University of Padua

Department of Mathematics

Data Science master's program

Predict the prices of Miami Houses

Statistical learning project (Mod. B)

July 2022

Sina Ainesazi Dovom, Bahar Khanshaghaghi

2050575 2054868

Contents

1. Introduction 3
1.1 Objectives of the study 3
2. Preparation of the Dataset 3
2.1. Collection of the Dataset 3
2.2.Preprocessing Data 5
2.3.Data Exploration and Data Analysis 7
3. Modelling and Data analysis 19
3.1. Numerical Variables 19
3.2. Categorical Variables 72
Final Model 78
4.1. Final Model selection 78
4.2. Cross-Validation of the Final model 83

1. Introduction

1.1 Objectives of the study

The data source of this study was found on Kaggle.com. This dataset is titled "Miami Housing Dataset". The dataset contains information on 13,932 single-family homes sold in Miami of USA. The goal is to obtain a regression model using 17 variables inside the dataset to predict the prices of the houses in Miami. A simple linear regression model proceeded with only the most important variable and then by checking on all the other explanatory variables, the most useful ones have been selected and the degree of the polynomial is checked to understand which degree is better for each of them using the response variable which is the price of the houses.

2. Preparation of the Dataset

Importing the required Libraries

```
library(ggplot2)
library(leaps)
library(boot)
library(Metrics)
library(knitr)
```

2.1. Collection of the Dataset

The dataset has 17 features giving information about 13,932 single-family homes sold in Miami, USA.

```
# Importing the Dataset #
house = 'miami-housing.csv'
Miami_house <- read.csv(house)
head(Miami_house)
```

```
##   LATITUDE LONGITUDE     PARCELNO SALE_PRC LND_SQFOOT TOT_LVG_AREA
## 1 25.89103 -80.16056 622280070620   440000       9375         1753
## 2 25.89132 -80.15397 622280100460   349000       9375         1715
## 3 25.89133 -80.15374 622280100470   800000       9375         2276
## 4 25.89176 -80.15266 622280100530   988000      12450         2058
## 5 25.89182 -80.15464 622280100200   755000      12800         1684
## 6 25.89206 -80.16135 622280070180   630000       9900         1531
##   SPEC_FEAT_VAL RAIL_DIST OCEAN_DIST WATER_DIST CNTR_DIST SUBCNTR_DI HWY_D
IST
## 1             0    2815.9    12811.4      347.6   42815.3    37742.2  1595
4.9
## 2             0    4359.1    10648.4      337.8   43504.9    37340.5  1812
5.0
## 3         49206    4412.9    10574.1      297.1   43530.4    37328.7  1820
0.5
```

```
## 4         10033    4585.0    10156.5        0.0   43797.5    37423.2  1851
4.4
## 5         16681    4063.4    10836.8      326.6   43599.7    37550.8  1790
3.4
## 6          2978    2391.4    13017.0      188.9   43135.1    38176.2  1568
7.2
##   age avno60plus month_sold structure_quality
## 1  67          0          8                 4
## 2  63          0          9                 4
## 3  61          0          2                 4
## 4  63          0          9                 4
## 5  42          0          7                 4
## 6  41          0          2                 4

attach(Miami_house)
```

First look at the Dataset

```
str(Miami_house)

## 'data.frame':    13932 obs. of  17 variables:
##  $ LATITUDE         : num  25.9 25.9 25.9 25.9 25.9 ...
##  $ LONGITUDE        : num  -80.2 -80.2 -80.2 -80.2 -80.2 ...
##  $ PARCELNO         : num  6.22e+11 6.22e+11 6.22e+11 6.22e+11 6.22e+11 ..
.
##  $ SALE_PRC         : num  440000 349000 800000 988000 755000 630000 10200
00 850000 250000 1220000 ...
##  $ LND_SQFOOT       : int  9375 9375 9375 12450 12800 9900 10387 10272 937
5 13803 ...
##  $ TOT_LVG_AREA     : int  1753 1715 2276 2058 1684 1531 1753 1663 1493 30
77 ...
##  $ SPEC_FEAT_VAL    : int  0 0 49206 10033 16681 2978 23116 34933 11668 34
580 ...
##  $ RAIL_DIST        : num  2816 4359 4413 4585 4063 ...
##  $ OCEAN_DIST       : num  12811 10648 10574 10156 10837 ...
##  $ WATER_DIST       : num  348 338 297 0 327 ...
##  $ CNTR_DIST        : num  42815 43505 43530 43798 43600 ...
##  $ SUBCNTR_DI       : num  37742 37340 37329 37423 37551 ...
##  $ HWY_DIST         : num  15955 18125 18200 18514 17903 ...
##  $ age              : int  67 63 61 63 42 41 63 21 56 63 ...
##  $ avno60plus       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ month_sold       : int  8 9 2 9 7 2 2 9 3 11 ...
##  $ structure_quality: int  4 4 4 4 4 4 5 4 4 5 ...
```

Variables in the Dataset and details related to them are as follows:

- PARCELNO: unique identifier for each property. About 1% appear multiple times.
- SALE_PRC: sale price ($)
- LND_SQFOOT: land area (square feet)

- TOT_LVG_AREA: floor area (square feet)
- SPEC_FEAT_VAL: the value of special features (e.g., swimming pools) ($)
- RAIL_DIST: distance to the nearest rail line (an indicator of noise) (feet)
- OCEAN_DIST: distance to the ocean (feet)
- WATER_DIST: distance to the nearest body of water (feet)
- CNTR_DIST: distance to the Miami central business district (feet)
- SUBCNTR_DI: distance to the nearest subcenter (feet)
- HWY_DIST: distance to the nearest highway (an indicator of noise) (feet)
- age: age of the structure
- avno60plus: dummy variable for airplane noise exceeding an acceptable level
- structure_quality: quality of the structure
- month_sold: sale month in 2016 (1 = Jan)
- LATITUDE
- LONGITUDE

2.2.Preprocessing Data

Checking the dimension of the dataset.

```
dim(Miami_house)
```

```
## [1] 13932    17
```

Dropping the ID columns since it has no specific information

```
Miami_house$PARCELNO = NULL
```

Checking to see if there are null values in the dataset.

```
sum(is.na(Miami_house))
```

```
## [1] 0
```

So, there are no null values in the dataset.

Take a look at the Target Variable

```
summary(Miami_house$SALE_PRC)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   72000  235000  310000  399942  428000 2650000
```

The distribution of the Target variable in this study is not normal and it will be displayed using a histogram in the data exploration part. When the original continuous data do not follow the bell curve, it can be transformed to its logarithmic form to be as "normal" as possible so that the statistical analysis results from this data become more valid. In other words, the log transformation reduces or removes the skewness of the original data.

```
Miami_house$log10_SALE_PRC = log10(SALE_PRC)
```

The Categorical variables which have been converted to factors in this study are:

1. avno60plus which is a dummy variable for airplane noise exceeding an acceptable level.
2. month_sold which shows the sale month of the structure in 2016 (1 = Jan).
3. structure_quality which has a value between 1 to 5 and shows the quality of the structure.

```
Miami_house$avno60plus = as.factor(Miami_house$avno60plus)
Miami_house$month_sold = as.factor(Miami_house$month_sold)
Miami_house$structure_quality = as.factor(Miami_house$structure_quality)
```

Two new columns were added to the Dataset, indicating if the house has a specific feature like a swimming pool and if the house has a body of water(there is a river or a lake inside the property or the property is very close to a river or a lake.)

```
Miami_house$has_SPECFEAT = as.factor(Miami_house$SPEC_FEAT_VAL!=0)
Miami_house$has_BODYOFWATER = as.factor(Miami_house$WATER_DIST==0)
```

Taking a look at the dimension and the new columns added to the dataset, after taking the Preprocessing actions.

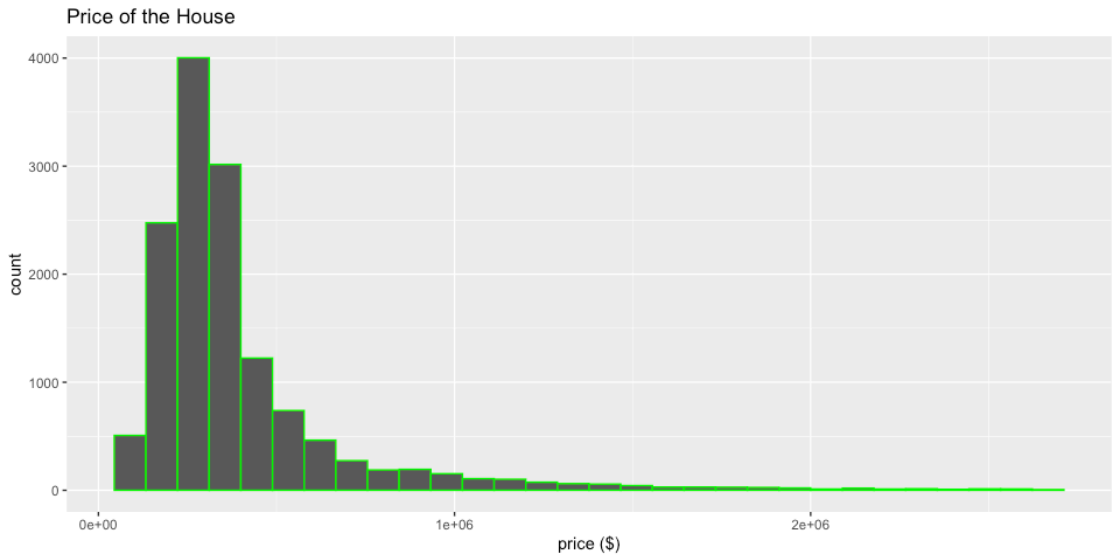
```
dim(Miami_house)

## [1] 13932    19

str(Miami_house)

## 'data.frame':    13932 obs. of  19 variables:
##  $ LATITUDE         : num  25.9 25.9 25.9 25.9 25.9 ...
##  $ LONGITUDE        : num  -80.2 -80.2 -80.2 -80.2 -80.2 ...
##  $ SALE_PRC         : num  440000 349000 800000 988000 755000 630000 10200
00 850000 250000 1220000 ...
##  $ LND_SQFOOT       : int  9375 9375 9375 12450 12800 9900 10387 10272 937
5 13803 ...
##  $ TOT_LVG_AREA     : int  1753 1715 2276 2058 1684 1531 1753 1663 1493 30
77 ...
##  $ SPEC_FEAT_VAL    : int  0 0 49206 10033 16681 2978 23116 34933 11668 34
580 ...
##  $ RAIL_DIST        : num  2816 4359 4413 4585 4063 ...
##  $ OCEAN_DIST       : num  12811 10648 10574 10156 10837 ...
##  $ WATER_DIST       : num  348 338 297 0 327 ...
```

```
##  $ CNTR_DIST         : num  42815 43505 43530 43798 43600 ...
##  $ SUBCNTR_DI        : num  37742 37340 37329 37423 37551 ...
##  $ HWY_DIST          : num  15955 18125 18200 18514 17903 ...
##  $ age               : int  67 63 61 63 42 41 63 21 56 63 ...
##  $ avno60plus        : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 ...
##  $ month_sold        : Factor w/ 12 levels "1","2","3","4",..: 8 9 2 9 7 2
2 9 3 11 ...
##  $ structure_quality: Factor w/ 5 levels "1","2","3","4",..: 4 4 4 4 4 4 5
4 4 5 ...
##  $ log10_SALE_PRC    : num  5.64 5.54 5.9 5.99 5.88 ...
##  $ has_SPECFEAT      : Factor w/ 2 levels "FALSE","TRUE": 1 1 2 2 2 2 2 2 2
2 ...
##  $ has_BODYOFWATER   : Factor w/ 2 levels "FALSE","TRUE": 1 1 1 2 1 1 2 1 1
1 ...
```

2.3.Data Exploration and Data Analysis

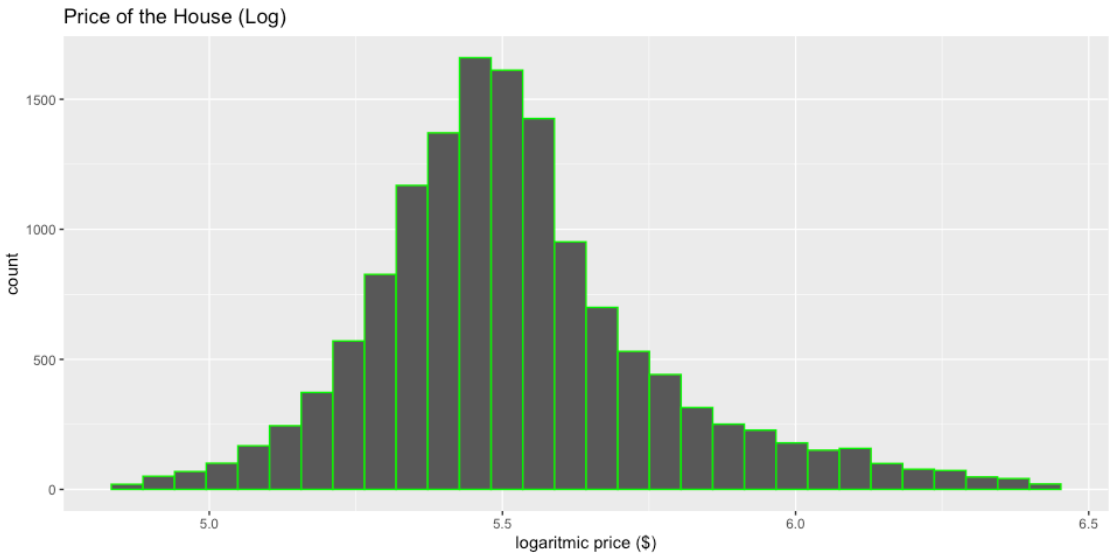
Checking on the distribution of the variables using plots.

```
# Checking the distribution of SALE PRICE variable #
ggplot(Miami_house, aes(x= SALE_PRC)) + geom_histogram(color = "green", bins=
30) +
  labs(x ="price ($)", title = "Price of the House")
```



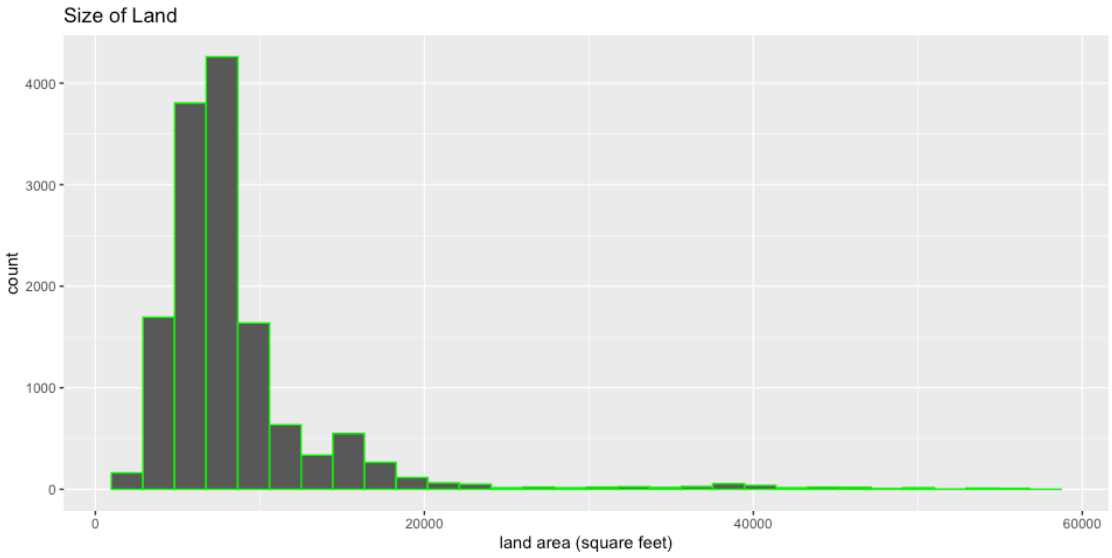
This histogram shows that the majority of the instances inside this dataset have prices less than 1 million dollars and the response variable in this study does not have a normal distribution.

```
# Logaritmic distribution of SALE PRICE variable #
ggplot(Miami_house, aes(x= log10_SALE_PRC)) + geom_histogram(color = "green",
bins=30) +
  labs(x ="logaritmic price ($)", title = "Price of the House (Log)")
```



This histogram shows the distribution of the new response variable which is the logarithmic type of the previous response variable. It perfectly shows the normal distribution of the values.

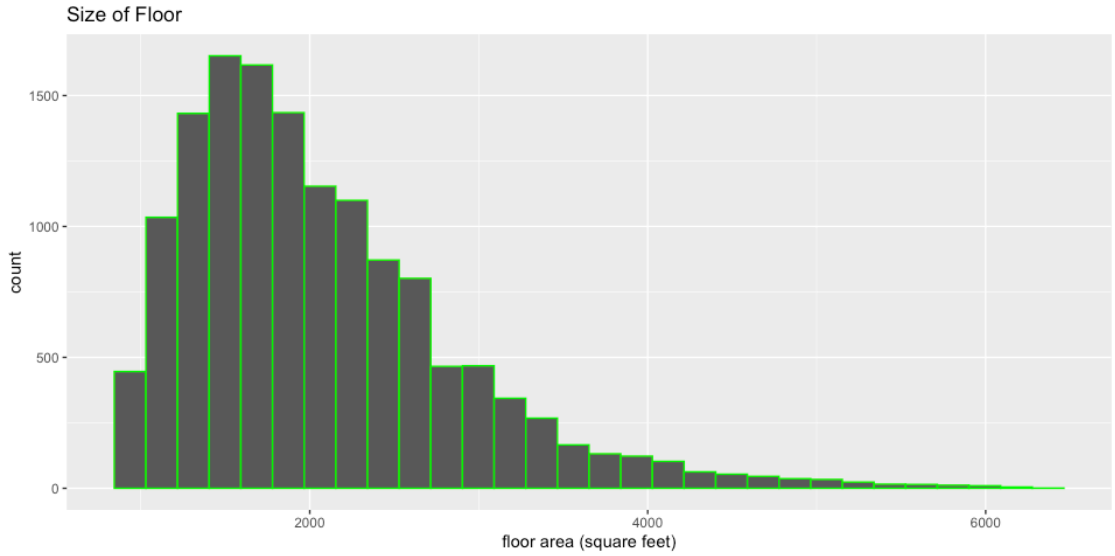
```
# Checking distribution of LAND SIZE variable #
ggplot(Miami_house, aes(x = LND_SQFOOT)) + geom_histogram(color = "green", bi
ns=30) +
  labs(x ="land area (square feet)", title = "Size of Land")
```



This histogram is for the variable related to the size of the land in which the house is located and it's obvious that the majority of the instances have a land area of less the 20 thousand square feet.

