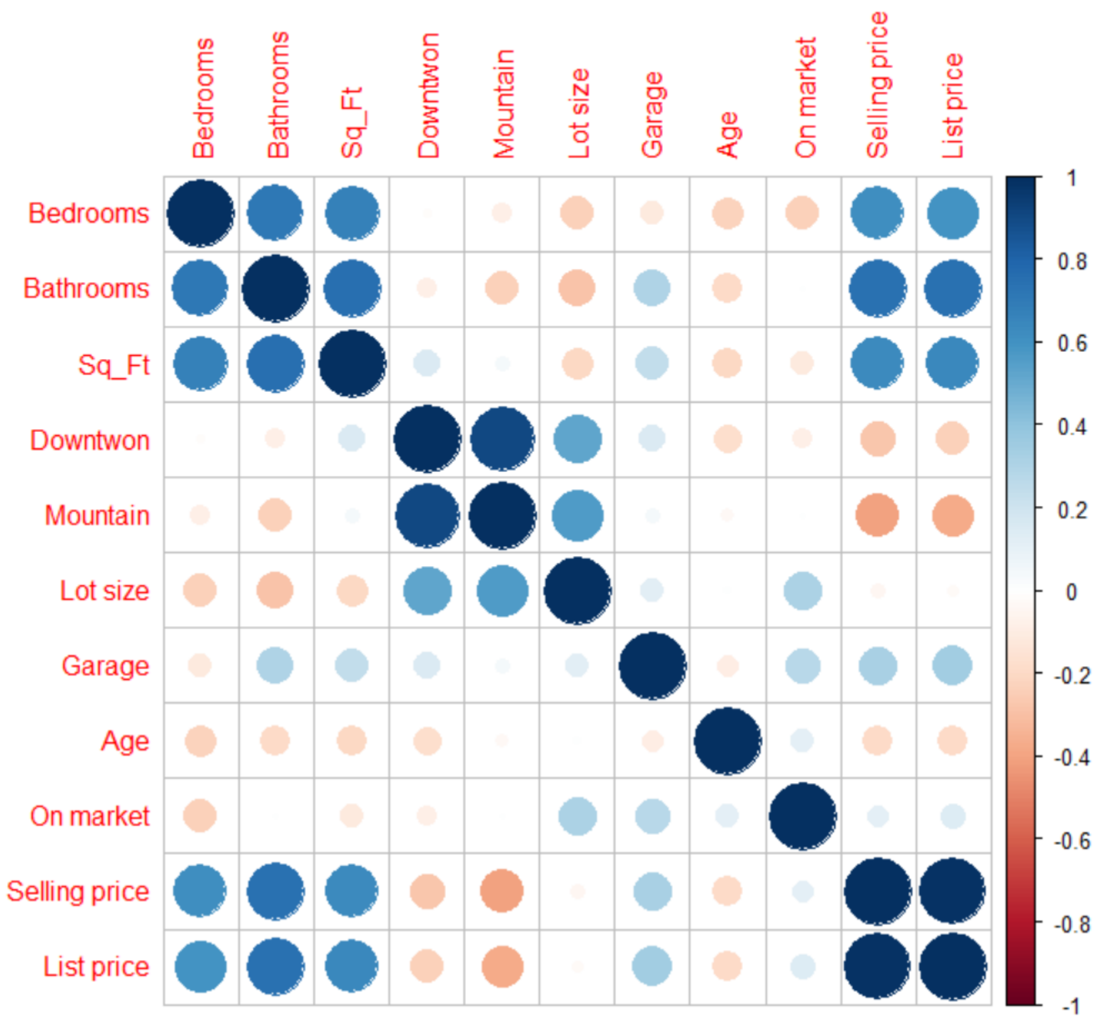
```
> corMatrix.rcorr= rcorr(as.matrix(skiData))
> corMatrix.rcorr
              Bedrooms Bathrooms Sq_Ft Downtwon Mountain Lot size Garage   Age On market Selling price
Bedrooms          1.00      0.72  0.68    -0.02    -0.09    -0.23  -0.12 -0.23     -0.24          0.61
Bathrooms         0.72      1.00  0.76    -0.09    -0.23    -0.28   0.30 -0.19      0.01          0.74
Sq_Ft             0.68      0.76  1.00     0.15     0.05    -0.21   0.25 -0.21     -0.12          0.64
Downtwon         -0.02     -0.09  0.15     1.00     0.90     0.52   0.15 -0.17     -0.08         -0.27
Mountain         -0.09     -0.23  0.05     0.90     1.00     0.57   0.05 -0.03      0.01         -0.40
Lot size         -0.23     -0.28 -0.21     0.52     0.57     1.00   0.12  0.01      0.31         -0.04
Garage           -0.12      0.30  0.25     0.15     0.05     0.12   1.00 -0.10      0.27          0.33
Age              -0.23     -0.19 -0.21    -0.17    -0.03     0.01  -0.10  1.00      0.11         -0.20
On market        -0.24      0.01 -0.12    -0.08     0.01     0.31   0.27  0.11      1.00          0.11
Selling price     0.61      0.74  0.64    -0.27    -0.40    -0.04   0.33 -0.20      0.11          1.00
List price        0.60      0.75  0.64    -0.23    -0.37    -0.03   0.35 -0.19      0.14          0.99
              List price
Bedrooms            0.60
Bathrooms           0.75
Sq_Ft               0.64
Downtwon           -0.23
Mountain           -0.37
Lot size           -0.03
Garage              0.35
Age                -0.19
On market           0.14
Selling price       0.99
List price          1.00

n= 39
```