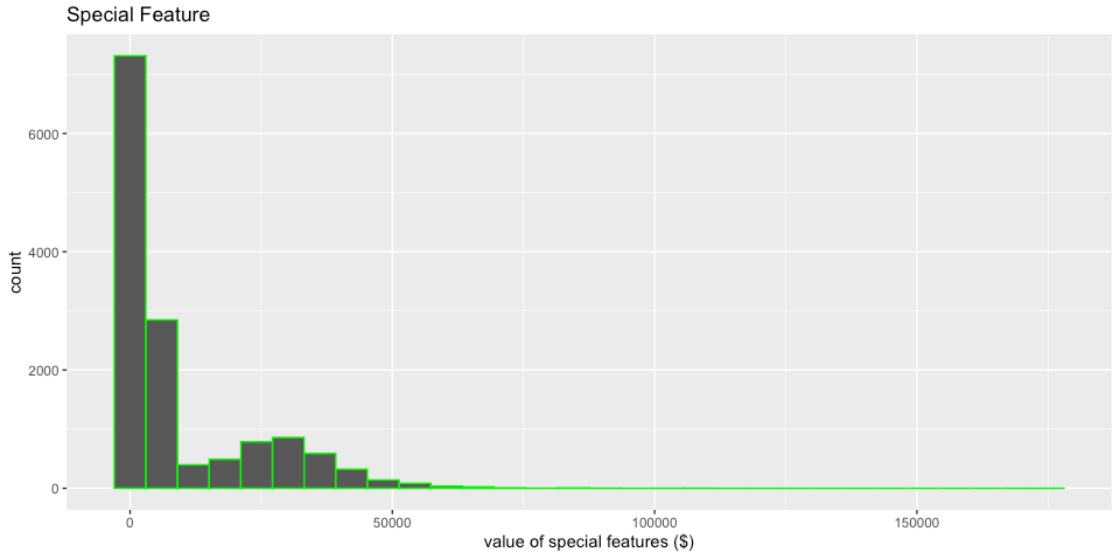
```
# Checking distribution of FLOOR SIZE variable #
ggplot(Miami_house, aes(x = TOT_LVG_AREA)) + geom_histogram(color = "green",b
```

```
ins=30) +
  labs(x ="floor area (square feet)", title = "Size of Floor")
```



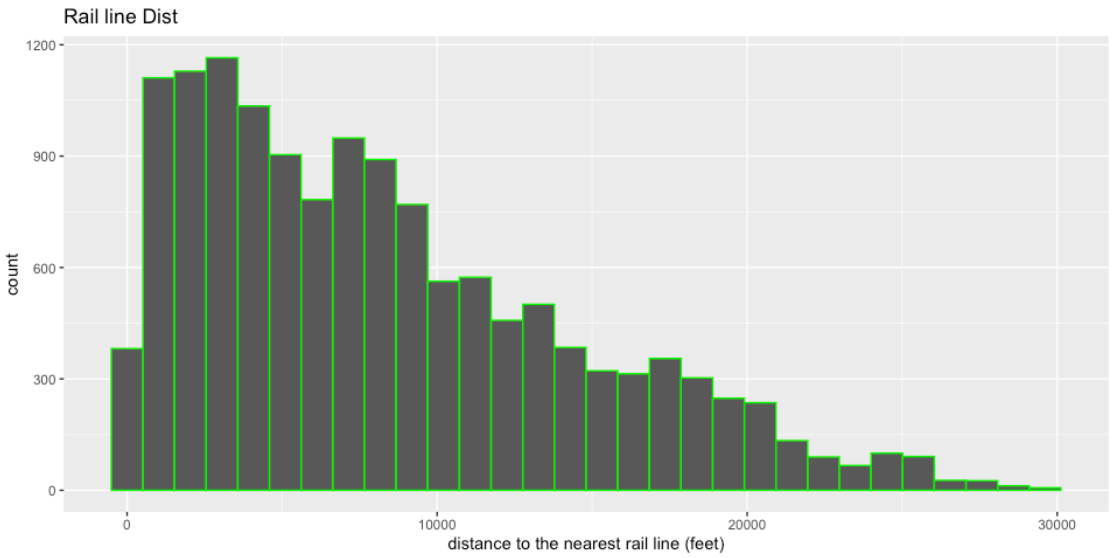
This histogram is for the variable related to the size of the floor of the house and it's obvious that the majority of the instances have a size less the 4 thousand square feet.

```
# Checking distribution of SPECIFIC FEATURE variable #
ggplot(Miami_house, aes(x = SPEC_FEAT_VAL)) + geom_histogram(color = "green",
bins=30) +
  labs(x ="value of special features ($)", title = "Special Feature")
```



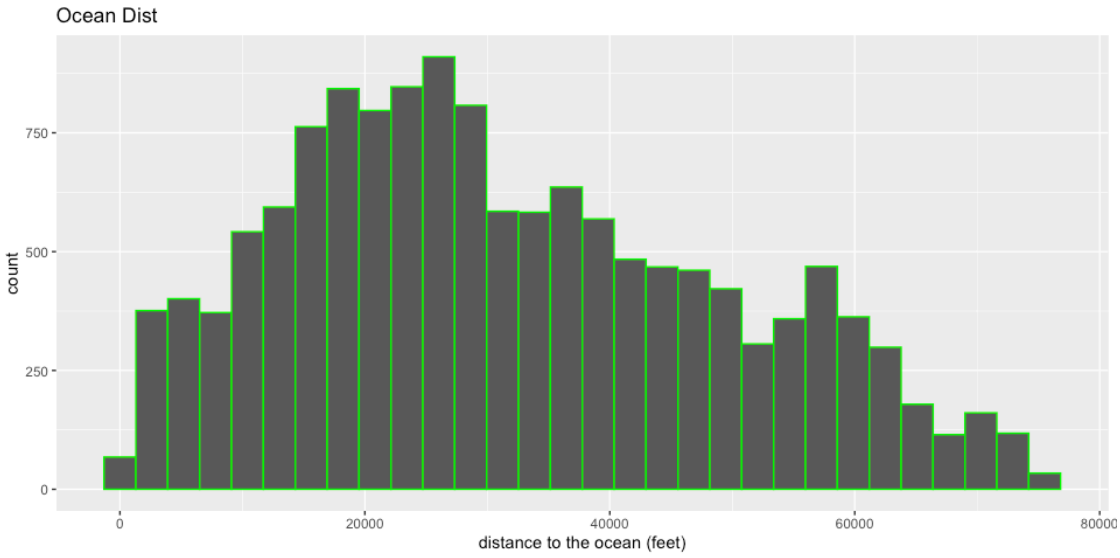
The variable related to the specific feature of the house takes two kinds of values which are 0 if there is no specific feature in the house or a number which indicated the price of the specific feature in the house if there is exist any specific feature. This histogram shows that around 2000 of the houses have no specific features and around 7000 have a specific feature with low value.

```
# Checking distribution of DISTANCE from RAILWAY variable #
ggplot(Miami_house, aes(x = RAIL_DIST)) + geom_histogram(color = "green", bin
s=30) +
  labs(x ="distance to the nearest rail line (feet)", title = "Rail line Dist
")
```



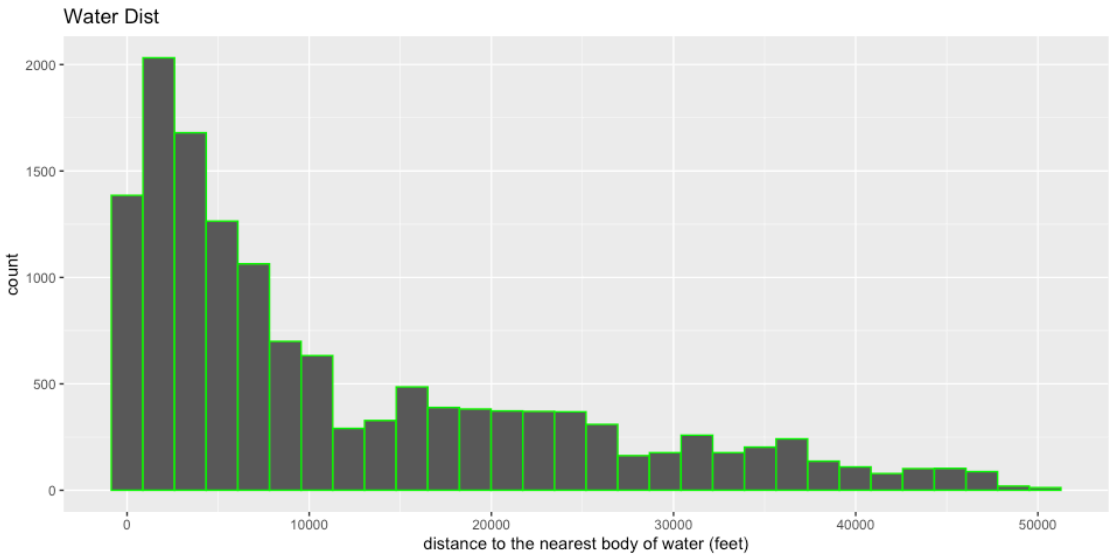
This variable related to the distance to the nearest railway is an indicator of the noise and this histogram shows that around 1200 of the houses have a distance fewer than 5000 feet and about 400 of the houses are very close to a railway and they're supposed to hear the noise of the train.

```
# Checking distribution of DISTANCE from OCEAN variable #
ggplot(Miami_house, aes(x = OCEAN_DIST)) + geom_histogram(color = "green", bi
ns=30) +
  labs(x ="distance to the ocean (feet)", title = "Ocean Dist")
```



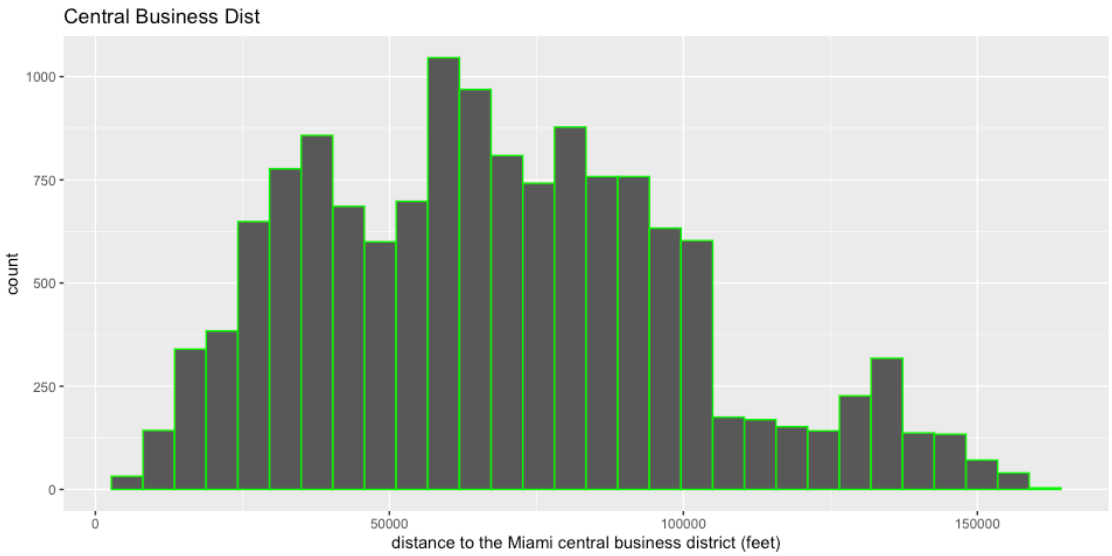
This histogram shows the distance of the houses from the ocean in feet, and it shows that a majority of the instances have a distance from the ocean of about 15000 to 40000 feet.

```
# Checking distribution of DISTANCE from WATER variable #
ggplot(Miami_house, aes(x = WATER_DIST)) + geom_histogram(color = "green", bi
ns=30) +
  labs(x ="distance to the nearest body of water (feet)", title = "Water Dist
")
```



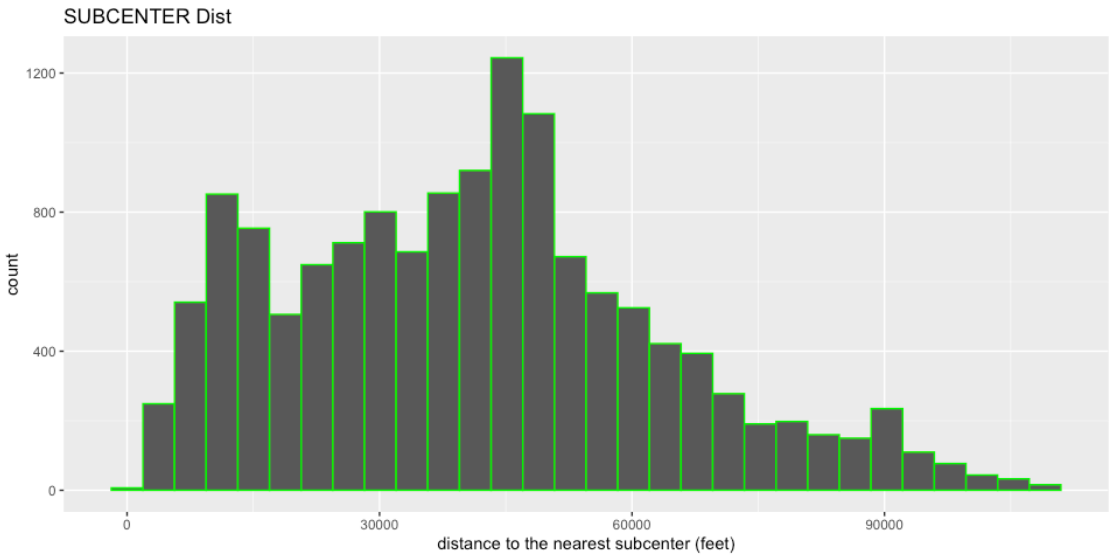
This histogram shows the distance of a house to the nearest body of water like a river or a lake and it's obvious that about 1400 of the houses are very close to a body of water and the majority of the houses have a distance of about 5000 feet.

```
# Checking distribution of DISTANCE from BUSINESS CENTER variable #
ggplot(Miami_house, aes(x = CNTR_DIST)) + geom_histogram(color = "green", bin
s=30) +
  labs(x ="distance to the Miami central business district (feet)",
       title = "Central Business Dist")
```



This variable shows the distance of houses to the Miami central business district in feet and this histogram shows that most of the houses have distances about 40000 to 100000 feet from the Miami central business district.

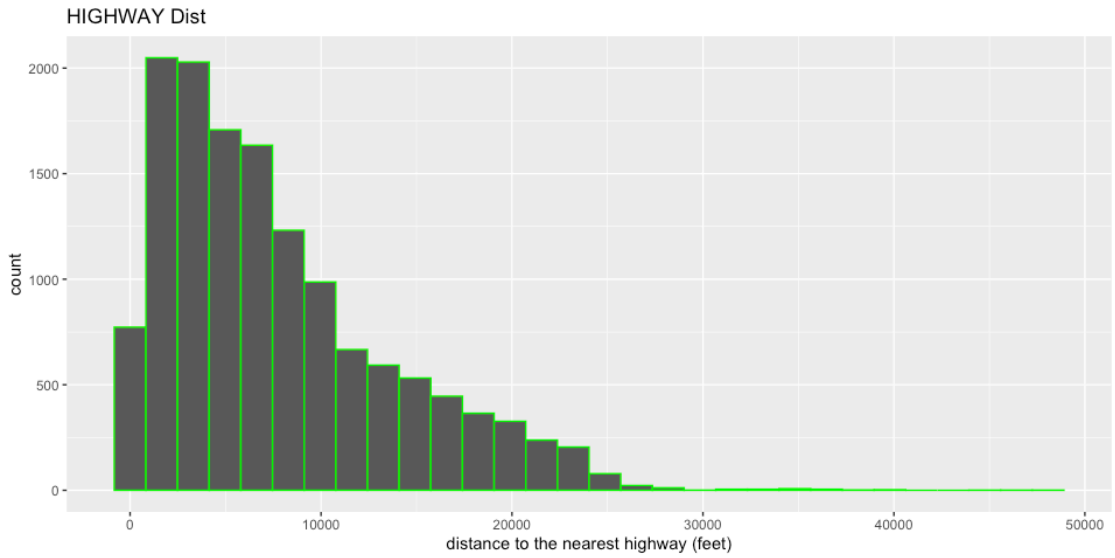
```
# Checking distribution of DISTANCE from SUBCENTER variable #
ggplot(Miami_house, aes(x = SUBCNTR_DI)) + geom_histogram(color = "green", bi
ns=30) +
  labs(x ="distance to the nearest subcenter (feet)",
       title = "SUBCENTER Dist")
```



From this histogram, it's obvious that the majority of the houses have distances around 50000 feet from the nearest subcenter in Miami.

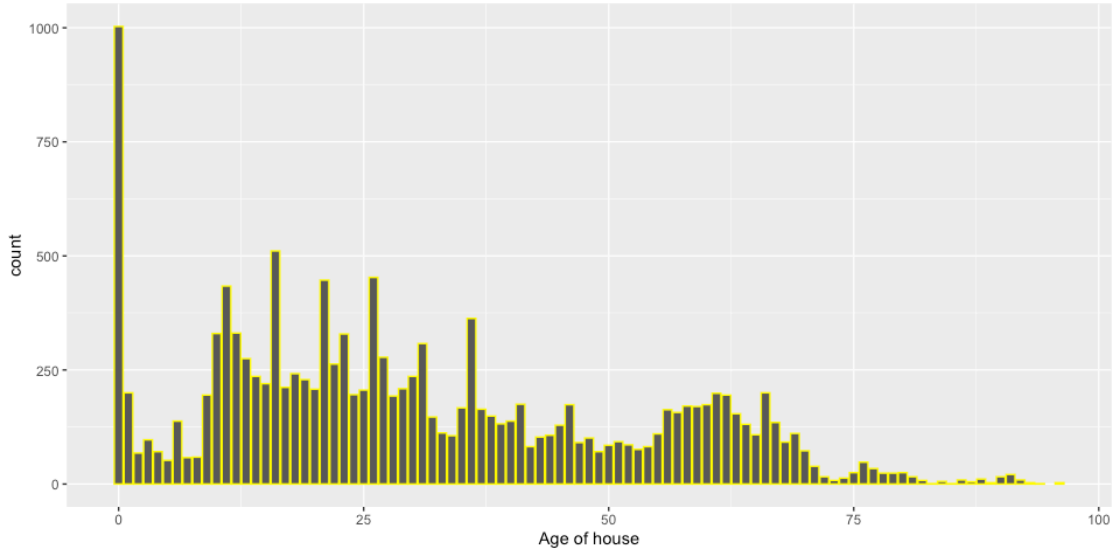
```
# Checking distribution of DISTANCE from HIGHWAY variable #
ggplot(Miami_house, aes(x = HWY_DIST)) + geom_histogram(color = "green", bins
=30) +
```

```
  labs(x ="distance to the nearest highway (feet)",
       title = "HIGHWAY Dist")
```



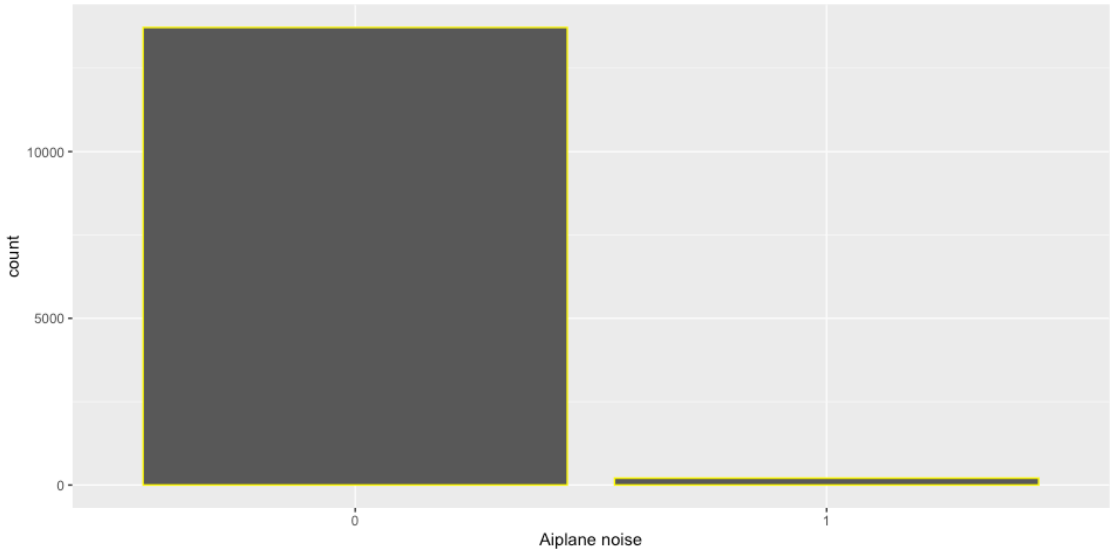
The variable HWY_DIST is related to the distance to the nearest highway which is an indicator of noise. This histogram shows that about 800 of the houses are very close to a highway and they may hear noise and most of the houses have distances about 5000 feet to a highway.

```
# Barplot of the age variable #
ggplot(Miami_house, aes(x = age)) + geom_bar(color="yellow") +
  labs(x ="Age of house")
```



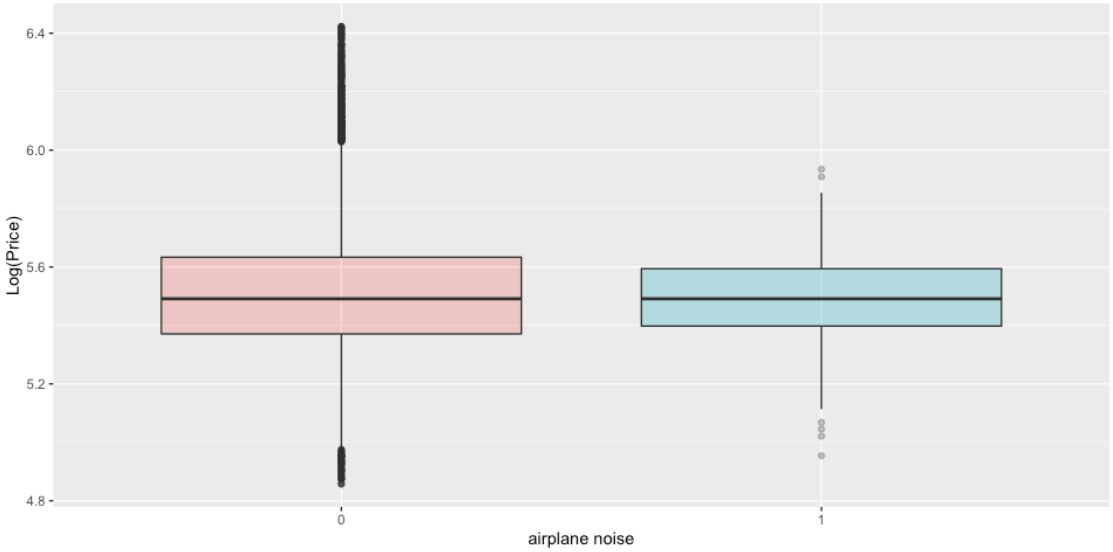
This variable age is about the number of years that the house has been built and from this barplot, it can be seen that about 1000 of the houses are very new and the small amount of the houses have aged over 75 years.

```
# Barplot of airplane noise variable #
ggplot(Miami_house, aes(x = avno60plus)) + geom_bar(color="yellow") +
  labs(x ="Aiplane noise")
```



The variable avno60plus is an indicator of airplane noise. This barplot shows that in a small number of houses the noise of the airplane exceeds an acceptable level.

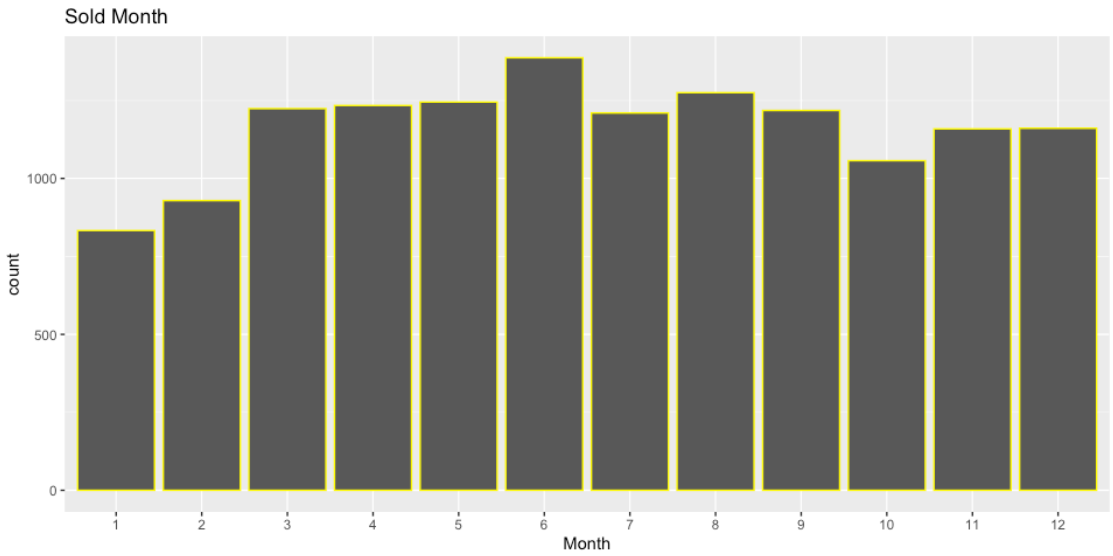
```
# Boxplot of the price concerning the noise of airplane #
ggplot(Miami_house, aes(x=avno60plus, y=log10_SALE_PRC,
fill=factor(avno60plus))) + geom_boxplot(alpha=0.3) + theme(legend.position="
none") +
labs(x = 'airplane noise', y = 'Log(Price)')
```



This boxplot shows the connection between the price of the houses and the variable related to the noise of the airplane. It's obvious that although there is a very small number of

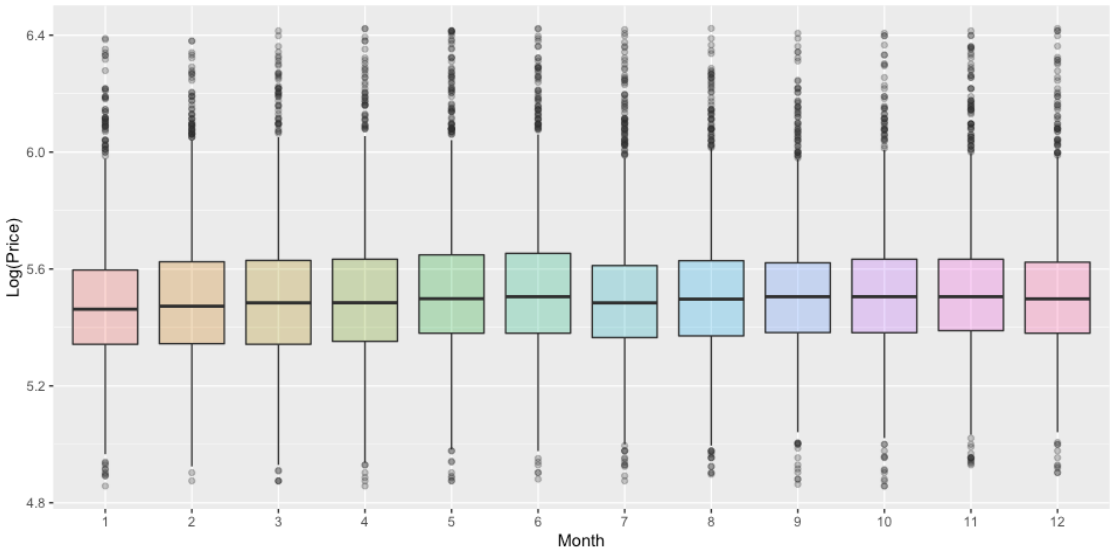
houses in which the airplane noise exceeds an acceptable level it does not have any negative effect on the price of the house.

```
# Barplot of SOLD MONTH variable #
ggplot(Miami_house, aes(x = month_sold)) + geom_bar(color="yellow") +
  labs(x ="Month",  title = "Sold Month")
```



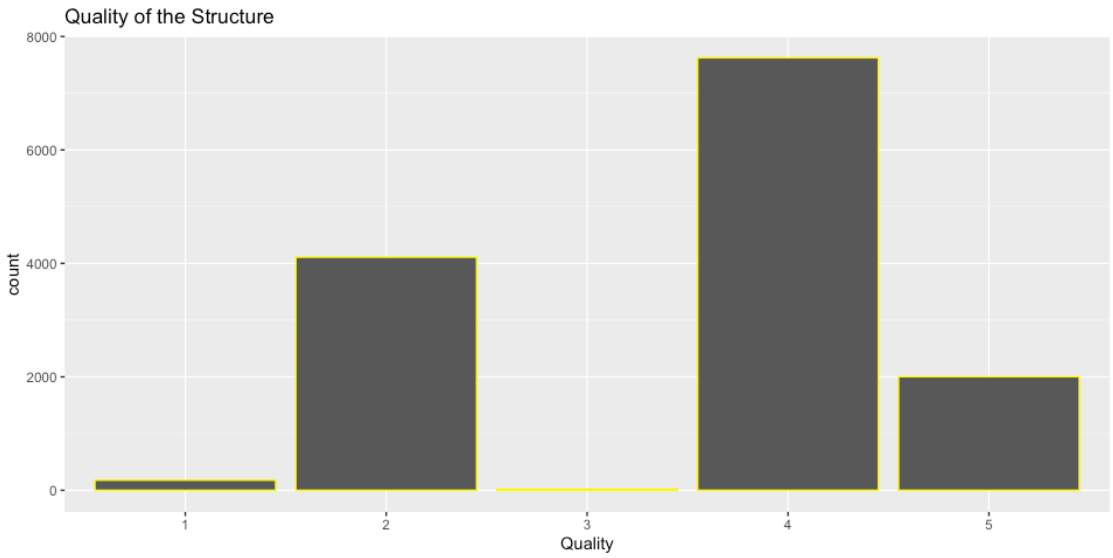
The variable month_sold shows the sale month of the house in 2016 (1 = Jan). This barplot shows that most of the houses have been sold in spring and summer.

```
# Boxplot of the price concerning the sold month #
ggplot(Miami_house, aes(x=month_sold, y=log10_SALE_PRC,
fill=factor(month_sold))) +
geom_boxplot(alpha=0.3) + theme(legend.position="none") +
  labs(x = 'Month', y = 'Log(Price)')
```



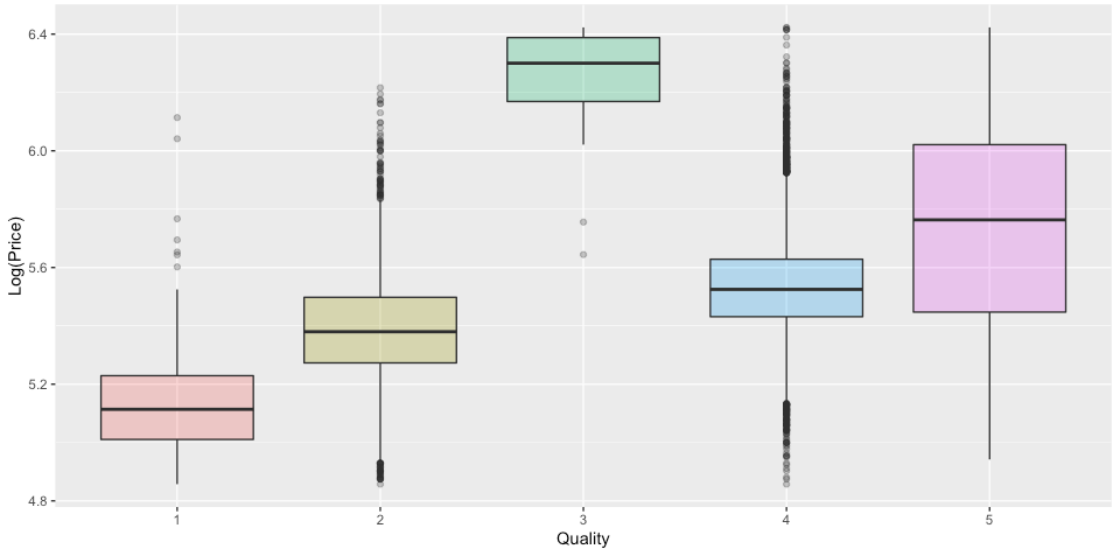
This barplot is related to the prices of the houses in different months and it's obvious that the prices are the same in the different months and only the number of the houses sold differs month to month, not the price.

```
# Barplot of STRUCTURE QUALITY variable #
ggplot(Miami_house, aes(x = structure_quality)) +
  geom_bar(color="yellow") + labs(x ="Quality",
title = "Quality of the Structure")
```



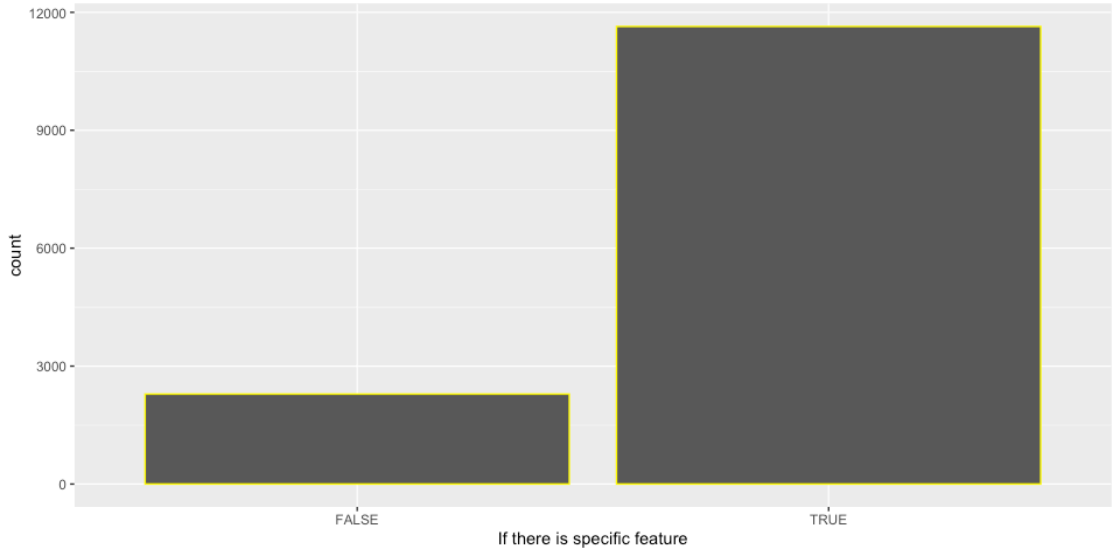
The variable structure_quality shows the quality of the structure. It takes a value between 1 to 5. This barplot shows that most of the houses have quality 4.

```
# Boxplot of the price concerning the quality of the structure#
ggplot(Miami_house, aes(x=structure_quality,
y=log10_SALE_PRC, fill=factor(structure_quality))) +
geom_boxplot(alpha=0.3) +
theme(legend.position="none") +
  labs(x = 'Quality', y = 'Log(Price)')
```