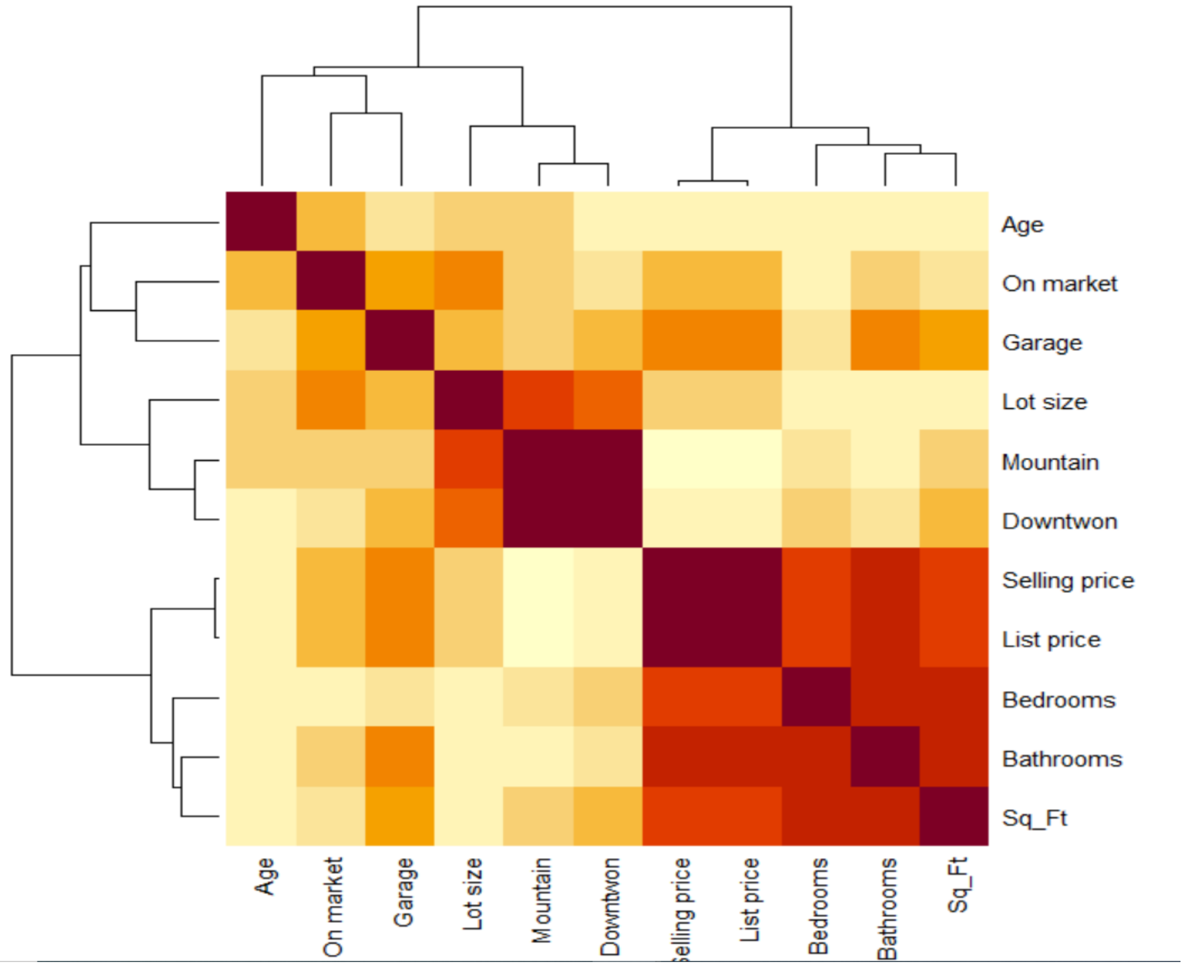
```
> corMatrix.rcorr$P       #P values
                 Bedrooms    Bathrooms        Sq_Ft     Downtwon     Mountain     Lot size      Garage
Bedrooms                NA 2.438010e-07 2.208953e-06 9.190391e-01 5.964336e-01 0.1525332474 0.47018599
Bathrooms     2.438010e-07           NA 2.733551e-08 6.052398e-01 1.516382e-01 0.0829785397 0.06010773
Sq_Ft         2.208953e-06 2.733551e-08           NA 3.471496e-01 7.828628e-01 0.2008283666 0.12893060
Downtwon      9.190391e-01 6.052398e-01 3.471496e-01           NA 6.217249e-15 0.0006049383 0.35221968
Mountain      5.964336e-01 1.516382e-01 7.828628e-01 6.217249e-15           NA 0.0001568188 0.78053180
Lot size      1.525332e-01 8.297854e-02 2.008284e-01 6.049383e-04 1.568188e-04           NA 0.45839012
Garage        4.701860e-01 6.010773e-02 1.289306e-01 3.522197e-01 7.805318e-01 0.4583901226         NA
Age           1.662457e-01 2.343162e-01 2.065741e-01 2.973089e-01 8.327345e-01 0.9576104660 0.56348162
On market     1.429497e-01 9.703903e-01 4.793179e-01 6.209958e-01 9.677490e-01 0.0519253157 0.09628162
Selling price 3.609446e-05 6.190049e-08 1.291908e-05 9.547368e-02 1.114796e-02 0.7892035173 0.04234814
List price    5.558683e-05 4.582433e-08 1.089991e-05 1.563672e-01 1.878815e-02 0.8706161223 0.03061801
                    Age  On market Selling price   List price
Bedrooms      0.1662457 0.14294971  3.609446e-05 5.558683e-05
Bathrooms     0.2343162 0.97039029  6.190049e-08 4.582433e-08
Sq_Ft         0.2065741 0.47931791  1.291908e-05 1.089991e-05
Downtwon      0.2973089 0.62099578  9.547368e-02 1.563672e-01
Mountain      0.8327345 0.96774898  1.114796e-02 1.878815e-02
Lot size      0.9576105 0.05192532  7.892035e-01 8.706161e-01
Garage        0.5634816 0.09628162  4.234814e-02 3.061801e-02
Age                  NA 0.49044374  2.289877e-01 2.386619e-01
On market     0.4904437         NA  5.031822e-01 3.906000e-01
Selling price 0.2289877 0.50318216            NA 0.000000e+00
List price    0.2386619 0.39060001  0.000000e+00           NA
> 
```

```
> corMatrix.rcorr$r       #correlation coefficients
                 Bedrooms    Bathrooms      Sq_Ft    Downtwon     Mountain     Lot size      Garage
Bedrooms       1.00000000  0.719609738  0.6770924 -0.01682229 -0.087476894 -0.233502216 -0.11910326
Bathrooms      0.71960974  1.000000000  0.7553783 -0.08539395 -0.233998536 -0.281124786  0.30376734
Sq_Ft          0.67709241  0.755378309  1.0000000  0.15466525  0.045590103 -0.209367986  0.24737977
Downtwon      -0.01682229 -0.085393946  0.1546652  1.00000000  0.900294003  0.524735673  0.15306569
Mountain      -0.08747689 -0.233998536  0.0455901  0.90029400  1.000000000  0.569027571  0.04609179
Lot size      -0.23350222 -0.281124786 -0.2093680  0.52473567  0.569027571  1.000000000  0.12226652
Garage        -0.11910326  0.303767338  0.2473798  0.15306569  0.046091790  0.122266519  1.00000000
Age           -0.22615016 -0.194951584 -0.2067820 -0.17122346 -0.034945032  0.008797268 -0.09539484
On market     -0.23893182  0.006143599 -0.1166832 -0.08169924  0.006691908  0.313551699  0.27012966
Selling price  0.61097980  0.742681869  0.6371482 -0.27076285 -0.402171776 -0.044227489  0.32672161
List price     0.59924120  0.747440292  0.6412548 -0.23139986 -0.374625744 -0.026953243  0.34666111
                       Age    On market Selling price  List price
Bedrooms      -0.226150164 -0.238931819    0.61097980  0.59924120
Bathrooms     -0.194951584  0.006143599    0.74268187  0.74744029
Sq_Ft         -0.206782024 -0.116683167    0.63714824  0.64125477
Downtwon      -0.171223456 -0.081699244   -0.27076285 -0.23139986
Mountain      -0.034945032  0.006691908   -0.40217178 -0.37462574
Lot size       0.008797268  0.313551699   -0.04422749 -0.02695324
Garage        -0.095394837  0.270129660    0.32672161  0.34666111
Age            1.000000000  0.113767013   -0.19714627 -0.19318664
On market      0.113767013  1.000000000    0.11046973  0.14138228
Selling price -0.197146266  0.110469731    1.00000000  0.98992618
List price    -0.193186636  0.141382280    0.98992618  1.00000000
> 
```

```
              Selling price List price
Bedrooms      0.0000        0.0000
Bathrooms     0.0000        0.0000
Sq_Ft         0.0000        0.0000
Downtwon      0.0955        0.1564
Mountain      0.0111        0.0188
Lot size      0.7892        0.8706
Garage        0.0423        0.0306
Age           0.2290        0.2387
On market     0.5032        0.3906
Selling price               0.0000
List price    0.0000
```

```
install.packages("corrplot")
library(corrplot)
corrplot(corMatrix)      #visualizing the correlation matrix
```






```
#correlation coefficients for Selling Price against all variables

cor(skiData[-10],skiData$`Selling price`)

""""
> cor(skiData[-10],skiData$`Selling price`)
                  [,1]
Bedrooms     0.61097980
Bathrooms    0.74268187
Sq_Ft        0.63714824
Downtwon    -0.27076285
Mountain    -0.40217178
Lot size    -0.04422749
Garage       0.32672161
Age         -0.19714627
On market    0.11046973
List price   0.98992618
>
""""
```

```
"""
############ Analysis per Variable Vs Selling Price

> #Selling price per bedroom
> data = mutate(skiData, Selling_price_per_bedroom =skiData$`Selling price`/skiData$Bedrooms )
> median(data$Selling_price_per_bedroom)
[1] 121.6667

> #Selling price per bathroom
> data =  mutate(skiData, Selling_price_per_bathroom =skiData$`Selling price`/skiData$Bathrooms )
> median(data$Selling_price_per_bathroom)
[1] 174.8

> #Selling price per Sq_Ft
> data =  mutate(skiData, Selling_price_per_sqft =skiData$`Selling price`/skiData$Sq_Ft )
> median(data$Selling_price_per_sqft)
[1] 0.214207

> #Selling price per Downtwon
> data =  mutate(skiData, Selling_price_per_dwntn =skiData$`Selling price`/skiData$Downtwon )
> median(data$Selling_price_per_dwntn)
[1] 75.71429

> #Selling price per Mountain
> data =  mutate(skiData, Selling_price_per_mntn =skiData$`Selling price`/skiData$Mountain )
> median(data$Selling_price_per_mntn)
[1] 67.14286

> #Selling price per Lot size
> data =  mutate(skiData, Selling_price_per_lotsize =skiData$`Selling price`/skiData$`Lot size` )
> median(data$Selling_price_per_lotsize)
[1] 1106.061

> #Selling price per Garage
> data =  mutate(skiData, Selling_price_per_garage =skiData$`Selling price`/skiData$Garage )
> median(data$Selling_price_per_garage)
[1] 228.75

> #Selling price per Age
> data =  mutate(skiData, Selling_price_per_age =skiData$`Selling price`/skiData$Age )
> median(data$Selling_price_per_age)
[1] 24.33333

> #Selling price per On Martket
> data =  mutate(skiData, Selling_price_per_onMarket =skiData$`Selling price`/skiData$`On market` )
> median(data$Selling_price_per_onMarket)
[1] 3.8125

> #Selling price per listPrice
> data =  mutate(skiData, Selling_price_per_listPrice =skiData$`Selling price`/skiData$`List price` )
> median(data$Selling_price_per_listPrice)
[1] 0.9798489
>
"""
```

Variable	Selling Price Per Variable
Bedrooms	121.6667
Bathrooms	174.8
Sq_Ft	0.214207
Downtwn	75.71429
Mountain	67.14286
Lot size	1106.061
Garage	228.75
Age	24.33333
On market	3.8125
List price	0.9798489

Multiple Linear Regression:

Model 1:

```
#Multiple Linear Regression
#Load ski data
#Delete the first column

ski=read_excel("ski.xlsx")
skiData=ski[-1]

#Store the columns separate
bed=skiData$Bedrooms
bath=skiData$Bathrooms
sqft=skiData$Sq_Ft
dwntn=skiData$Downtwon
mntn=skiData$Mountain
lot_size=skiData$'Lot size'
garage=skiData$Garage
age=skiData$Age
on_market=skiData$'On market'
sp=skiData$'Selling price'
lp=skiData$'List price'
```

```
#Model 1 with all variables as predictors

#Case 1 : (Using AIC stepwise regression both sides)

library(MASS)
fit <- lm(sp~ bed+bath+sqft+dwntn+mntn+lot_size+garage+age+on_market+lp,data=skiData)
step <- stepAIC(fit, direction="both")
step$anova # display results

""""
> fit <- lm(sp~ bed+bath+sqft+dwntn+mntn+lot_size+garage+age+on_market+lp,data=skiData)
> step <- stepAIC(fit, direction="both")
Start:  AIC=174.58
sp ~ bed + bath + sqft + dwntn + mntn + lot_size + garage + age +
    on_market + lp

            Df Sum of Sq     RSS    AIC
- mntn       1       0.0  1950.7 172.58
- bath       1       0.2  1950.9 172.59
- age        1      32.2  1982.9 173.22
- garage     1      37.3  1988.1 173.32
- bed        1      50.9  2001.6 173.59
- sqft       1      70.2  2020.9 173.96
<none>                    1950.7 174.58
- on_market  1     120.3  2071.1 174.92
- lot_size   1     179.2  2129.9 176.01
- dwntn      1     234.0  2184.7 177.00
- lp         1   15933.3 17884.0 259.00
```

```
Step:  AIC=172.58
sp ~ bed + bath + sqft + dwntn + lot_size + garage + age + on_market +
    lp

            Df Sum of Sq     RSS    AIC
- bath       1       0.2  1950.9 170.59
- age        1      33.1  1983.8 171.24
- garage     1      37.7  1988.4 171.33
- bed        1      56.3  2007.0 171.69
- sqft       1      80.0  2030.8 172.15
<none>                    1950.7 172.58
- on_market  1     128.5  2079.2 173.07
+ mntn       1       0.0  1950.7 174.58
- lot_size   1     214.3  2165.0 174.65
- dwntn      1     521.4  2472.1 179.82
- lp         1   21198.6 23149.3 267.06

Step:  AIC=170.59
sp ~ bed + sqft + dwntn + lot_size + garage + age + on_market +
    lp

            Df Sum of Sq     RSS    AIC
- age        1      33.0  1983.9 169.24
- garage     1      42.3  1993.2 169.42
- bed        1      72.2  2023.1 170.00
- sqft       1      85.5  2036.4 170.26
<none>                    1950.9 170.59
- on_market  1     131.4  2082.3 171.13
+ bath       1       0.2  1950.7 172.58
+ mntn       1       0.0  1950.9 172.59
- lot_size   1     230.9  2181.8 172.95
- dwntn      1     523.9  2474.8 177.86
- lp         1   23544.8 25495.7 268.82
```

```
Step:  AIC=169.24
sp ~ bed + sqft + dwntn + lot_size + garage + on_market + lp

            Df Sum of Sq     RSS    AIC
- garage     1      43.7  2027.6 168.09
- sqft       1      75.4  2059.3 168.69
- bed        1      78.3  2062.1 168.75
<none>                    1983.9 169.24
- on_market  1     141.5  2125.4 169.93
+ age        1      33.0  1950.9 170.59
- lot_size   1     213.1  2197.0 171.22
+ mntn       1       0.9  1982.9 171.22
+ bath       1       0.1  1983.8 171.24
- dwntn      1     491.5  2475.3 175.87
- lp         1   24550.5 26534.4 268.38

Step:  AIC=168.09
sp ~ bed + sqft + dwntn + lot_size + on_market + lp

            Df Sum of Sq   RSS    AIC
- bed        1        41  2068 166.87
- sqft       1        84  2111 167.67
<none>                    2028 168.09
- on_market  1       121  2148 168.34
+ garage     1        44  1984 169.24
- lot_size   1       181  2208 169.42
+ age        1        34  1993 169.42
+ bath       1         4  2023 170.01
+ mntn       1         0  2027 170.08
- dwntn      1       449  2477 173.90
- lp         1     33506 35534 277.77
```

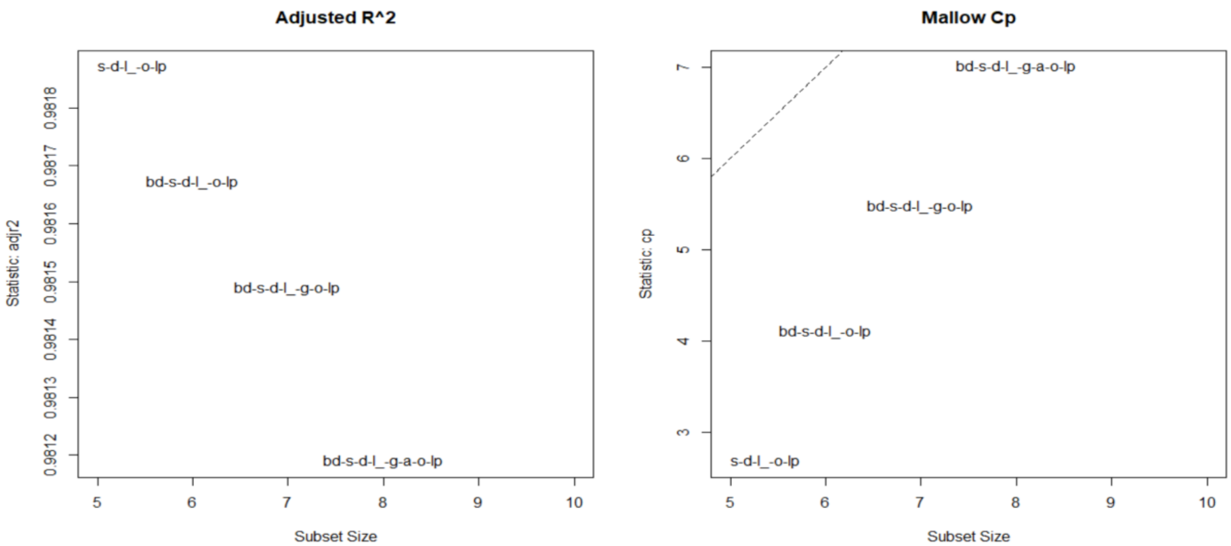
```
Step:  AIC=166.87
sp ~ sqft + dwntn + lot_size + on_market + lp

            Df Sum of Sq   RSS    AIC
<none>                    2068 166.87
- sqft       1       130  2198 167.24
- lot_size   1       163  2231 167.83
- on_market  1       177  2245 168.07
+ bed        1        41  2028 168.09
+ age        1        39  2030 168.13
+ bath       1        17  2051 168.55
+ garage     1         6  2062 168.75
+ mntn       1         1  2067 168.84
- dwntn      1       433  2501 172.28
- lp         1     39083 41151 281.50
> step$anova # display results
Stepwise Model Path
Analysis of Deviance Table

Initial Model:
sp ~ bed + bath + sqft + dwntn + mntn + lot_size + garage + age +
    on_market + lp

Final Model:
sp ~ sqft + dwntn + lot_size + on_market + lp


      Step Df     Deviance Resid. Df Resid. Dev      AIC
1                                   28   1950.729 174.5835
2   - mntn  1 6.304123e-05         29   1950.729 172.5835
3   - bath  1 1.700475e-01         30   1950.899 170.5869
4    - age  1 3.295124e+01         31   1983.851 169.2401
5 - garage  1 4.374427e+01         32   2027.595 168.0907
6    - bed  1 4.066041e+01         33   2068.255 166.8651
>
"""
```

```
#Case 2: (Using Ols_step best subset)

library(olsrr)
ols_step_best_subset(fit)

"""
> ols_step_best_subset(fit)
                       Best Subsets Regression
----------------------------------------------------------------------
Model Index    Predictors
----------------------------------------------------------------------
     1         lp
     2         dwntn lp
     3         dwntn on_market lp
     4         dwntn lot_size on_market lp
     5         sqft dwntn lot_size on_market lp
     6         bed sqft dwntn lot_size on_market lp
     7         bed sqft dwntn lot_size garage on_market lp
     8         bed sqft dwntn lot_size garage age on_market lp
     9         bed bath sqft dwntn lot_size garage age on_market lp
    10         bed bath sqft dwntn mntn lot_size garage age on_market lp
----------------------------------------------------------------------
```

```
                                                  Subsets Regression Summary
-------------------------------------------------------------------------------------------------------------------------------
                       Adj.        Pred
Model    R-Square    R-Square    R-Square     C(p)        AIC         SBIC         SBC          MSEP         FPE        HSP       APC
-------------------------------------------------------------------------------------------------------------------------------
  1        0.9800      0.9794      0.9777     2.7998     280.9645    170.4123    285.9552    2776.0221    74.8248    1.9771    0.0222
  2        0.9818      0.9808      0.9794     1.3363     279.2167    169.3175    285.8709    2593.7143    71.5608    1.8986    0.0212
  3        0.9828      0.9813       0.977     1.4044     278.9582    169.8264    287.2760    2519.7719    71.1179    1.8971    0.0211
  4        0.9833      0.9813      0.9752     2.5466     279.9118    171.4856    289.8932    2527.4003    72.9290    1.9588    0.0217
  5        0.9843      0.9819      0.9747     2.6869     279.5423    172.4328    291.1872    2452.7386    72.3166    1.9586    0.0215
  6        0.9846      0.9817      0.9732     4.1033     280.7679    174.6202    294.0764    2482.0847    74.7351    2.0439    0.0222
  7        0.9849      0.9815      0.9704     5.4754     281.9173    176.8805    296.8894    2509.4862    77.1224    2.1332    0.0229
  8        0.9851      0.9812      0.9705     7.0024     283.2641    179.3476    299.8997    2552.9010    80.0369    2.2424    0.0238
  9        0.9852      0.9805      0.9679     9.0000     285.2607    182.1319    303.5599    2643.8455    84.5144    2.4024    0.0251
 10        0.9852      0.9798      0.9624    11.0000     287.2607    184.9177    307.2234    2741.7657    89.3191    2.5803    0.0265
-------------------------------------------------------------------------------------------------------------------------------
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

>
"""
```

```
###Case 3 : (Using regsubsets)

library(leaps)
library(car)
attach(skiData)
regsubsets.out<-regsubsets(sp~bed+bath+sqft+dwntn+mntn+lot_size+garage+age+on_market+lp,data=skiData,nbest=1)
regsubsets.out
summary.out <- summary(regsubsets.out)
as.data.frame(summary.out$outmat)
which.max(summary.out$adjr2)
summary.out$which[5,]

"""
> attach(skiData)
> regsubsets.out<-regsubsets(sp~bed+bath+sqft+dwntn+mntn+lot_size+garage+age+on_market+lp,data=skiData,nbest=1)
> regsubsets.out
Subset selection object
Call: regsubsets.formula(sp ~ bed + bath + sqft + dwntn + mntn + lot_size +
    garage + age + on_market + lp, data = skiData, nbest = 1)
10 Variables  (and intercept)
          Forced in Forced out
bed           FALSE      FALSE
bath          FALSE      FALSE
sqft          FALSE      FALSE
dwntn         FALSE      FALSE
mntn          FALSE      FALSE
lot_size      FALSE      FALSE
garage        FALSE      FALSE
age           FALSE      FALSE
on_market     FALSE      FALSE
lp            FALSE      FALSE
1 subsets of each size up to 8
Selection Algorithm: exhaustive
> summary.out <- summary(regsubsets.out)
> as.data.frame(summary.out$outmat)
         bed bath sqft dwntn mntn lot_size garage age on_market lp
1  ( 1 )                                                        *
2  ( 1 )                *                                       *
3  ( 1 )                *                                 *     *
4  ( 1 )                *              *                  *     *
5  ( 1 )           *    *              *                  *     *
6  ( 1 )   *       *    *              *                  *     *
7  ( 1 )   *       *    *              *      *           *     *
8  ( 1 )   *       *    *              *      *     *     *     *

> which.max(summary.out$adjr2)
[1] 5
> summary.out$which[5,]
(Intercept)         bed        bath        sqft       dwntn        mntn
       TRUE       FALSE       FALSE        TRUE        TRUE       FALSE
   lot_size      garage         age   on_market          lp
       TRUE       FALSE       FALSE        TRUE        TRUE
>

"""
```

```
> ## Adjusted R2
> res.legend <-
+   subsets(regsubsets.out, statistic="adjr2", legend = FALSE, min.size = 5, main = "Adjusted R^2")
> ## Mallow Cp
> res.legend <-
+   subsets(regsubsets.out, statistic="cp", legend = FALSE, min.size = 5, main = "Mallow Cp")
> abline(a = 1, b = 1, lty = 2)
> res.legend
          Abbreviation
bed                 bd
bath                bt
sqft                 s
dwntn                d
mntn                 m
lot_size            l_
garage               g
age                  a
on_market            o
lp                  lp
> 
```