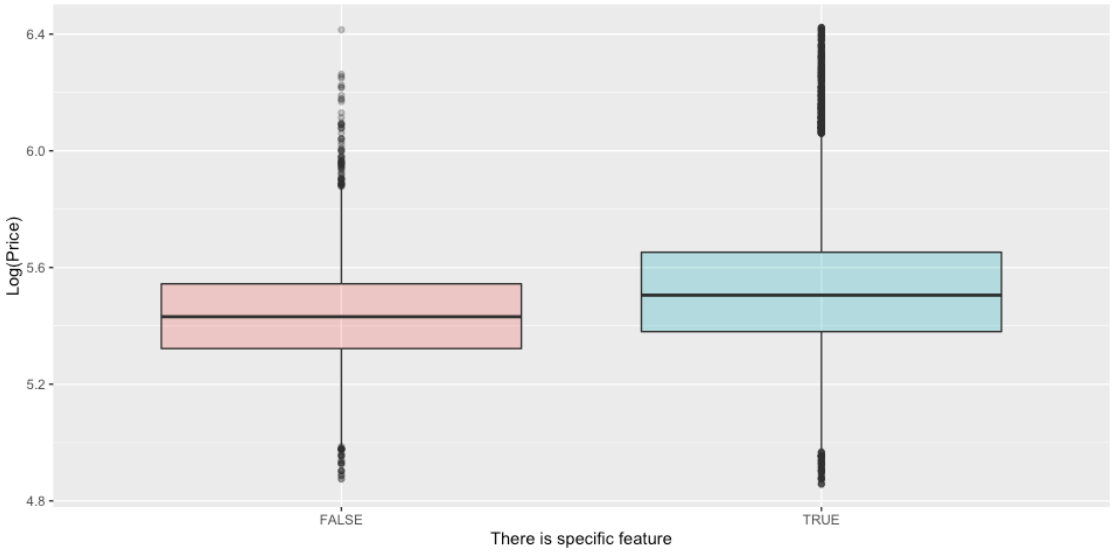
This boxplot shows that the price of the house increases by the quality value of the house. This variable has a strong effect on the price of the house.

```
# Barplot of has specific feature variable #
ggplot(Miami_house, aes(x = has_SPECFEAT)) + geom_bar(color="yellow") +
labs(x ="If there is specific feature")
```



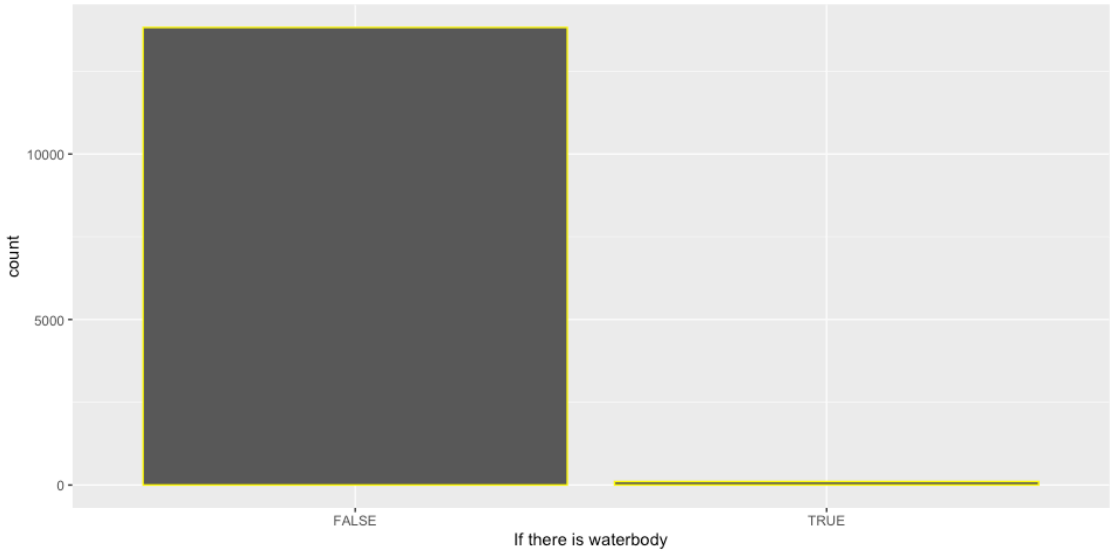
This barplot has been made from the variable related to the specific feature of the house. It shows that most of the houses have a specific feature.

```
# Boxplot of the price and if there is specific feature #
ggplot(Miami_house, aes(x=has_SPECFEAT, y=log10_SALE_PRC, fill=factor(has_SPE
CFEAT))) +
geom_boxplot(alpha=0.3) +
  theme(legend.position="none") +
  labs(x = 'There is specific feature', y = 'Log(Price)')
```



This boxplot shows the connection between the price and the feature of the houses. It shows that the houses with the specific feature have higher prices but, it's obvious that there is not much difference between the price of the houses with a specific feature and without a specific feature.

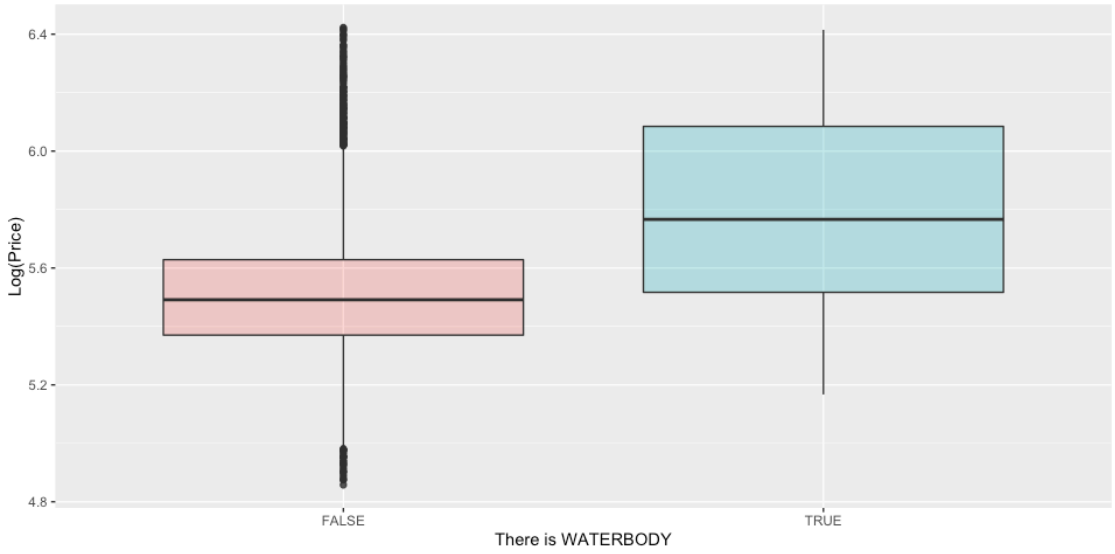
```
# Barplot of has BODYOFWATER variable #
ggplot(Miami_house, aes(x = has_BODYOFWATER)) + geom_bar(color="yellow") +
  labs(x ="If there is waterbody")
```



This variable shows if the house has a river or a lake. Most of the instances have no body of water.

```
# Boxplot of the price concerning the fact that if there is WATERBODY #
ggplot(Miami_house, aes(x=has_BODYOFWATER,
y=log10_SALE_PRC,
fill=factor(has_BODYOFWATER))) +
```

```
  geom_boxplot(alpha=0.3) + theme(legend.position="none") +
  labs(x = 'There is WATERBODY', y = 'Log(Price)')
```



There is a clear difference between the houses with a body of water and the houses without a body of water.

3. Modelling and Data analysis

This part of the study consists of three steps. In the first step, only the Numerical features have been checked. Then in the second step, only the variables with categorical type have been taken into account. In the last step, the chosen variables of each type have been combined.

3.1. Numerical Variables

First, to begin this step the correlation between the target variable, logarithmic price, and other numerical variables has been checked.

```
numerical_variables <- subset(Miami_house, select =
c(log10_SALE_PRC, LND_SQFOOT, TOT_LVG_AREA, SPEC_FEAT_VAL, RAIL_DIST,
OCEAN_DIST, WATER_DIST, CNTR_DIST,
SUBCNTR_DI, HWY_DIST, LATITUDE, LONGITUDE))
round(cor(numerical_variables[]), 3)
```

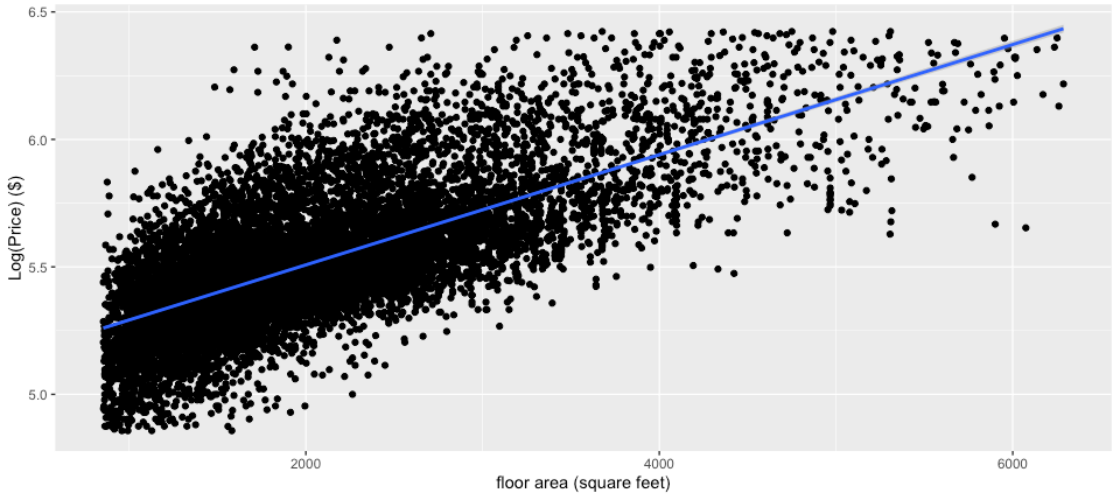
```
##                log10_SALE_PRC LND_SQFOOT TOT_LVG_AREA SPEC_FEAT_VAL RAIL_D
IST
## log10_SALE_PRC          1.000      0.370        0.713         0.506    -0.
073
## LND_SQFOOT              0.370      1.000        0.437         0.391    -0.
084
## TOT_LVG_AREA            0.713      0.437        1.000         0.506     0.
075
```

```
## SPEC_FEAT_VAL           0.506      0.391        0.506         1.000    -0.
022
## RAIL_DIST              -0.073     -0.084        0.075        -0.022     1.
000
## OCEAN_DIST             -0.209     -0.162       -0.050        -0.055     0.
259
## WATER_DIST             -0.030     -0.055        0.148         0.014     0.
162
## CNTR_DIST              -0.233     -0.023        0.137        -0.049     0.
444
## SUBCNTR_DI             -0.404     -0.159       -0.045        -0.152     0.
485
## HWY_DIST                0.295      0.130        0.229         0.154    -0.
092
## LATITUDE                0.010     -0.077       -0.194        -0.008    -0.
172
## LONGITUDE               0.083      0.018       -0.181        -0.009    -0.
303
##                OCEAN_DIST WATER_DIST CNTR_DIST SUBCNTR_DI HWY_DIST LATITUD
E
## log10_SALE_PRC     -0.209     -0.030    -0.233     -0.404    0.295    0.01
0
## LND_SQFOOT         -0.162     -0.055    -0.023     -0.159    0.130   -0.07
7
## TOT_LVG_AREA       -0.050      0.148     0.137     -0.045    0.229   -0.19
4
## SPEC_FEAT_VAL      -0.055      0.014    -0.049     -0.152    0.154   -0.00
8
## RAIL_DIST           0.259      0.162     0.444      0.485   -0.092   -0.17
2
## OCEAN_DIST          1.000      0.491     0.245      0.426    0.094    0.24
3
## WATER_DIST          0.491      1.000     0.527      0.195    0.400   -0.42
3
## CNTR_DIST           0.245      0.527     1.000      0.766    0.076   -0.71
7
## SUBCNTR_DI          0.426      0.195     0.766      1.000   -0.094   -0.19
6
## HWY_DIST            0.094      0.400     0.076     -0.094    1.000   -0.11
3
## LATITUDE            0.243     -0.423    -0.717     -0.196   -0.113    1.00
0
## LONGITUDE          -0.457     -0.764    -0.792     -0.380   -0.216    0.72
1
##                LONGITUDE
## log10_SALE_PRC     0.083
## LND_SQFOOT         0.018
## TOT_LVG_AREA      -0.181
## SPEC_FEAT_VAL     -0.009
## RAIL_DIST         -0.303
```

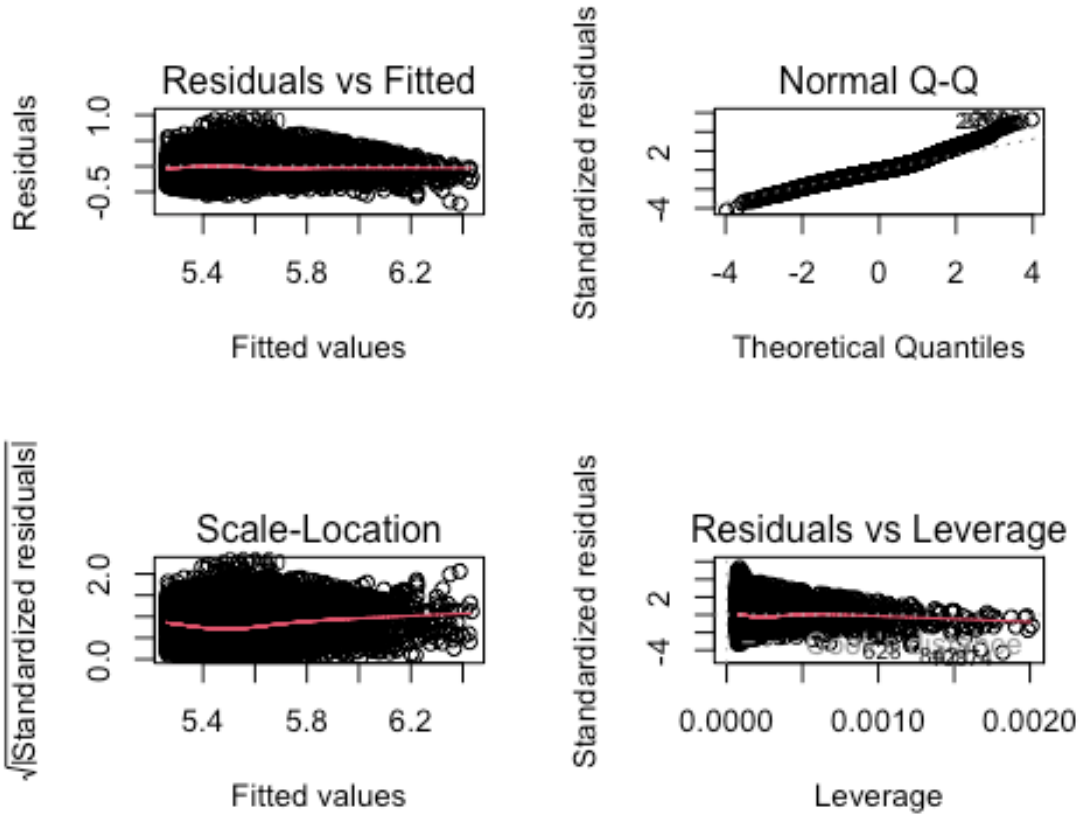
```
## OCEAN_DIST        -0.457
## WATER_DIST        -0.764
## CNTR_DIST         -0.792
## SUBCNTR_DI        -0.380
## HWY_DIST          -0.216
## LATITUDE           0.721
## LONGITUDE          1.000
```

As it can be seen, there is a high correlation between the variable TOT_LVG_AREA, which is the indicator of the floor area of the house, and the logarithm of price. So, it should be checked to see which degree of regression is the best.

```
# Linear model #
model.num1 <- lm(data = Miami_house, log10_SALE_PRC ~ TOT_LVG_AREA)
summary(model.num1)

##
## Call:
## lm(formula = log10_SALE_PRC ~ TOT_LVG_AREA, data = Miami_house)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.73501 -0.10746 -0.01606  0.08546  0.91663
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.076e+00  3.985e-03    1274   <2e-16 ***
## TOT_LVG_AREA 2.161e-04  1.801e-06     120   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1729 on 13930 degrees of freedom
## Multiple R-squared:  0.5083, Adjusted R-squared:  0.5083
## F-statistic: 1.44e+04 on 1 and 13930 DF,  p-value: < 2.2e-16

BIC(model.num1)

## [1] -9335.795

# Plotting model #
ggplot(Miami_house, aes(x = TOT_LVG_AREA, y = log10_SALE_PRC)) +
  geom_point() + stat_smooth(method = "lm", formula = y ~ x) +
  labs(x = 'floor area (square feet)', y = 'Log(Price) ($)')
```



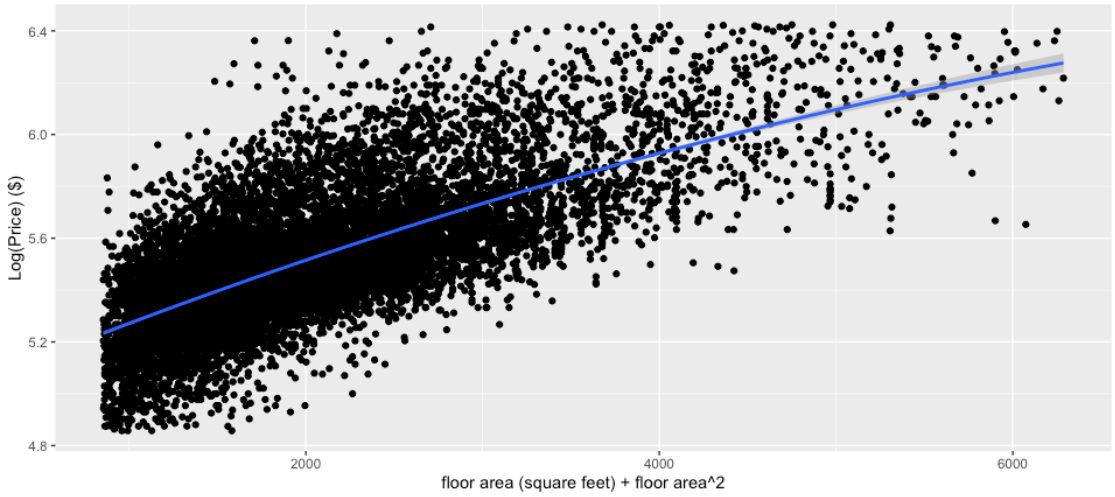

```
# Diagnostic #
par(mfrow=c(2,2))
plot(model.num1)
```



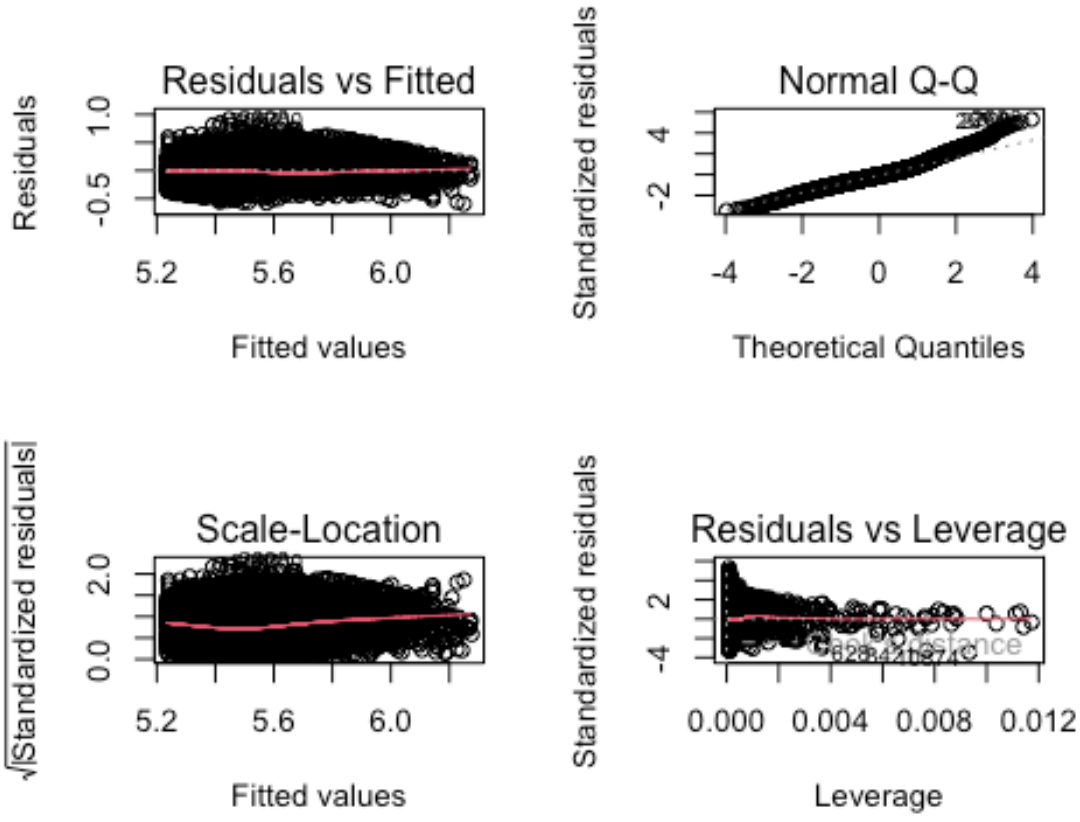
The simple linear regression can interpolate the data to some extent. The plot of the residuals shows that residuals are somehow randomly distributed. The scale-location plot, which is a type of plot that displays the fitted values of the regression model along the x-axis and the square root of the standardized residuals along the y-axis, shows that although

the red line is not flat, it's somehow horizontal across the plot and the assumption of homoscedasticity is almost satisfied for this regression model. The simple linear model is satisfying but for the sake of improvement, the second-degree polynomial model has been tested.

```
par(mfrow=c(1,1))

# Polynomial model degree = 2 #
model.num2 <- lm(data=Miami_house, log10_SALE_PRC ~ TOT_LVG_AREA + I(TOT_LVG_
AREA**2))
summary(model.num2)

##
## Call:
## lm(formula = log10_SALE_PRC ~ TOT_LVG_AREA + I(TOT_LVG_AREA^2),
##     data = Miami_house)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.59598 -0.10963 -0.01754  0.08647  0.91463
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         5.002e+00  8.815e-03 567.480   <2e-16 ***
## TOT_LVG_AREA        2.816e-04  7.261e-06  38.779   <2e-16 ***
## I(TOT_LVG_AREA^2) -1.256e-08  1.350e-09  -9.306   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1724 on 13929 degrees of freedom
## Multiple R-squared:  0.5114, Adjusted R-squared:  0.5113
## F-statistic:  7289 on 2 and 13929 DF,  p-value: < 2.2e-16

BIC(model.num2)

## [1] -9412.609

# Plotting model #
ggplot(Miami_house, aes(x = TOT_LVG_AREA, y = log10_SALE_PRC)) +
  geom_point() + stat_smooth(method = "lm", formula = y ~ x + I(x**2)) +
  labs(x = 'floor area (square feet) + floor area^2', y = 'Log(Price) ($)')
```




```
# Diagnostic #
par(mfrow=c(2,2))
plot(model.num2)
```

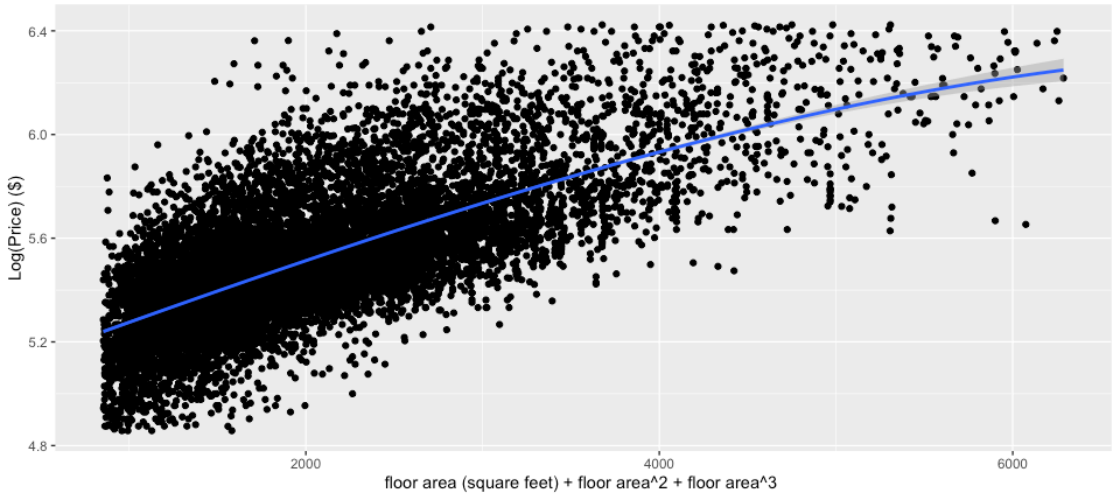



```
par(mfrow=c(1,1))

anova(model.num1, model.num2)
```

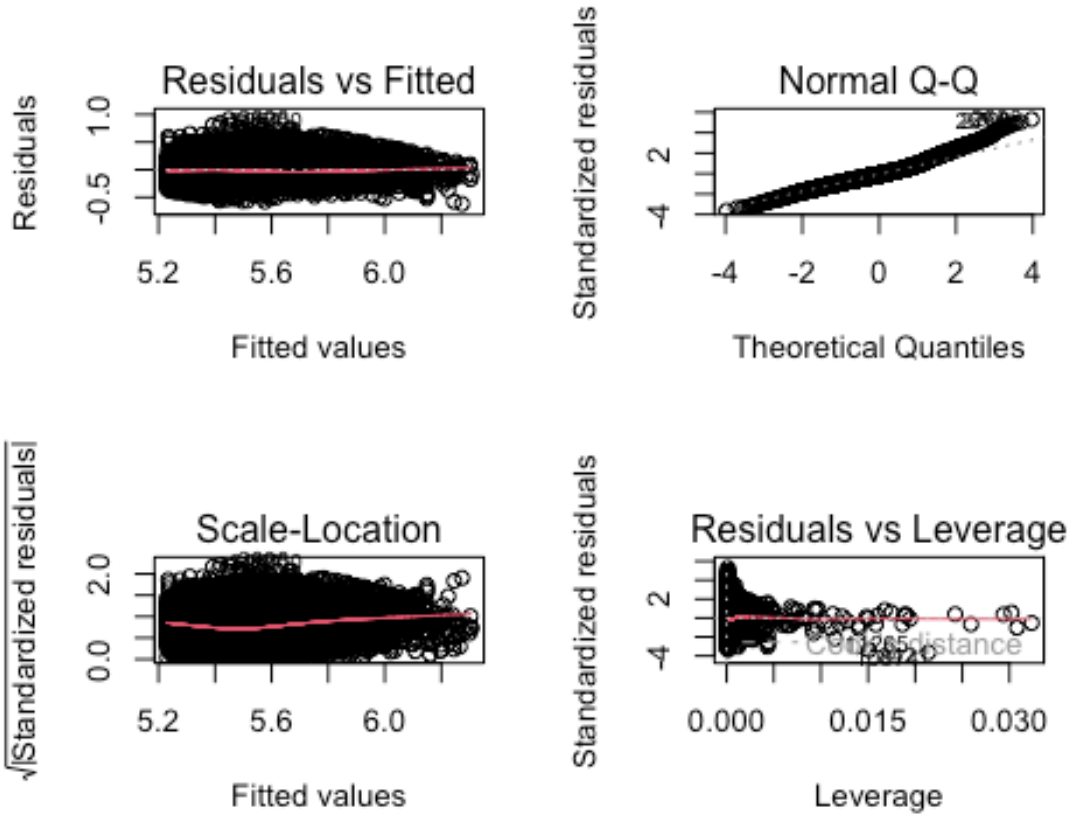
```
## Analysis of Variance Table
##
## Model 1: log10_SALE_PRC ~ TOT_LVG_AREA
## Model 2: log10_SALE_PRC ~ TOT_LVG_AREA + I(TOT_LVG_AREA^2)
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1  13930 416.51
## 2  13929 413.94  1    2.5737 86.605 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The regression line now fits a little bit better the data and residuals plot shows a little improvement and also the scale-location plot is flatter. Having such a small P-value shows the model including TOT_LVG_AREA^2 provides a better fit than the previous model. For a further check, the 3rd-degree polynomial for this feature is checked.

```
# Polynomial model degree 3 #
model.num3 <- lm(data=Miami_house, log10_SALE_PRC ~ TOT_LVG_AREA +
I(TOT_LVG_AREA**2) + I(TOT_LVG_AREA**3))
summary(model.num3)

##
## Call:
## lm(formula = log10_SALE_PRC ~ TOT_LVG_AREA + I(TOT_LVG_AREA^2) +
##     I(TOT_LVG_AREA^3), data = Miami_house)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.61981 -0.10948 -0.01786  0.08604  0.91349
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         4.980e+00  1.947e-02 255.745  < 2e-16 ***
## TOT_LVG_AREA        3.101e-04  2.380e-05  13.029  < 2e-16 ***
## I(TOT_LVG_AREA^2) -2.347e-08  8.779e-09  -2.674  0.00751 **
## I(TOT_LVG_AREA^3)  1.227e-12  9.759e-13   1.258  0.20851
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1724 on 13928 degrees of freedom
## Multiple R-squared:  0.5114, Adjusted R-squared:  0.5113
## F-statistic:  4860 on 3 and 13928 DF,  p-value: < 2.2e-16

BIC(model.num3)

## [1] -9404.65

# Plotting model #
ggplot(Miami_house, aes(x = TOT_LVG_AREA, y = log10_SALE_PRC)) + geom_point()
+ stat_smooth(method = "lm", formula = y ~ x + I(x**3)) +
labs(x = 'floor area (square feet) + floor area^2 + floor area^3 ',
y = 'Log(Price) ($)')
```




```
# Diagnostic #
par(mfrow=c(2,2))
plot(model.num3)
```




```
par(mfrow=c(1,1))

anova(model.num2, model.num3)
```

```
## Analysis of Variance Table
##
## Model 1: log10_SALE_PRC ~ TOT_LVG_AREA + I(TOT_LVG_AREA^2)
## Model 2: log10_SALE_PRC ~ TOT_LVG_AREA + I(TOT_LVG_AREA^2) + I(TOT_LVG_ARE
A^3)
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1  13929 413.94
## 2  13928 413.89  1  0.047009 1.5819 0.2085
```

Now the results show that adding the 3rd degree is not a good idea and the ANOVA function shows that the P-value is not a small number and it's not significant. So, the best model for the variable TOT_LVG_AREA is (model.num2).

The next step is the best subset selection step for other numerical variables by using the regsubsets() function which performs the best subset selection by identifying the best model.

```
# Using Regsubset() function #
regfit.num <- regsubsets(log10_SALE_PRC ~ LND_SQFOOT + TOT_LVG_AREA +
SPEC_FEAT_VAL + RAIL_DIST + OCEAN_DIST + WATER_DIST + CNTR_DIST +
SUBCNTR_DI + HWY_DIST + LATITUDE + LONGITUDE, data=Miami_house, nvmax = 11)

reg.summary <- summary(regfit.num)
reg.summary

## Subset selection object
## Call: regsubsets.formula(log10_SALE_PRC ~ LND_SQFOOT + TOT_LVG_AREA +
##     SPEC_FEAT_VAL + RAIL_DIST + OCEAN_DIST + WATER_DIST + CNTR_DIST +
##     SUBCNTR_DI + HWY_DIST + LATITUDE + LONGITUDE, data = Miami_house,
##     nvmax = 11)
## 11 Variables  (and intercept)
##               Forced in Forced out
## LND_SQFOOT        FALSE      FALSE
## TOT_LVG_AREA      FALSE      FALSE
## SPEC_FEAT_VAL     FALSE      FALSE
## RAIL_DIST         FALSE      FALSE
## OCEAN_DIST        FALSE      FALSE
## WATER_DIST        FALSE      FALSE
## CNTR_DIST         FALSE      FALSE
## SUBCNTR_DI        FALSE      FALSE
## HWY_DIST          FALSE      FALSE
## LATITUDE          FALSE      FALSE
## LONGITUDE         FALSE      FALSE
## 1 subsets of each size up to 11
## Selection Algorithm: exhaustive
##          LND_SQFOOT TOT_LVG_AREA SPEC_FEAT_VAL RAIL_DIST OCEAN_DIST WATER
_DIST
## 1  ( 1 )  " "        "*"          " "           " "       " "         " "
## 2  ( 1 )  " "        "*"          " "           " "       " "         " "
## 3  ( 1 )  " "        "*"          "*"           " "       " "         " "
```

```
## 4  ( 1 )  " "         "*"           " "           " "         " "         "*"
## 5  ( 1 )  " "         "*"           "*"           " "         " "         "*"
## 6  ( 1 )  " "         "*"           "*"           "*"         " "         "*"
## 7  ( 1 )  " "         "*"           "*"           "*"         " "         "*"
## 8  ( 1 )  " "         "*"           "*"           "*"         "*"         "*"
## 9  ( 1 )  " "         "*"           "*"           "*"         "*"         "*"
## 10  ( 1 ) "*"         "*"           "*"           "*"         "*"         "*"
## 11  ( 1 ) "*"         "*"           "*"           "*"         "*"         "*"
##           CNTR_DIST SUBCNTR_DI HWY_DIST LATITUDE LONGITUDE
## 1  ( 1 )  " "       " "        " "      " "      " "
## 2  ( 1 )  " "       "*"        " "      " "      " "
## 3  ( 1 )  " "       "*"        " "      " "      " "
## 4  ( 1 )  " "       "*"        "*"      " "      " "
## 5  ( 1 )  " "       "*"        "*"      " "      " "
## 6  ( 1 )  " "       "*"        "*"      " "      " "
## 7  ( 1 )  " "       "*"        "*"      "*"      " "
## 8  ( 1 )  " "       "*"        "*"      "*"      " "
## 9  ( 1 )  " "       "*"        "*"      "*"      "*"
## 10  ( 1 ) " "       "*"        "*"      "*"      "*"
## 11  ( 1 ) "*"       "*"        "*"      "*"      "*"
```

Here the best 11 models can be seen. An asterisk ("*") indicates that a given variable is included in the corresponding model. For instance, this output indicates that the best two-variable model contains only TOT_LVG_AREA and SUBCNTR_DI.

```
reg.summary$rsq
```

```
##  [1] 0.5083367 0.6466949 0.6595746 0.6707769 0.6814560 0.6883694 0.6900921
##  [8] 0.6906771 0.6916633 0.6921789 0.6924320
```

We see that the R2 statistic increases from 50% when only one variable is included in the model to almost 69% when all variables are included. As expected, the R2 statistic increases monotonically when more variables are included.

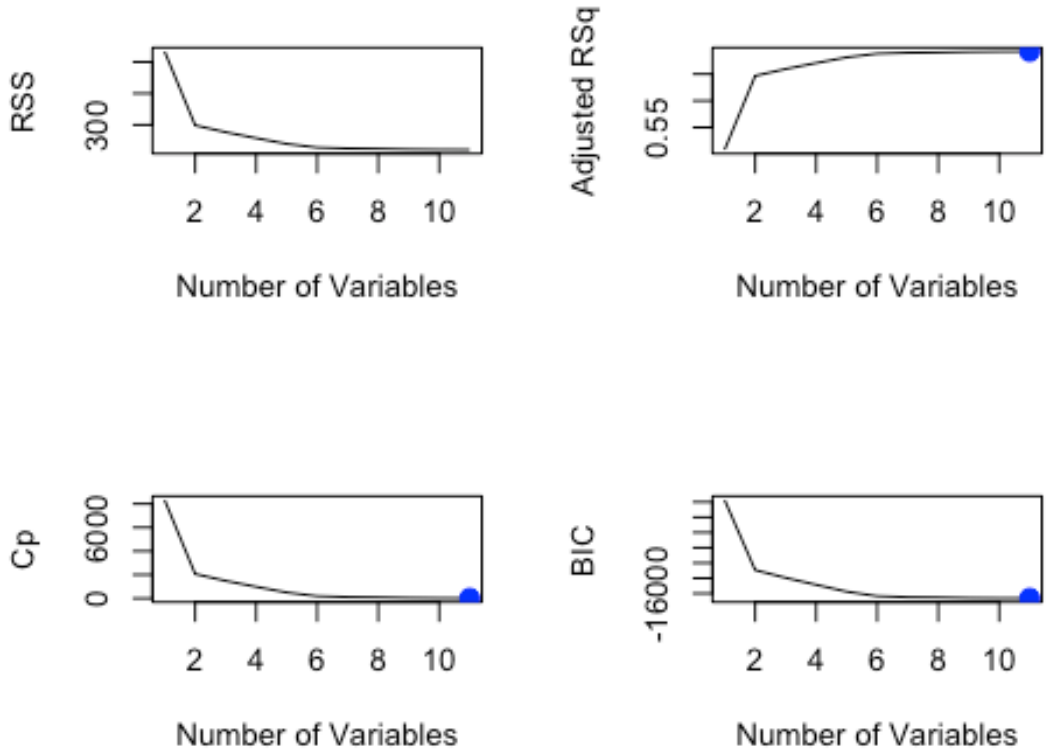
Plotting RSS, and adjusted R2, Cp, and BIC for all of the models at once will help to decide which model should be selected.