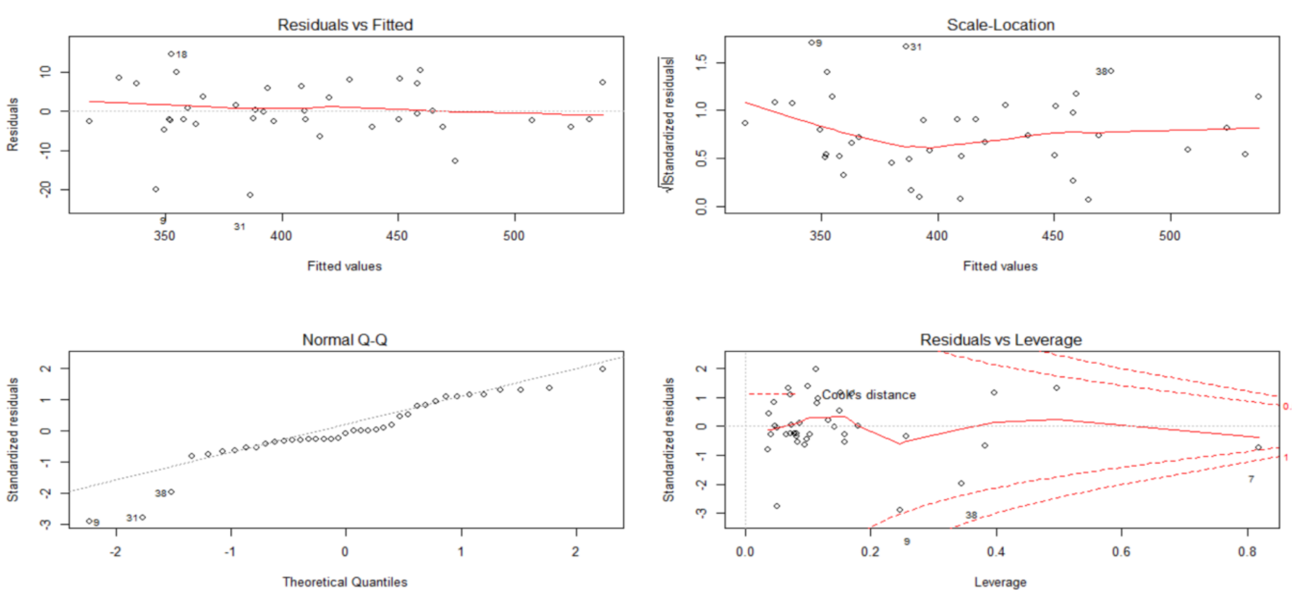



```
#Model Building

install.packages("boot")
install.packages("carData")
library(boot)
library(carData)
library(car)
set.seed(4)

# The best model from all above regression methods is
# with 5 variables
#sp ~ sqft + dwntn + lot_size + on_market + lp

model=lm( sp ~ sqft + dwntn + lot_size + on_market +lp,data=skiData)
summary(model)
anova(model)
vif(model)

""""
> model=lm( sp ~ sqft + dwntn + lot_size + on_market + lp,data=skiData)
> summary(model)

Call:
lm(formula = sp ~ sqft + dwntn + lot_size + on_market + lp, data = skiData)

Residuals:
     Min       1Q   Median       3Q      Max
-21.3916  -2.5665  -0.5458   6.1540  14.6149
```

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  9.959120  11.868016   0.839   0.4074
sqft         0.004975   0.003460   1.438   0.1599
dwntn       -0.519804   0.197749  -2.629   0.0129 *
lot_size     0.447494   0.277291   1.614   0.1161
on_market   -0.025186   0.014982  -1.681   0.1022
lp           0.943459   0.037781  24.972   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Residual standard error: 7.917 on 33 degrees of freedom
Multiple R-squared:  0.9843,   Adjusted R-squared:  0.9819
F-statistic: 412.6 on 5 and 33 DF,  p-value: < 2.2e-16

> anova(model)
Analysis of Variance Table

Response: sp
          Df Sum Sq Mean Sq  F value Pr(>F)
sqft       1  53331   53331 850.9186 <2e-16 ***
dwntn      1  18356   18356 292.8860 <2e-16 ***
lot_size   1  18522   18522 295.5254 <2e-16 ***
on_market  1     10      10   0.1546 0.6967
lp         1  39083   39083 623.5915 <2e-16 ***
Residuals 33   2068      63
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> vif(model)
     sqft     dwntn  lot_size on_market        lp
 3.205170  2.703068  2.450604  1.233597  2.965515
>
"""
```

 # Diagnostic plots
layout(matrix(c(1,2,3,4),2,2)) # optional 4 graphs/page
plot(model)



#####Model 2:

```
##Model 2 : All variables except listing price as predictors

#Case 1 : (Using AIC Stepwise regression both sides)

library(MASS)
fit <- lm(sp~bed+bath+sqft+dwntn+mntn+lot_size+garage+age+on_market,data=skiData)
step <- stepAIC(fit, direction="both")
step$anova # display results
|
""""
> fit <- lm(sp~bed+bath+sqft+dwntn+mntn+lot_size+garage+age+on_market,data=skiData)
> step <- stepAIC(fit, direction="both")
Start:  AIC=259
sp ~ bed + bath + sqft + dwntn + mntn + lot_size + garage + age +
    on_market

            Df Sum of Sq   RSS    AIC
- on_market  1      46.4 17930 257.10
- age        1     200.2 18084 257.43
- dwntn      1     294.6 18179 257.63
- bath       1     534.2 18418 258.14
<none>                   17884 259.00
- garage     1    2893.2 20777 262.84
- bed        1    3776.5 21660 264.47
- mntn       1    5265.3 23149 267.06
- sqft       1    7140.2 25024 270.10
- lot_size   1   16638.6 34523 282.65

Step:  AIC=257.1
sp ~ bed + bath + sqft + dwntn + mntn + lot_size + garage + age

            Df Sum of Sq   RSS    AIC
- age        1     198.2 18129 255.53
- dwntn      1     416.6 18347 255.99
- bath       1     660.8 18591 256.51
<none>                   17930 257.10
+ on_market  1      46.4 17884 259.00
- garage     1    3038.6 20969 261.20
- bed        1    3801.6 21732 262.60
- mntn       1    5295.1 23226 265.19
- sqft       1    7106.9 25037 268.12
- lot_size   1   20099.4 38030 284.42

Step:  AIC=255.53
sp ~ bed + bath + sqft + dwntn + mntn + lot_size + garage

            Df Sum of Sq   RSS    AIC
- dwntn      1     293.5 18422 254.15
- bath       1     624.0 18753 254.85
<none>                   18129 255.53
+ age        1     198.2 17930 257.10
+ on_market  1      44.4 18084 257.43
- garage     1    3163.0 21292 259.80
- bed        1    4186.0 22315 261.63
- mntn       1    6103.6 24232 264.84
- sqft       1    7110.0 25239 266.43
- lot_size   1   20050.2 38179 282.57
```