```
par(mfrow=c(2,2))

# residual sum of squares
plot(reg.summary$rss, xlab="Number of Variables", ylab="RSS", type="l")

# adjusted-R^2 with its largest value
plot(reg.summary$adjr2, xlab="Number of Variables", ylab="Adjusted RSq", type
="l")
which.max(reg.summary$adjr2)
```

```
## [1] 11
```

```
points(11, reg.summary$adjr2[11], col="blue",cex=2,pch=20)
```

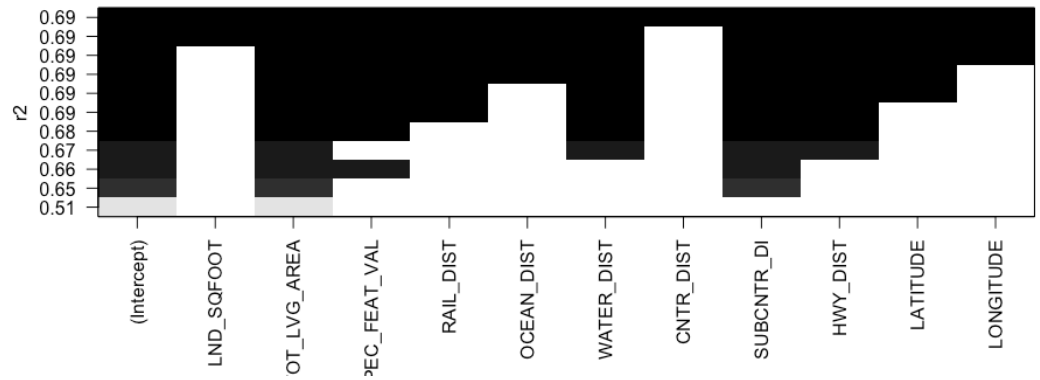
```
# Mallow's Cp with its smallest value
plot(reg.summary$cp, xlab="Number of Variables", ylab="Cp", type='l')
which.min(reg.summary$cp)

## [1] 11

points(11, reg.summary$cp[11], col="blue", cex=2, pch=20)

# BIC with its smallest value
plot(reg.summary$bic, xlab="Number of Variables", ylab="BIC", type='l')
which.min(reg.summary$bic)

## [1] 11

points(11, reg.summary$bic[11], col="blue", cex=2, pch=20)
```



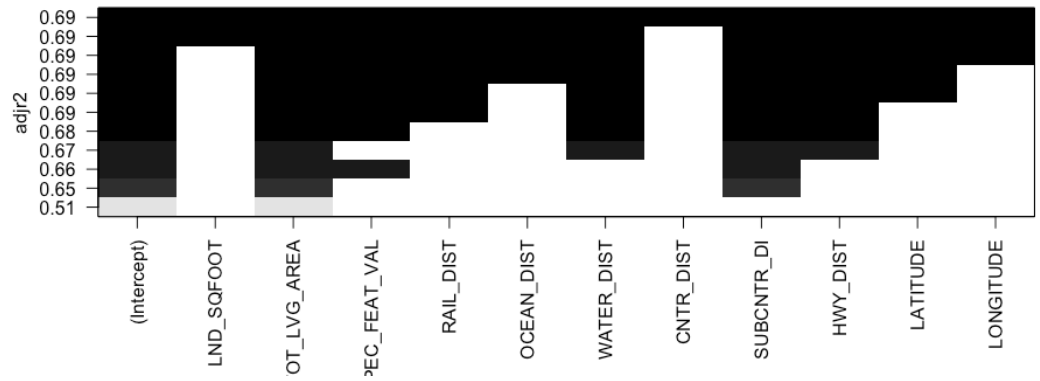

```
par(mfrow=c(1,1))
```

The RSS plot shows that the RSS decreases as the number of variables increase and the Adjusted R2 plot shows that it increases as the number of variables increase, The Cp plot admits that it decreases as the number of the variables increase this fact is also true for the BIC plot. The plot() command indicates the selected variables for the best model, ranked according to a chosen statistic. The top row of each plot contains a black square for each variable selected according to the optimal model associated with that statistic.

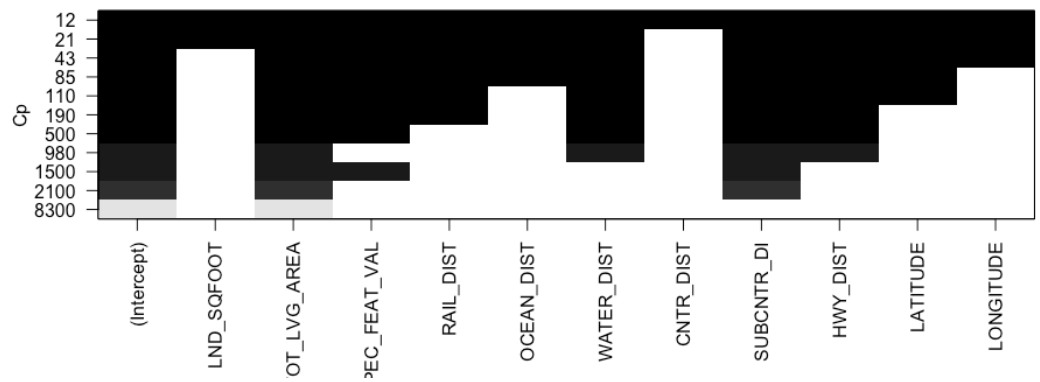
```
plot(regfit.num,scale="r2")
```



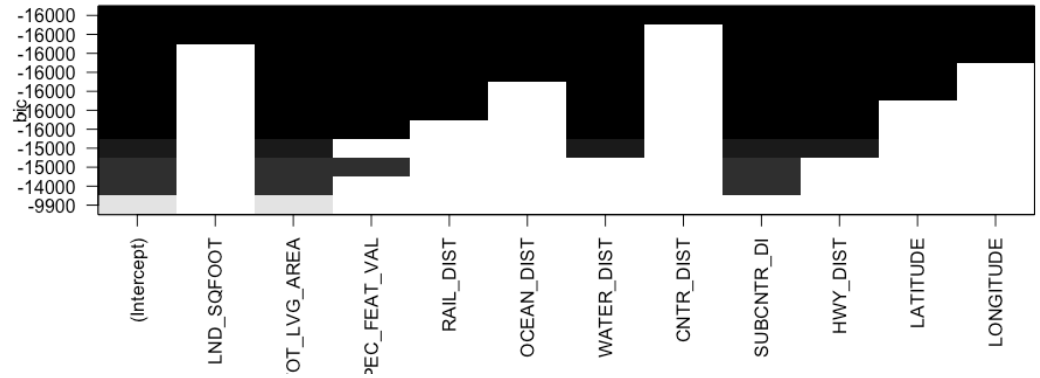

```
plot(regfit.num,scale="adjr2")
```




```
plot(regfit.num,scale="Cp")
```




```
plot(regfit.num,scale="bic")
```




```
coef(regfit.num, 6)
```

```
##   (Intercept)  TOT_LVG_AREA SPEC_FEAT_VAL     RAIL_DIST    WATER_DIST
##  5.234276e+00  1.850578e-04  2.133237e-06  3.854344e-06 -2.758651e-06
##    SUBCNTR_DI      HWY_DIST
## -4.041813e-06  6.703182e-06
```

```
# Best model "BIC"
best.bic <- lm(log10_SALE_PRC ~ TOT_LVG_AREA + SPEC_FEAT_VAL + RAIL_DIST +
WATER_DIST + SUBCNTR_DI + HWY_DIST, data=Miami_house)
summary(best.bic)
```

```
##
## Call:
## lm(formula = log10_SALE_PRC ~ TOT_LVG_AREA + SPEC_FEAT_VAL +
##     RAIL_DIST + WATER_DIST + SUBCNTR_DI + HWY_DIST, data = Miami_house)
##
## Residuals:
```

```
##      Min       1Q   Median       3Q      Max
## -0.59955 -0.07172  0.00180  0.07691  0.78157
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    5.234e+00  4.104e-03 1275.54   <2e-16 ***
## TOT_LVG_AREA   1.851e-04  1.707e-06  108.39   <2e-16 ***
## SPEC_FEAT_VAL  2.133e-06  9.873e-08   21.61   <2e-16 ***
## RAIL_DIST      3.854e-06  2.193e-07   17.58   <2e-16 ***
## WATER_DIST    -2.759e-06  1.114e-07  -24.75   <2e-16 ***
## SUBCNTR_DI    -4.042e-06  6.196e-08  -65.24   <2e-16 ***
## HWY_DIST       6.703e-06  2.190e-07   30.61   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1377 on 13925 degrees of freedom
## Multiple R-squared:  0.6884, Adjusted R-squared:  0.6882
## F-statistic:  5127 on 6 and 13925 DF,  p-value: < 2.2e-16
```

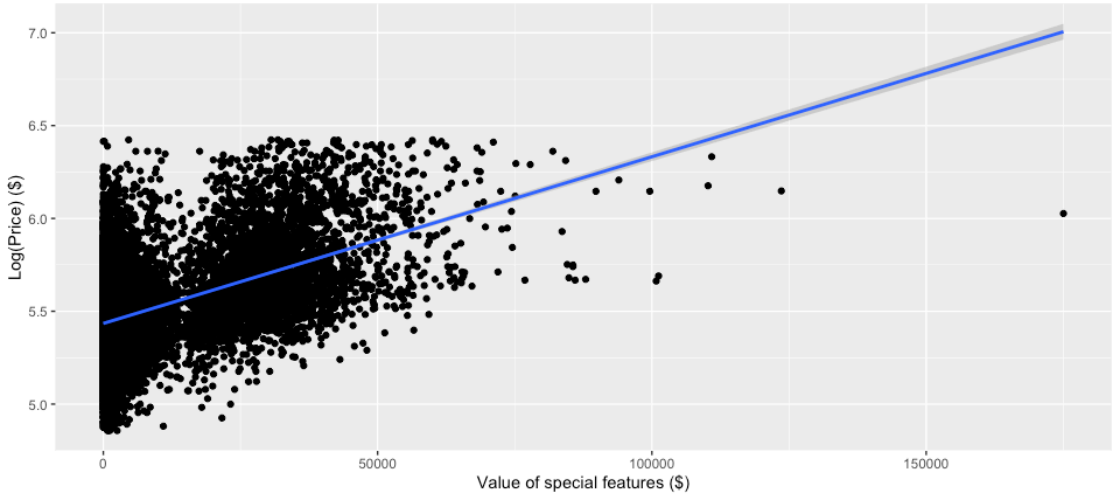
In this study, BIC has been used to evaluate the models to obtain a model with a lower number of variables. The plots showed when the number of variables exceeds number 6 there is not a big difference between the model with 6 variables and the one with 8 variables. So the model in this study has the following numerical variables.

1. TOT_LVG_AREA
2. SUBCNTR_DI
3. SPEC_FEAT_VAL
4. HWY_DIST
5. WATER_DIST
6. RAIL_DIST

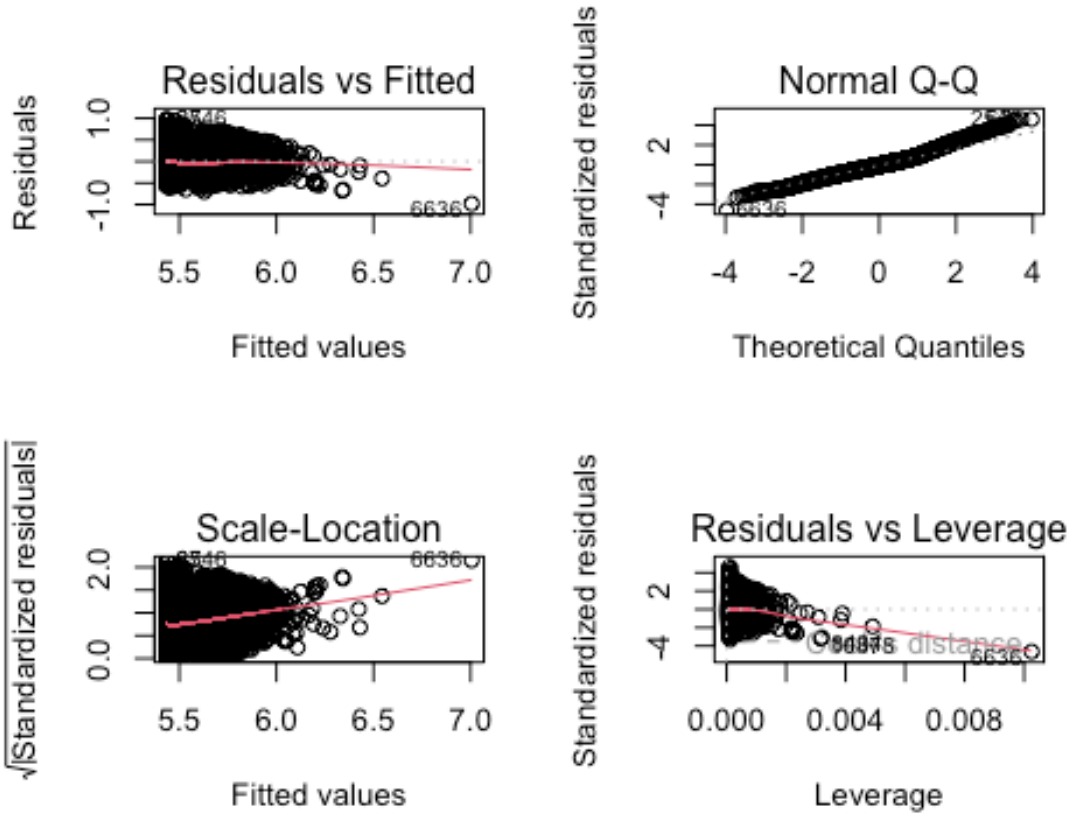
The next step is to check the best degree for each selected variable to have a better fit of the regression line in the plot of each variable against the response variable.

```
# log(price) vs Special Feature variable #
model.SPEC_FEAT_VAL <- lm(data=Miami_house, log10_SALE_PRC ~ SPEC_FEAT_VAL)
summary(model.SPEC_FEAT_VAL)

##
## Call:
## lm(formula = log10_SALE_PRC ~ SPEC_FEAT_VAL, data = Miami_house)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.97906 -0.13603 -0.01231  0.10887  0.98051
##
```

```
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   5.434e+00  2.188e-03 2483.45   <2e-16 ***
## SPEC_FEAT_VAL 8.977e-06  1.298e-07   69.18   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2128 on 13930 degrees of freedom
## Multiple R-squared:  0.2557, Adjusted R-squared:  0.2557
## F-statistic:  4786 on 1 and 13930 DF,  p-value: < 2.2e-16

BIC(model.SPEC_FEAT_VAL)

## [1] -3559.431

# Plotting model #
ggplot(Miami_house, aes(x = SPEC_FEAT_VAL, y = log10_SALE_PRC)) +
geom_point()+ stat_smooth(method = "lm", formula = y ~ x) +
labs(x = 'Value of special features ($)', y = 'Log(Price) ($)')
```




```
# Diagnostic
par(mfrow=c(2,2))
plot(model.SPEC_FEAT_VAL)
```

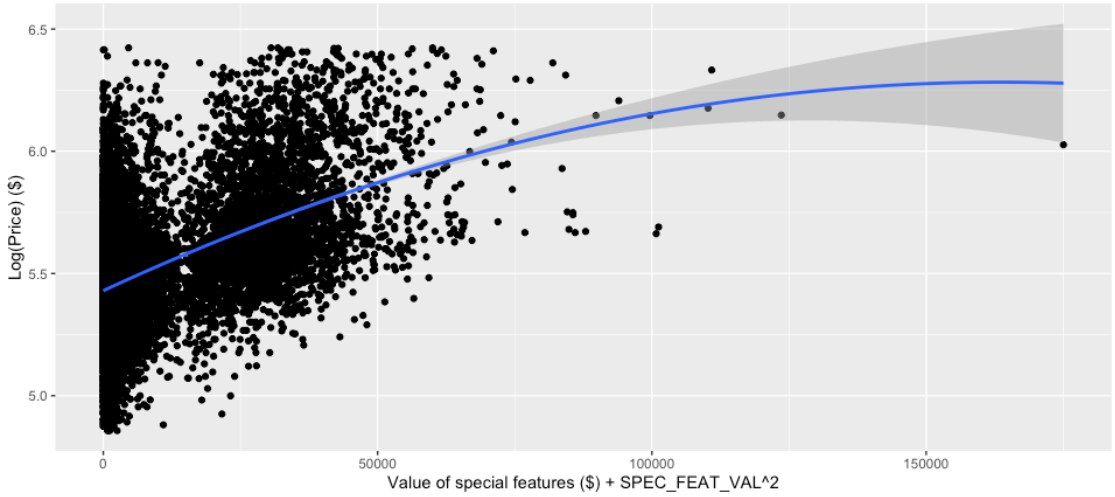



```
par(mfrow=c(1,1))
```

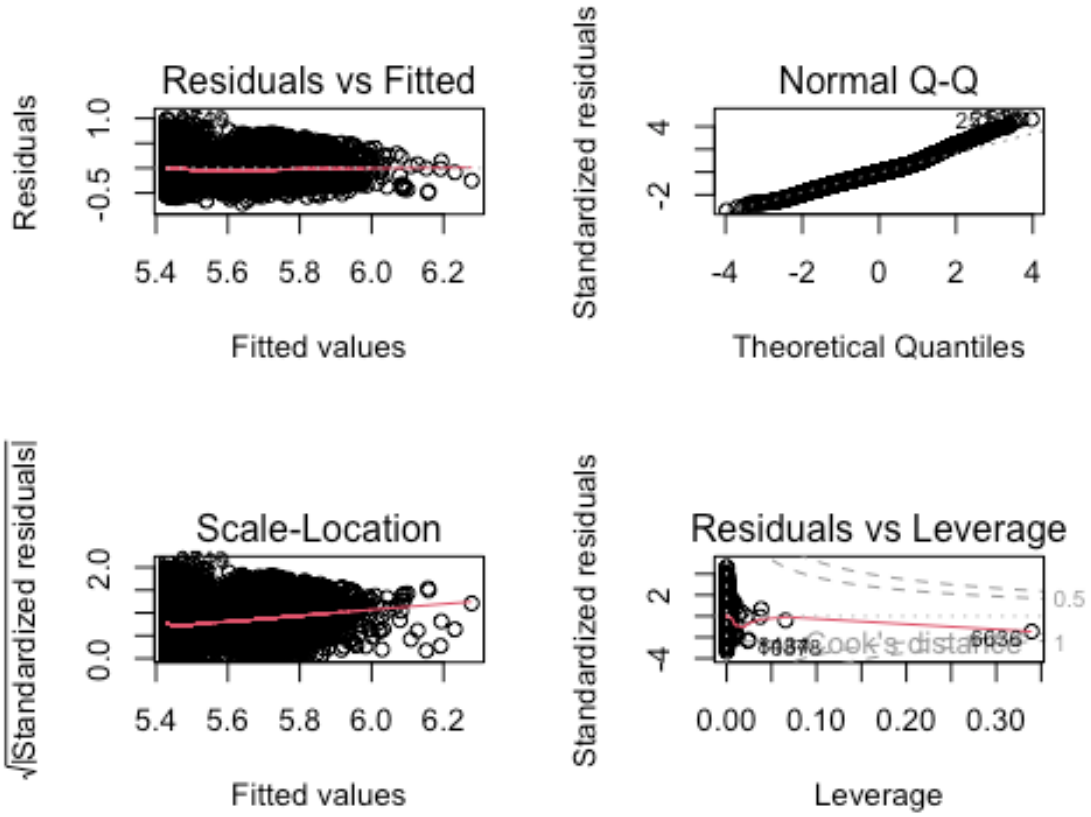
The residual plot does not show the random distribution of the residuals and other polynomial degrees should be checked. Let's check the polynomial degree 2 for the variable SPEC_FEAT_VAL.