```
Step:  AIC=254.15
sp ~ bed + bath + sqft + mntn + lot_size + garage

            Df Sum of Sq   RSS    AIC
- bath       1     539.3 18961 253.28
<none>                   18422 254.15
+ dwntn      1     293.5 18129 255.53
+ on_market  1     144.1 18278 255.85
+ age        1      75.1 18347 255.99
- garage     1    2982.1 21404 258.00
- bed        1    4269.6 22692 260.28
- sqft       1    6932.9 25355 264.61
- lot_size   1   19791.9 38214 280.61
- mntn       1   31413.9 49836 290.96

Step:  AIC=253.28
sp ~ bed + sqft + mntn + lot_size + garage

            Df Sum of Sq   RSS    AIC
<none>                   18961 253.28
+ bath       1       539 18422 254.15
+ on_market  1       276 18686 254.71
+ dwntn      1       209 18753 254.85
+ age        1        67 18894 255.14
- garage     1      5287 24248 260.87
- bed        1      8788 27750 266.13
- sqft       1     10671 29632 268.69
- lot_size   1     19804 38765 279.17
- mntn       1     38755 57716 294.69
> step$anova # display results
Stepwise Model Path
Analysis of Deviance Table

Initial Model:
sp ~ bed + bath + sqft + dwntn + mntn + lot_size + garage + age +
    on_market


Final Model:
sp ~ bed + sqft + mntn + lot_size + garage


         Step Df  Deviance Resid. Df Resid. Dev      AIC
1                                29   17883.98 258.9959
2 - on_market  1  46.44029        30   17930.42 257.0970
3       - age  1 198.19433        31   18128.62 255.5257
4     - dwntn  1 293.50690        32   18422.13 254.1521
5      - bath  1 539.25150        33   18961.38 253.2773
>
```

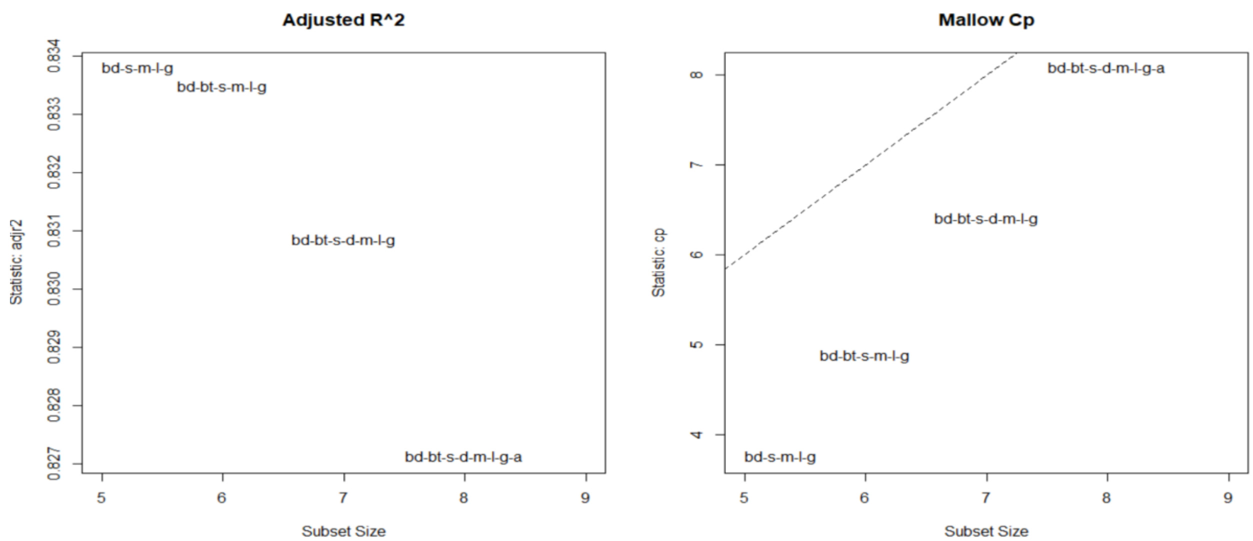
""""

```
#Case 2: (Using Ols_step best subset)

library(olsrr)
ols_step_best_subset(fit)

"""

> ols_step_best_subset(fit)
                       Best Subsets Regression
---------------------------------------------------------------------
Model Index    Predictors
---------------------------------------------------------------------
     1         bath
     2         bath mntn
     3         sqft mntn lot_size
     4         bath sqft mntn lot_size
     5         bed sqft mntn lot_size garage
     6         bed bath sqft mntn lot_size garage
     7         bed bath sqft dwntn mntn lot_size garage
     8         bed bath sqft dwntn mntn lot_size garage age
     9         bed bath sqft dwntn mntn lot_size garage age on_market
---------------------------------------------------------------------


                                                     Subsets Regression Summary
-----------------------------------------------------------------------------------------------------------------------------------------
                       Adj.        Pred
Model    R-Square    R-Square    R-Square     C(p)        AIC         SBIC         SBC          MSEP         FPE          HSP       APC
-----------------------------------------------------------------------------------------------------------------------------------------
  1        0.5516      0.5395      0.5178    60.5255    402.1649    288.4205    407.1556    62098.3918    1673.7989    44.2264    0.4969
  2        0.6068      0.5849      0.5518    50.7705    399.0437    284.5886    405.6979    56012.7070    1545.3956    41.0003    0.4588
  3        0.7776      0.7586      0.6311    16.3692    378.8095    266.6564    387.1273    32604.7325     920.2340    24.5480    0.2732
  4        0.8234      0.8026      0.6124     8.6196    371.8220    261.5091    381.8034    26678.6352     769.8207    20.6770    0.2285
  5        0.8557      0.8338      0.7066     3.7471    365.9545    258.3542    377.5994    22486.2479     662.9852    17.9558    0.1968
  6        0.8598      0.8335      0.6724     4.8726    366.8293    260.2430    380.1378    22551.4848     679.0206    18.5707    0.2016
  7        0.8620      0.8308      0.6543     6.3967    368.2029    262.5392    383.1750    22931.9273     704.7519    19.4931    0.2092
  8        0.8635      0.8271      0.6489     8.0753    369.7742    265.0082    386.4098    23463.3310     735.6071    20.6097    0.2184
  9        0.8639      0.8216      0.6267    10.0000    371.6731    267.6546    389.9723    24238.3660     774.8145    22.0246    0.2300
-----------------------------------------------------------------------------------------------------------------------------------------
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

>

"""
```

```
###Case 3 : (Using regsubsets)

library(leaps)
attach(skiData)
library(car)
regsubsets.out<-regsubsets(sp~bed+bath+sqft+dwntn+mntn+lot_size+garage+age+on_market,data=skiData,nbest=1)
regsubsets.out
summary.out <- summary(regsubsets.out)
as.data.frame(summary.out$outmat)
which.max(summary.out$adjr2)
summary.out$which[5,]
```

```
"""
> library(leaps)
> attach(skiData)
> library(car)
> regsubsets.out<-regsubsets(sp~bed+bath+sqft+dwntn+mntn+lot_size+garage+age+on_market,data=skiData,nbest=1)
> regsubsets.out
Subset selection object
Call: regsubsets.formula(sp ~ bed + bath + sqft + dwntn + mntn + lot_size +
    garage + age + on_market, data = skiData, nbest = 1)
9 Variables  (and intercept)
          Forced in Forced out
bed           FALSE      FALSE
bath          FALSE      FALSE
sqft          FALSE      FALSE
dwntn         FALSE      FALSE
mntn          FALSE      FALSE
lot_size      FALSE      FALSE
garage        FALSE      FALSE
age           FALSE      FALSE
on_market     FALSE      FALSE
1 subsets of each size up to 8
Selection Algorithm: exhaustive
> summary.out <- summary(regsubsets.out)
> as.data.frame(summary.out$outmat)
         bed bath sqft dwntn mntn lot_size garage age on_market
1  ( 1 )          *
2  ( 1 )          *              *
3  ( 1 )               *         *         *
4  ( 1 )          *    *         *         *
5  ( 1 )     *         *         *         *      *
6  ( 1 )     *    *    *         *         *      *
7  ( 1 )     *    *    *     *   *         *      *
8  ( 1 )     *    *    *     *   *         *      *   *
> which.max(summary.out$adjr2)
[1] 5
> summary.out$which[5,]
(Intercept)         bed        bath        sqft       dwntn        mntn    lot_size      garage         age
       TRUE        TRUE       FALSE        TRUE       FALSE        TRUE        TRUE        TRUE       FALSE
  on_market
      FALSE
>
"""
```

```
> layout(matrix(1:2, ncol = 2))
> ## Adjusted R2
> res.legend <-
+   subsets(regsubsets.out, statistic="adjr2", legend = FALSE, min.size = 5, main = "Adjusted R^2")
> ## Mallow Cp
> res.legend <-
+   subsets(regsubsets.out, statistic="cp", legend = FALSE, min.size = 5, main = "Mallow Cp")
> abline(a = 1, b = 1, lty = 2)
> res.legend
          Abbreviation
bed                 bd
bath                bt
sqft                 s
dwntn                d
mntn                 m
lot_size             l
garage               g
age                  a
on_market            o
>
```




```
#Model Building

install.packages("boot")
install.packages("carData")
library(boot)
library(carData)
library(car)
set.seed(4)