```
# log(price) vs SPEC_FEAT_VAL + SPEC_FEAT_VAL2
model.SPEC_FEAT_VAL2 <- lm(data=Miami_house, log10_SALE_PRC ~ SPEC_FEAT_VAL +
I(SPEC_FEAT_VAL**2))
summary(model.SPEC_FEAT_VAL2)

##
## Call:
## lm(formula = log10_SALE_PRC ~ SPEC_FEAT_VAL + I(SPEC_FEAT_VAL^2),
##     data = Miami_house)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.7149 -0.1381 -0.0126  0.1092  0.9853
##
## Coefficients:
##                       Estimate Std. Error  t value Pr(>|t|)
## (Intercept)         5.430e+00  2.330e-03 2330.253  < 2e-16 ***
```

```
## SPEC_FEAT_VAL       1.043e-05  2.758e-07   37.817  < 2e-16 ***
## I(SPEC_FEAT_VAL^2) -3.188e-11  5.345e-12   -5.964 2.52e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2125 on 13929 degrees of freedom
## Multiple R-squared:  0.2576, Adjusted R-squared:  0.2575
## F-statistic:  2417 on 2 and 13929 DF,  p-value: < 2.2e-16

BIC(model.SPEC_FEAT_VAL2)

## [1] -3585.42

# Plotting model #
ggplot(Miami_house, aes(x = SPEC_FEAT_VAL, y = log10_SALE_PRC)) +
  geom_point() +
  stat_smooth(method = "lm", formula = y ~ poly(x,2)) +
  labs(x = 'Value of special features ($) + SPEC_FEAT_VAL^2', y = 'Log(Price)
($)')
```




```
# Diagnostic
par(mfrow=c(2,2))
plot(model.SPEC_FEAT_VAL2)
```



These plots show that adding degrees improves the interpolation of the data. But to have more proof of this fact the ANOVA function has been used.

```
par(mfrow=c(1,1))

anova(model.SPEC_FEAT_VAL, model.SPEC_FEAT_VAL2)

## Analysis of Variance Table
##
## Model 1: log10_SALE_PRC ~ SPEC_FEAT_VAL
## Model 2: log10_SALE_PRC ~ SPEC_FEAT_VAL + I(SPEC_FEAT_VAL^2)
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1  13930 630.51
## 2  13929 628.90  1     1.606 35.569 2.521e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

It can be seen that with having such a small P-value, it's understandable that model including SPEC_FEAT_VAL^2 provides a significantly better fit.

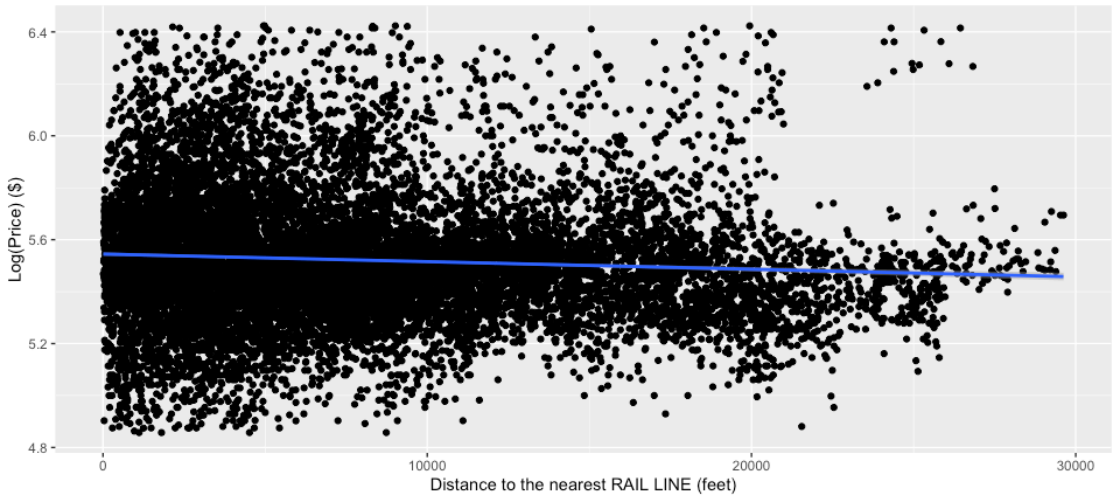
```
# log(price) vs Distance to the nearest RAIL LINE variable #
model.RAIL_DIST <- lm(data=Miami_house, log10_SALE_PRC ~ RAIL_DIST)
summary(model.RAIL_DIST)
```

```
##
## Call:
## lm(formula = log10_SALE_PRC ~ RAIL_DIST, data = Miami_house)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.68414 -0.14794 -0.02354  0.11006  0.94757
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.545e+00  3.503e-03 1582.88   <2e-16 ***
## RAIL_DIST   -2.924e-06  3.373e-07   -8.67   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2459 on 13930 degrees of freedom
## Multiple R-squared:  0.005367,   Adjusted R-squared:  0.005295
## F-statistic: 75.16 on 1 and 13930 DF,  p-value: < 2.2e-16
```

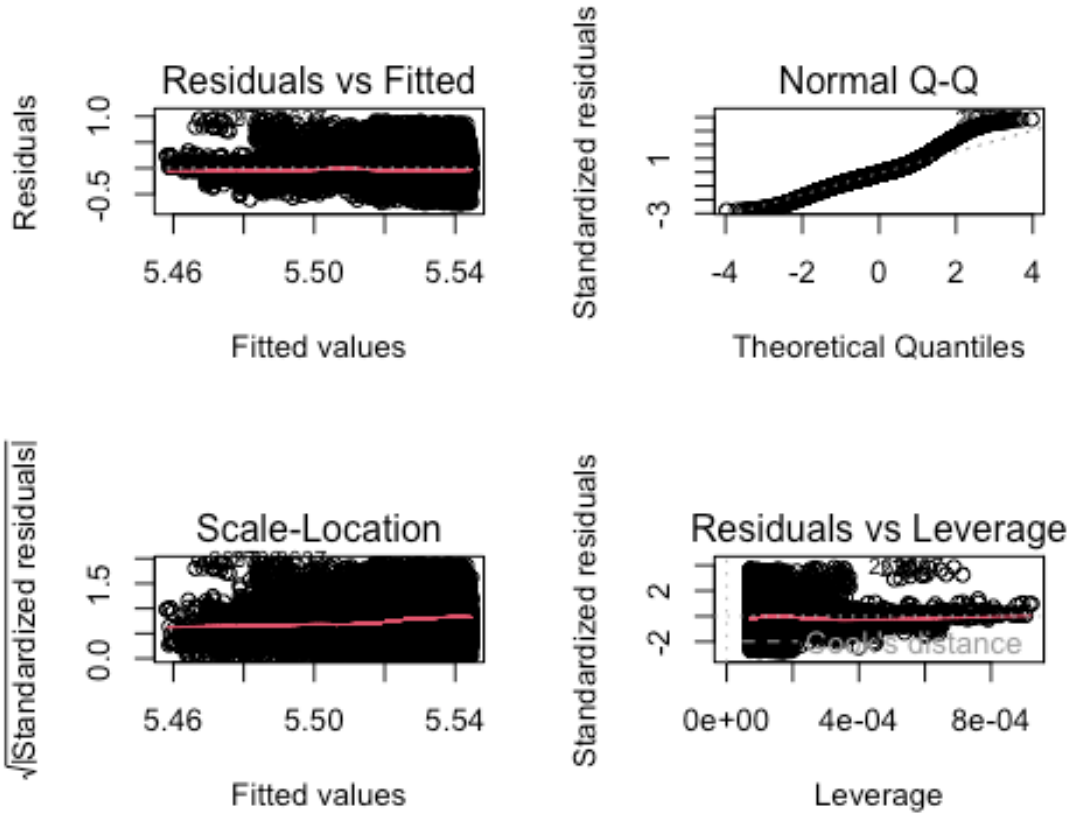
```
BIC(model.RAIL_DIST)
```

```
## [1] 480.4123
```

```
# Plotting the model #
ggplot(Miami_house, aes(x = RAIL_DIST, y = log10_SALE_PRC)) + geom_point() +
stat_smooth(method = "lm", formula = y ~ x) +
labs(x = 'Distance to the nearest RAIL LINE (feet)', y = 'Log(Price) ($)')
```



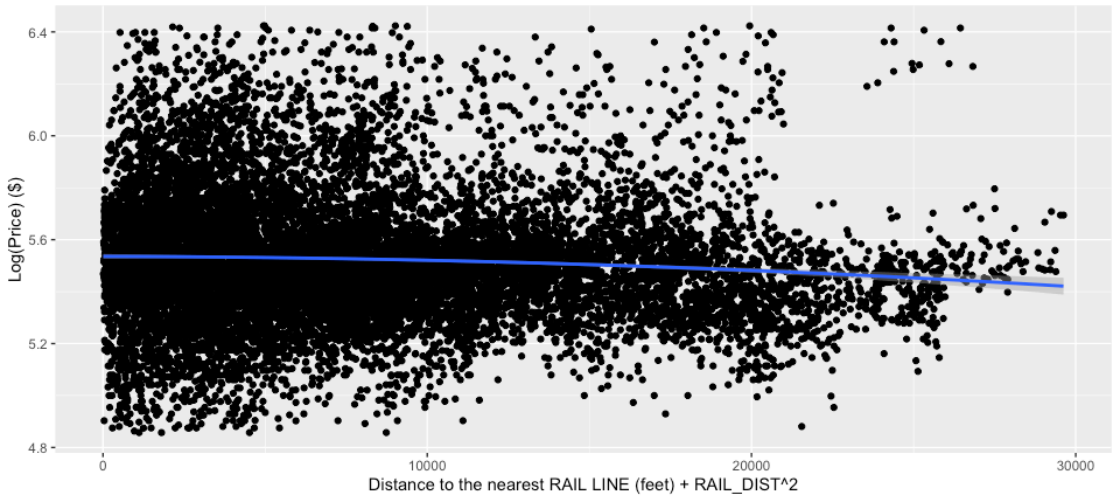

```
# Diagnostic
par(mfrow=c(2,2))
plot(model.RAIL_DIST)
```



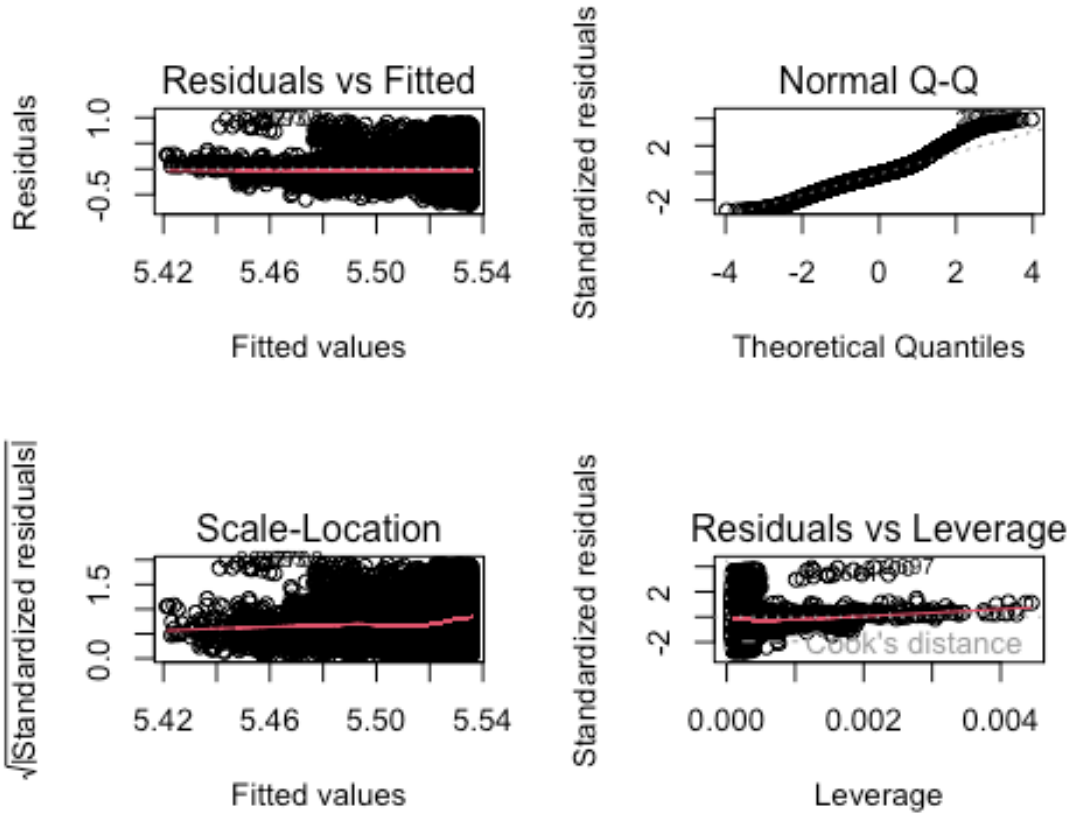
For the variable RAIL_DIST, the higher degree polynomial has been tested.

```
par(mfrow=c(1,1))

model.RAIL_DIST2 <- lm(data=Miami_house, log10_SALE_PRC ~ RAIL_DIST + I(RAIL_
DIST**2))
summary(model.RAIL_DIST2)

##
## Call:
## lm(formula = log10_SALE_PRC ~ RAIL_DIST + I(RAIL_DIST^2), data = Miami_hou
se)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.67799 -0.14788 -0.02357  0.11020  0.97133
##
## Coefficients:
##                   Estimate Std. Error  t value Pr(>|t|)
## (Intercept)     5.536e+00  4.976e-03 1112.566   <2e-16 ***
## RAIL_DIST      -2.984e-07  1.093e-06   -0.273   0.7849
## I(RAIL_DIST^2) -1.205e-10  4.774e-11   -2.525   0.0116 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 0.2459 on 13929 degrees of freedom
## Multiple R-squared:  0.005822,   Adjusted R-squared:  0.005679
## F-statistic: 40.78 on 2 and 13929 DF,  p-value: < 2.2e-16

BIC(model.RAIL_DIST2)

## [1] 483.5809

# Plotting model #
ggplot(Miami_house, aes(x = RAIL_DIST, y = log10_SALE_PRC)) +
  geom_point() +
  stat_smooth(method = "lm", formula = y ~ poly(x,2)) +
  labs(x = 'Distance to the nearest RAIL LINE (feet) + RAIL_DIST^2', y = 'Log
(Price) ($)')
```




```
# Diagnostic
par(mfrow=c(2,2))
plot(model.RAIL_DIST2)
```




```
par(mfrow=c(1,1))

anova(model.RAIL_DIST, model.RAIL_DIST2)

## Analysis of Variance Table
##
## Model 1: log10_SALE_PRC ~ RAIL_DIST
## Model 2: log10_SALE_PRC ~ RAIL_DIST + I(RAIL_DIST^2)
##   Res.Df    RSS Df Sum of Sq      F Pr(>F)
## 1  13930 842.60
## 2  13929 842.22  1   0.38537 6.3734 0.0116 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

For variable RAIL_DIST a polynomial of grade 2 does not improve the fit so much. Moreover, comparing the residuals plots, the first one seems to be better. The ANOVA function shows no proof that adding higher polynomial regression helps the model to fit the data.

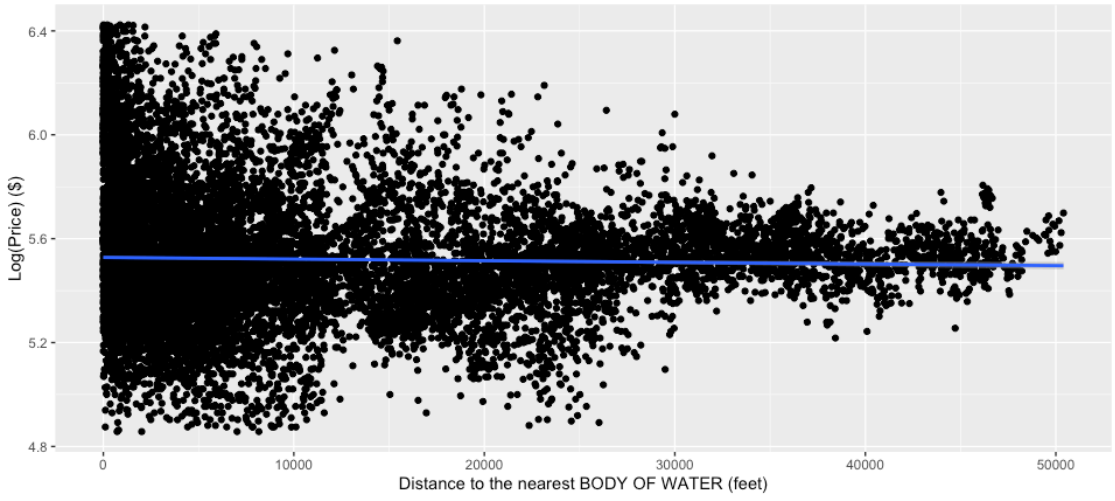
```
# log(price) vs Distance to the nearest BODY OF WATER variable #
model.WATER_DIST <- lm(data=Miami_house, log10_SALE_PRC ~ WATER_DIST)
summary(model.WATER_DIST)
```