# The best model from all above regression methods is
# with 5 variables
# sp ~ sqft + dwntn + lot_size + on_market + lp

model=lm( sp ~ bed + sqft + mntn + lot_size + garage,data=skiData)
summary(model)
anova(model)
vif(model)
```

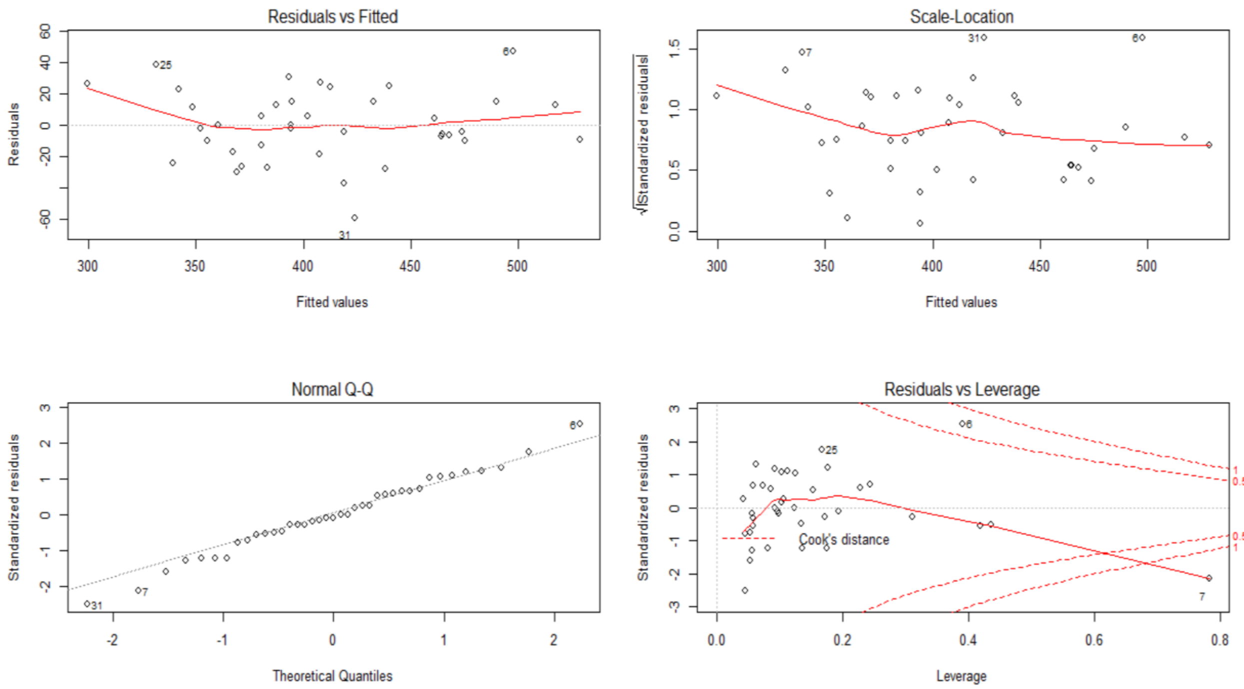
""""

```
> model=lm( sp ~ bed + sqft + mntn + lot_size + garage,data=skiData)
> summary(model)

Call:
lm(formula = sp ~ bed + sqft + mntn + lot_size + garage, data = skiData)

Residuals:
    Min      1Q  Median      3Q     Max 
-59.104 -11.554  -2.188  15.104  47.318 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept) 263.063011  16.343425  16.096  < 2e-16 ***
bed          21.836826   5.583632   3.911 0.000433 ***
sqft          0.040099   0.009305   4.309 0.000139 ***
mntn         -4.302038   0.523829  -8.213 1.75e-09 ***
lot_size      4.084563   0.695742   5.871 1.41e-06 ***
garage       13.657168   4.502471   3.033 0.004688 ** 
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
Residual standard error: 23.97 on 33 degrees of freedom
Multiple R-squared:  0.8557,	Adjusted R-squared:  0.8338 
F-statistic: 39.13 on 5 and 33 DF,  p-value: 6.087e-13
```

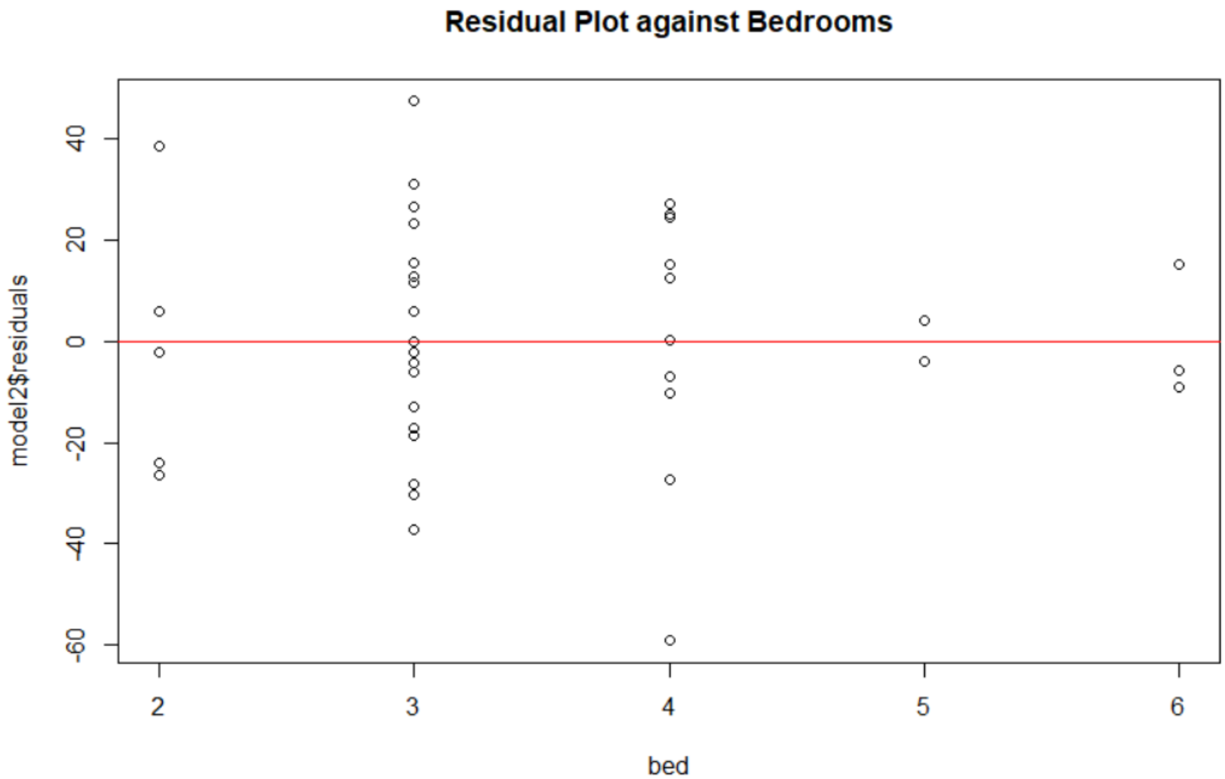
```
> anova(model)
Analysis of Variance Table

Response: sp
          Df Sum Sq Mean Sq F value    Pr(>F)    
bed        1  49040   49040 85.3483 1.132e-10 ***
sqft       1  12113   12113 21.0814 6.116e-05 ***
mntn       1  20765   20765 36.1385 9.325e-07 ***
lot_size   1  25204   25204 43.8654 1.560e-07 ***
garage     1   5287    5287  9.2007  0.004688 ** 
Residuals 33  18961     575                      
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> vif(model)
     bed     sqft     mntn lot_size   garage 
2.264996 2.528167 1.595349 1.682800 1.326219 
>
```

""""

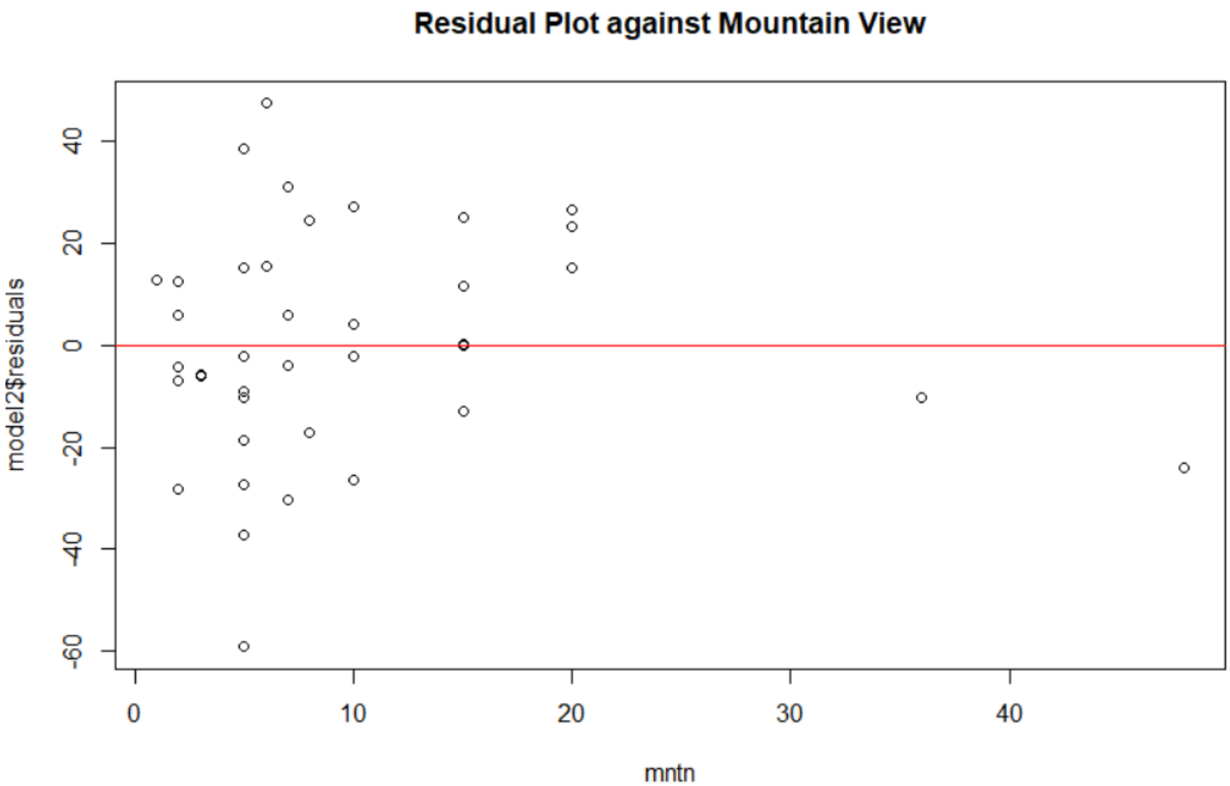
```
# Diagnostic plots
layout(matrix(c(1,2,3,4),2,2)) # optional 4 graphs/page
plot(model)
```

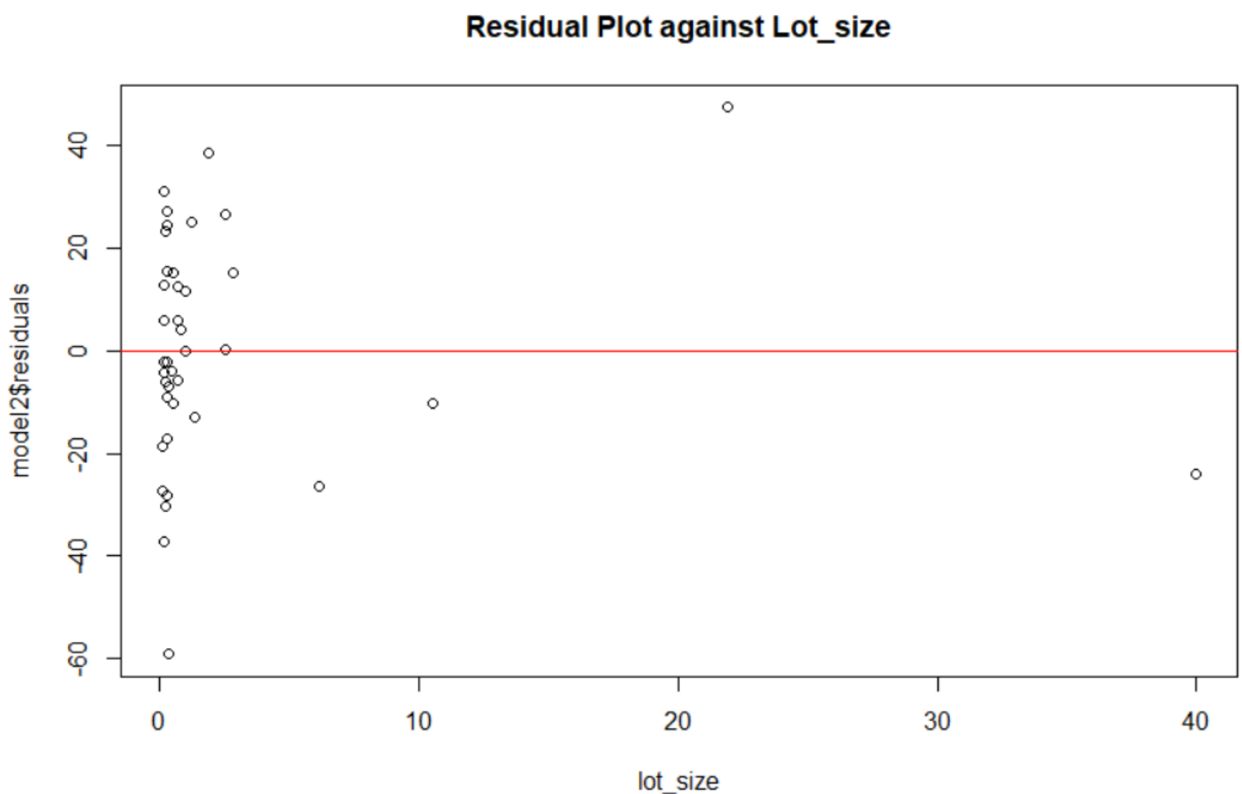


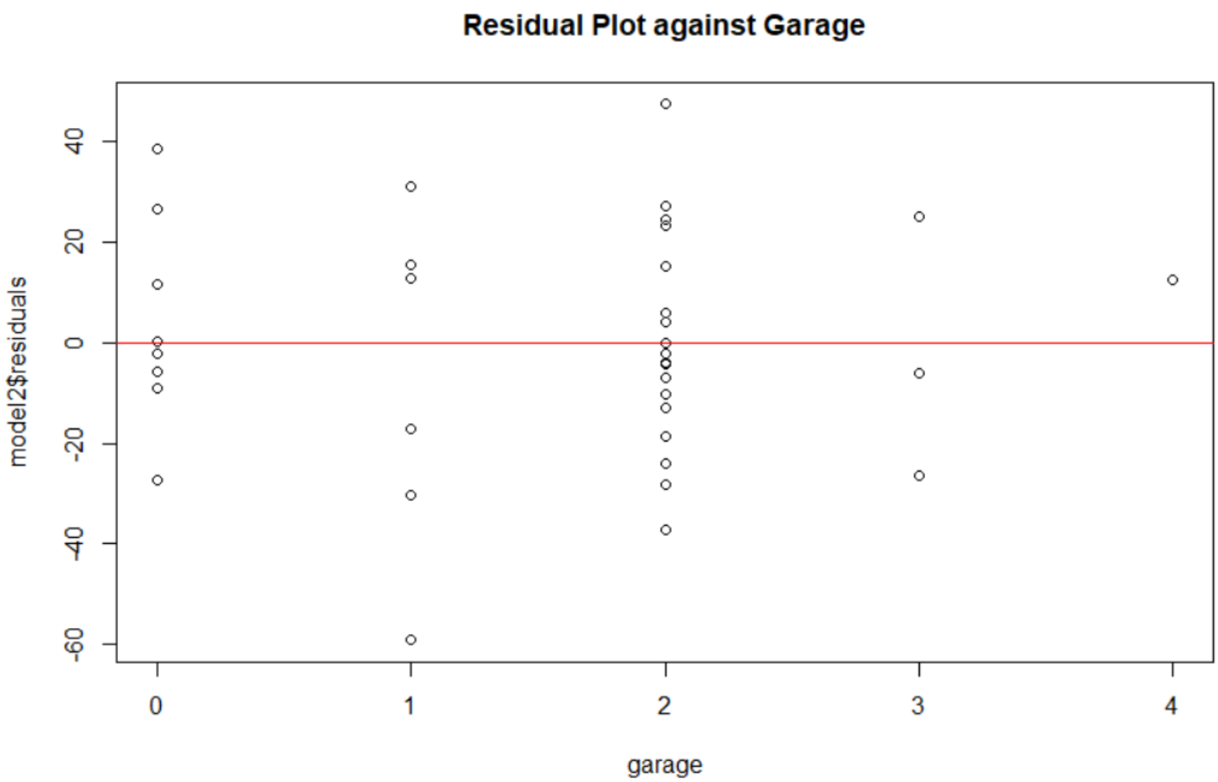

```
#Residual Plots against independent variables

plot(bed,model2$residuals , main = 'Residual Plot against Bedrooms')
abline(h=0, col = 'red')

plot(sqft,model2$residuals, main = 'Residual Plot against Sq_ft')
abline(h=0, col = 'red')

plot(mntn,model2$residuals, main = 'Residual Plot against Mountain View')
abline(h=0, col = 'red')

plot(lot_size,model2$residuals, main = 'Residual Plot against Lot_size')
abline(h=0, col = 'red')

plot(garage,model2$residuals, main = 'Residual Plot against Garage')
abline(h=0, col = 'red')
```


Residual Plot against Bedrooms
model2$residuals
bed
-60
-40
-20
0
20
40
2
3
4
5
6


Residual Plot against Sq_ft
model2$residuals
sqft
-60
-40
-20
0
20
40
1000
1500
2000
2500
3000
3500


Residual Plot against Mountain View
model2$residuals
mntn
-60
-40
-20
0
20
40
0
10
20
30
40


Residual Plot against Lot_size
model2$residuals
lot_size
-60
-40
-20
0
20
40
0
10
20
30
40


Residual Plot against Garage
model2$residuals
40
20
0
-20
-40
-60
0
1
2
3
4
garage

```
###############  Validation of Model by Data Splitting  ###########

#split the data into training and testing

library(caret)
install.packages("boot")
install.packages("carData")
library(boot)
library(carData)
library(car)
set.seed(4)
n=nrow(skiData)
shuffled=skiData[sample(n),]
train=shuffled[1:round(0.85 * n),]
test = shuffled[(round(0.85 * n) + 1):n,]

#Model 1 - Validation with Traning Data

Vm1=lm(`Selling price`~ Sq_Ft+Downtwon+`Lot size` + `On market` + `List price`,data=train)
summary(Vm1)
vif(Vm1)

#Prediction
prediction=predict.lm(Vm1,newdata=test)
prediction

#Compute metrics R2, RMSE, MAE

R2(prediction, test$'Selling price')
RMSE(prediction, test$'Selling price')
MAE(prediction, test$'Selling price')

""""
> summary(Vm1)

Call:
lm(formula = `Selling price` ~ Sq_Ft + Downtwon + `Lot size` +
    `On market` + `List price`, data = train)

Residuals:
     Min       1Q   Median       3Q      Max
-21.4104  -3.0630   0.2741   3.6272  12.2826

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)   9.021384  11.606212   0.777   0.4437
Sq_Ft         0.002036   0.003323   0.613   0.5451
Downtwon     -0.359600   0.193405  -1.859   0.0739 .
`Lot size`    0.222097   0.271473   0.818   0.4205
`On market`  -0.013364   0.015509  -0.862   0.3964
`List price`  0.955545   0.036335  26.298   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 7.104 on 27 degrees of freedom
Multiple R-squared:  0.9877,	Adjusted R-squared:  0.9854
F-statistic: 432.8 on 5 and 27 DF,  p-value: < 2.2e-16

> vif(Vm1)
       Sq_Ft     Downtwon   `Lot size`  `On market` `List price`
    3.157480     3.087699     2.874854     1.401524     2.952450
>
```

```
#######################################################

> prediction=predict.lm(Vm1,newdata=test)

> prediction
       1        2        3        4        5        6 
416.0652 351.3674 353.6140 461.1635 390.3674 332.5647 

> test$`Selling price`
[1] 410.0 326.0 350.0 470.0 386.3 339.0


> R2(prediction, test$'Selling price')
[1] 0.9562514

> RMSE(prediction, test$'Selling price')
[1] 11.7572


> MAE(prediction, test$'Selling price')
[1] 9.064281

""
```

```
#########################################################

#Model2 - Validation

Vm2=lm(`Selling price`~Bedrooms+Sq_Ft+Mountain+Garage+`Lot size`,data=train)
summary(Vm2)
vif(Vm2)


#predict and compute r2,RMSE
prediction=predict.lm(Vm2,newdata=test)
prediction
test$`Selling price`

R2(prediction, test$'Selling price')
RMSE(prediction, test$'Selling price')
MAE(prediction, test$'Selling price')

""
> summary(Vm2)

Call:
lm(formula = `Selling price` ~ Bedrooms + Sq_Ft + Mountain + 
    Garage + `Lot size`, data = train)

Residuals:
    Min      1Q  Median      3Q     Max 
-60.647 -11.706  -1.705  13.391  42.812 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept) 266.257830  18.547033  14.356 3.71e-14 ***
Bedrooms     20.648675   5.981347   3.452 0.001847 ** 
Sq_Ft         0.042030   0.009896   4.247 0.000229 ***
Mountain     -4.555631   0.563856  -8.079 1.11e-08 ***
Garage       14.139980   4.689123   3.015 0.005531 ** 
`Lot size`    4.160842   0.723339   5.752 4.07e-06 ***
```

```
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 24.32 on 27 degrees of freedom
Multiple R-squared:  0.8556,   Adjusted R-squared:  0.8289
F-statistic:    32 on 5 and 27 DF,  p-value: 1.547e-10

> vif(Vm2)
  Bedrooms      Sq_Ft   Mountain     Garage `Lot size`
  2.189031   2.389787   1.664013   1.259722   1.741934
>


###################################################################

> prediction
       1        2        3        4        5        6
442.5593 296.3310 368.0926 475.3602 384.1502 370.2530


> test$`Selling price`
[1] 410.0 326.0 350.0 470.0 386.3 339.0

> library(caret)
> RMSE(prediction, test$'Selling price')
[1] 23.37307

> MAE(prediction, test$'Selling price')
[1] 19.8473

> R2(prediction, test$'Selling price')
[1] 0.8649214

"""
```

<table>
<tr><th colspan="3">Validation by Data Splitting</th></tr>
<tr><th></th><th>Model 1</th><th>Model 2</th></tr>
<tr><td>RMSE</td><td>11.7572</td><td>23.37307</td></tr>
<tr><td>MAE</td><td>9.064281</td><td>19.8473</td></tr>
<tr><td>R2</td><td>0.9562514</td><td>0.8649</td></tr>
<tr><td>Vif</td><td>> vif(Vm1)
Sq_Ft Downtwon `Lot size` `On market` `List price`
3.157480 3.087699 2.874854 1.401524 2.952450</td><td>> vif(Vm2)
Bedrooms Sq_Ft Mountain Garage `Lot size`
2.189031 2.389787 1.664013 1.259722 1.741934</td></tr>
<tr><td>Coefficients by Significance</td><td>Coefficients:
Estimate Std. Error t value Pr(>|t|)
(Intercept) 9.021384 11.606212 0.777 0.4437
Sq_Ft 0.002036 0.003323 0.613 0.5451
Downtwon -0.359600 0.193405 -1.859 0.0739 .
`Lot size` 0.222097 0.271473 0.818 0.4205
`On market` -0.013364 0.015509 -0.862 0.3964
`List price` 0.955545 0.036335 26.298 <2e-16 ***

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1</td><td>Coefficients:
Estimate Std. Error t value Pr(>|t|)
(Intercept) 266.257830 18.547033 14.356 3.71e-14 ***
Bedrooms 20.648675 5.981347 3.452 0.001847 **
Sq_Ft 0.042030 0.009896 4.247 0.000229 ***
Mountain -4.555631 0.563856 -8.079 1.11e-08 ***
Garage 14.139980 4.689123 3.015 0.005531 **
`Lot size` 4.160842 0.723339 5.752 4.07e-06 ***

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1</td></tr>
</table>

```
######### Case 2 : Validation of Model by K FOLD CROSS VALIDATION  ###########

#Model 1

Model_CV1 <- train(`Selling price`~ Sq_Ft+Downtwon+`Lot size` + `On market` + `List price`,
            training,method="lm",trControl=trainControl(method="cv",number=10,verboseIter=TRUE))
Model_CV1

""
> Model_CV1 <- train(`Selling price`~ Sq_Ft+Downtwon+`Lot size` + `On market` + `List price`,
training,method="lm",trControl=trainControl(method="cv",number=10,verboseIter=TRUE))
+ Fold01: intercept=TRUE
- Fold01: intercept=TRUE
+ Fold02: intercept=TRUE
- Fold02: intercept=TRUE
+ Fold03: intercept=TRUE
- Fold03: intercept=TRUE
+ Fold04: intercept=TRUE
- Fold04: intercept=TRUE
+ Fold05: intercept=TRUE
- Fold05: intercept=TRUE
+ Fold06: intercept=TRUE
- Fold06: intercept=TRUE
+ Fold07: intercept=TRUE
- Fold07: intercept=TRUE
+ Fold08: intercept=TRUE
- Fold08: intercept=TRUE
+ Fold09: intercept=TRUE
- Fold09: intercept=TRUE
+ Fold10: intercept=TRUE
- Fold10: intercept=TRUE
Aggregating results
Fitting final model on full training set


> Model_CV1
Linear Regression

33 samples
 5 predictor

No pre-processing
Resampling: Cross-Validated (10 fold)
Summary of sample sizes: 30, 29, 30, 29, 30, 30, ...
Resampling results:

  RMSE      Rsquared   MAE
  7.642107  0.9807507  6.077977

Tuning parameter 'intercept' was held constant at a value of TRUE
>

""
```

```
#######  Model 2 : Kfold cross validation  ###############################################

Model_CV2 <- train(`Selling price`~Bedrooms+Sq_Ft+Mountain+Garage+`Lot size`,training,method="lm",
                   trControl=trainControl(method="cv",number=10,verboseIter=TRUE))
Model_CV2

"""
> Model_CV2 <- train(`Selling price`~Bedrooms+Sq_Ft+Mountain+Garage+`Lot size`,training,method="lm",
trControl=trainControl(method="cv",number=10,verboseIter=TRUE))
+ Fold01: intercept=TRUE
- Fold01: intercept=TRUE
+ Fold02: intercept=TRUE
- Fold02: intercept=TRUE
+ Fold03: intercept=TRUE
- Fold03: intercept=TRUE
+ Fold04: intercept=TRUE
- Fold04: intercept=TRUE
+ Fold05: intercept=TRUE
- Fold05: intercept=TRUE
+ Fold06: intercept=TRUE
- Fold06: intercept=TRUE
+ Fold07: intercept=TRUE
- Fold07: intercept=TRUE
+ Fold08: intercept=TRUE
- Fold08: intercept=TRUE
+ Fold09: intercept=TRUE
- Fold09: intercept=TRUE
+ Fold10: intercept=TRUE
- Fold10: intercept=TRUE
Aggregating results
Fitting final model on full training set

> Model_CV2
Linear Regression

33 samples
 5 predictor

No pre-processing
Resampling: Cross-Validated (10 fold)
Summary of sample sizes: 30, 30, 30, 30, 30, 29, ...
Resampling results:

  RMSE      Rsquared   MAE
  28.57572  0.7697075  23.04362

Tuning parameter 'intercept' was held constant at a value of TRUE
>
"""
```

Validation by Kfold Cross Validation		
	Model 1	**Model 2**
RMSE	7.642	28.575
MAE	6.078	23.043
R2	0.98	0.769