```
##
## Call:
## lm(formula = log10_SALE_PRC ~ WATER_DIST, data = Miami_house)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.67006 -0.15346 -0.02453  0.11584  0.89623
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.528e+00  2.957e-03  1869.6  < 2e-16 ***
## WATER_DIST  -6.300e-07  1.750e-07    -3.6 0.000319 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2465 on 13930 degrees of freedom
## Multiple R-squared:  0.0009295,  Adjusted R-squared:  0.0008578
## F-statistic: 12.96 on 1 and 13930 DF,  p-value: 0.0003194
```

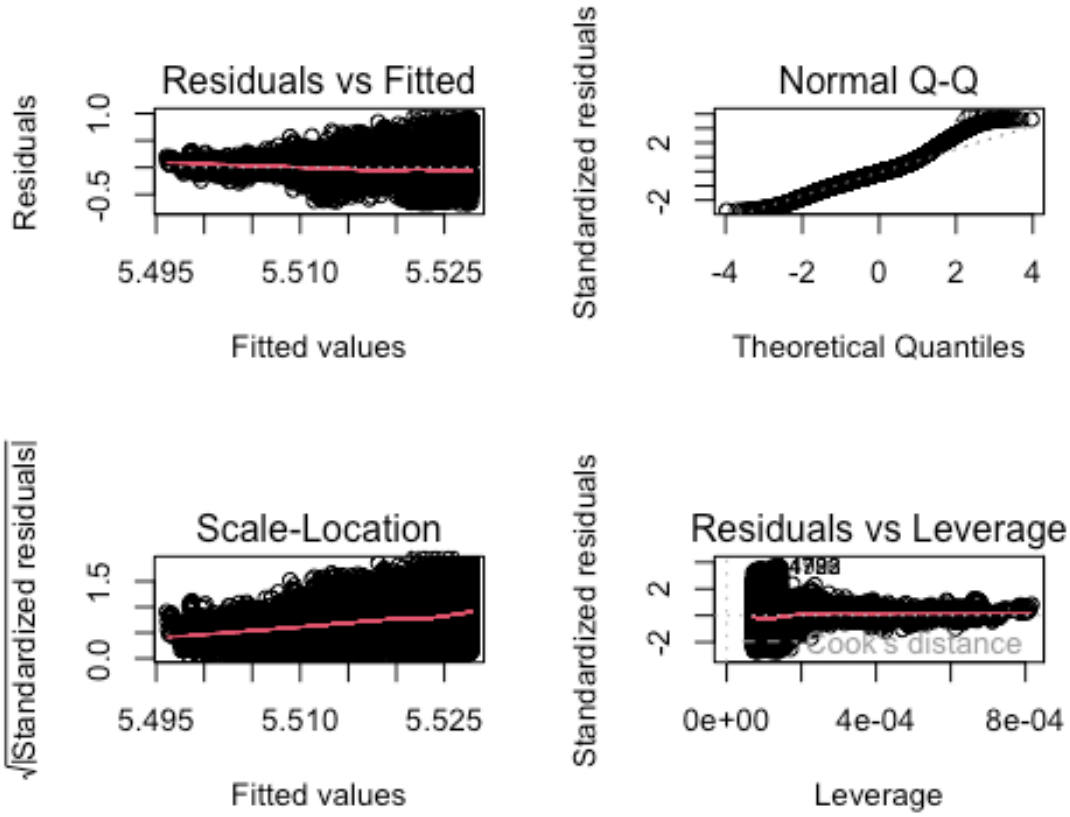
```
BIC(model.WATER_DIST)
```

```
## [1] 542.4287
```

```
# Plotting mode #
ggplot(Miami_house, aes(x = WATER_DIST, y = log10_SALE_PRC)) + geom_point() +
stat_smooth(method = "lm", formula = y ~ x) +
labs(x = 'Distance to the nearest BODY OF WATER (feet)', y = 'Log(Price) ($)'
)
```




```
# Diagnostic
par(mfrow=c(2,2))
plot(model.WATER_DIST)
```

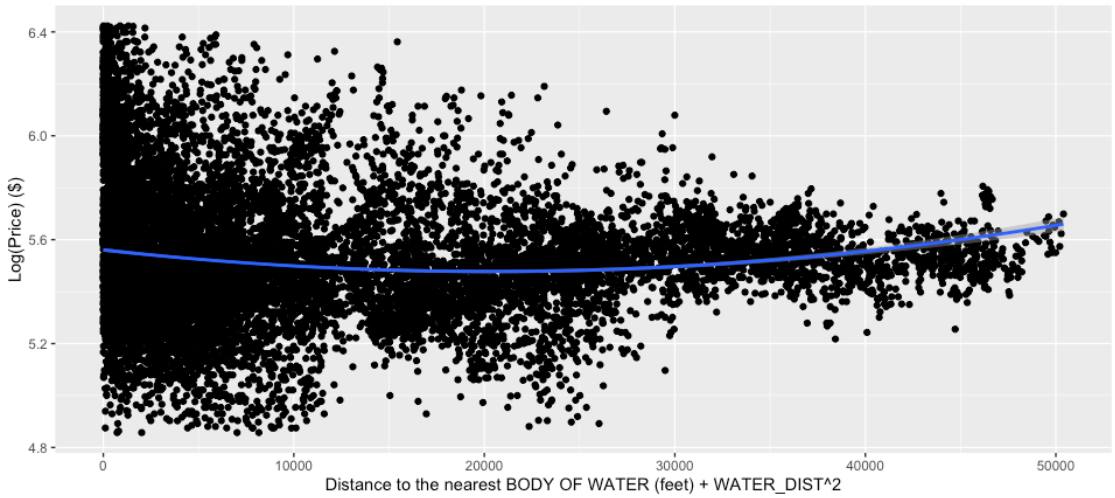



```
par(mfrow=c(1,1))
```

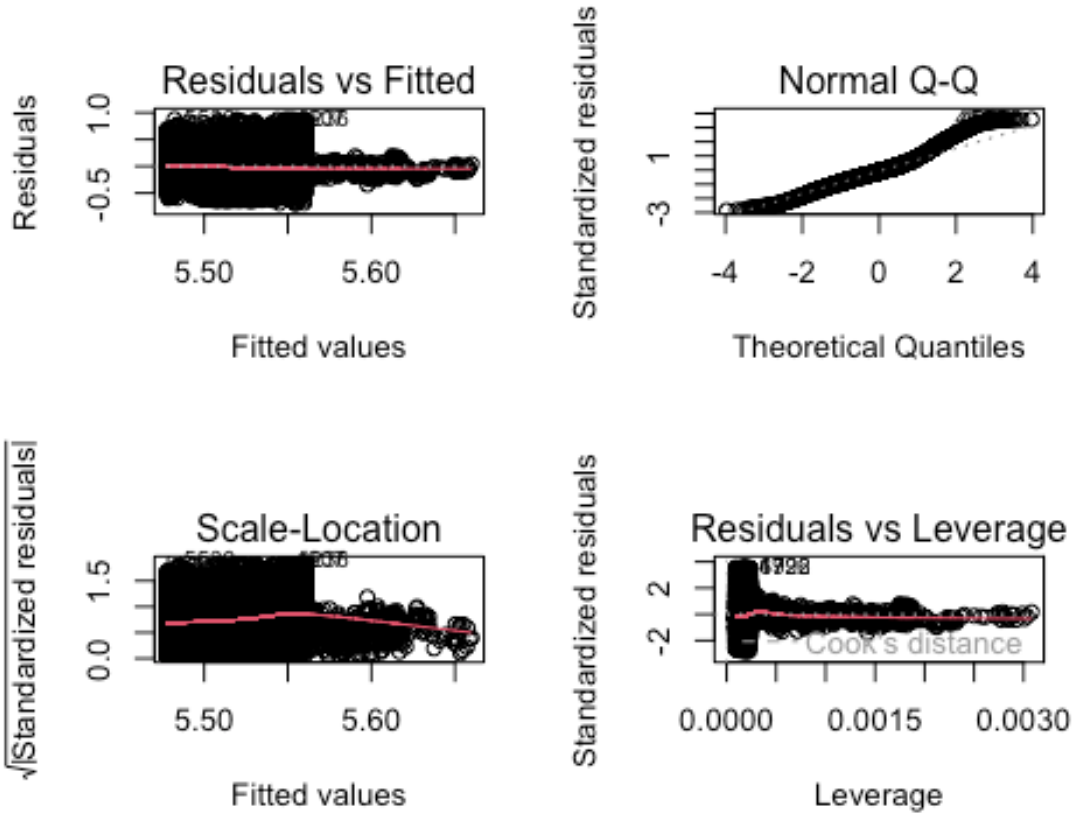
It seems that there is a relationship more complex than a simple linear model. So, the 2nd polynomial degree has been tested and the ANOVA function has been used.

```
# log(price) vs WATER_DIST + WATER_DIST2
model.WATER_DIST2 <- lm(data=Miami_house, log10_SALE_PRC ~ WATER_DIST + I(WAT
ER_DIST**2))
summary(model.WATER_DIST2)

##
## Call:
## lm(formula = log10_SALE_PRC ~ WATER_DIST + I(WATER_DIST^2), data = Miami_h
ouse)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.69756 -0.15122 -0.02534  0.11418  0.87937
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      5.561e+00  3.752e-03 1482.08   <2e-16 ***
## WATER_DIST      -8.188e-06  5.646e-07  -14.50   <2e-16 ***
## I(WATER_DIST^2)  2.015e-10  1.433e-11   14.07   <2e-16 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2448 on 13929 degrees of freedom
## Multiple R-squared:  0.01493,    Adjusted R-squared:  0.01479
## F-statistic: 105.5 on 2 and 13929 DF,  p-value: < 2.2e-16

BIC(model.WATER_DIST2)

## [1] 355.3946

ggplot(Miami_house, aes(x = WATER_DIST, y = log10_SALE_PRC)) + geom_point() +
stat_smooth(method = "lm", formula = y ~ poly(x,2)) +
labs(x = 'Distance to the nearest BODY OF WATER (feet) + WATER_DIST^2',
y = 'Log(Price) ($)')
```




```
# Diagnostic
par(mfrow=c(2,2))
plot(model.WATER_DIST2)
```




```
par(mfrow=c(1,1))

anova(model.WATER_DIST, model.WATER_DIST2)

## Analysis of Variance Table
##
## Model 1: log10_SALE_PRC ~ WATER_DIST
## Model 2: log10_SALE_PRC ~ WATER_DIST + I(WATER_DIST^2)
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1  13930 846.36
## 2  13929 834.50  1    11.858 197.93 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Adding a higher degree of the polynomial is helping the model to interpolate the data. A small P-value in the ANOVA function shows that model including WATER_DIST^2 fits better than the model without. But the residuals plot does not show a random distribution of the residuals, therefor the 3rd degree polynomial for the variable WATER_DIST has been checked.

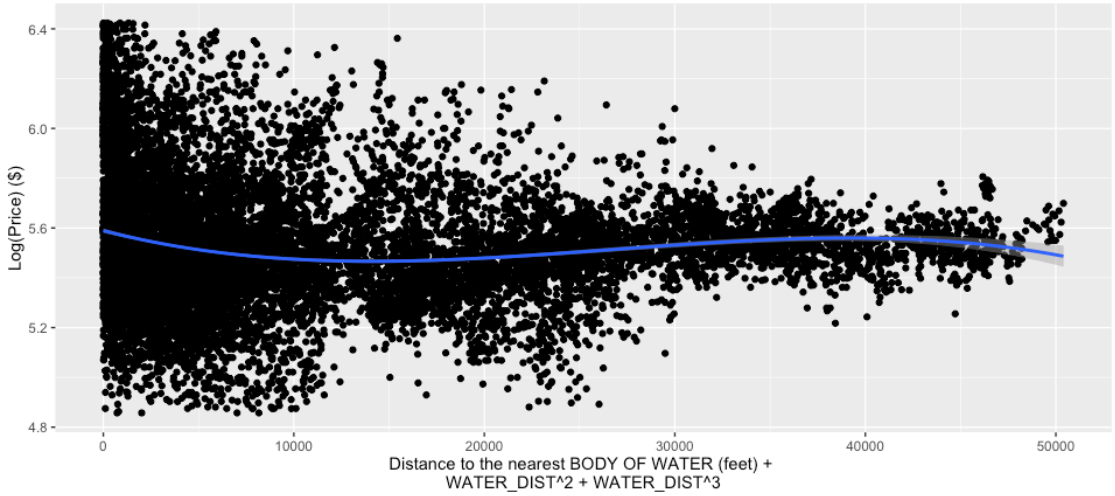
```
# log(price) vs WATER_DIST + WATER_DIST2 + WATER_DIST3
model.WATER_DIST3 <- lm(data=Miami_house, log10_SALE_PRC ~ WATER_DIST +
I(WATER_DIST**2) + I(WATER_DIST**3))
summary(model.WATER_DIST3)
```

```
##
## Call:
## lm(formula = log10_SALE_PRC ~ WATER_DIST + I(WATER_DIST^2) +
##     I(WATER_DIST^3), data = Miami_house)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.71822 -0.14941 -0.02415  0.11066  0.89467
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)      5.589e+00  4.631e-03 1206.92   <2e-16 ***
## WATER_DIST      -1.979e-05  1.242e-06  -15.93   <2e-16 ***
## I(WATER_DIST^2)  9.523e-10  7.308e-11   13.03   <2e-16 ***
## I(WATER_DIST^3) -1.191e-14  1.137e-15  -10.48   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2438 on 13928 degrees of freedom
## Multiple R-squared:  0.02263,    Adjusted R-squared:  0.02242
## F-statistic: 107.5 on 3 and 13928 DF,  p-value: < 2.2e-16
```

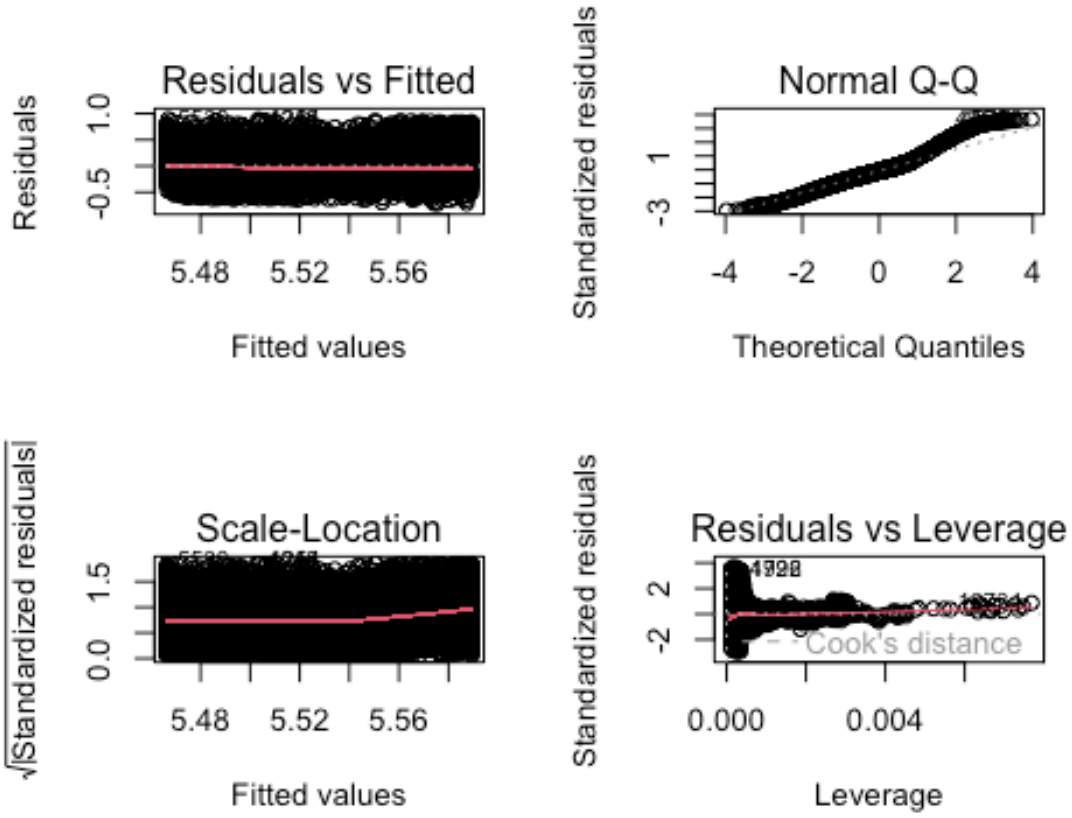
```
BIC(model.WATER_DIST3)
```

```
## [1] 255.5919
```

```
ggplot(Miami_house, aes(x = WATER_DIST, y = log10_SALE_PRC)) +
geom_point() + stat_smooth(method = "lm", formula = y ~ poly(x,3)) +
labs(x = 'Distance to the nearest BODY OF WATER (feet) +
WATER_DIST^2 + WATER_DIST^3',
y = 'Log(Price) ($)')
```




```
# Diagnostic
par(mfrow=c(2,2))
plot(model.WATER_DIST3)
```




```
par(mfrow=c(1,1))

anova(model.WATER_DIST2, model.WATER_DIST3)

## Analysis of Variance Table
##
## Model 1: log10_SALE_PRC ~ WATER_DIST + I(WATER_DIST^2)
## Model 2: log10_SALE_PRC ~ WATER_DIST + I(WATER_DIST^2) + I(WATER_DIST^3)
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1  13929 834.50
## 2  13928 827.98  1    6.5239 109.74 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Adding a 3rd-degree polynomial is also helping the model interpolate the data. From the ANOVA function with a small P-value WATER_DIST^3 has been chosen. Also, the residuals plot shows a random distribution of the residuals, since adding more polynomial terms for the model including this variable has been helpful the 4th degree also has been checked.

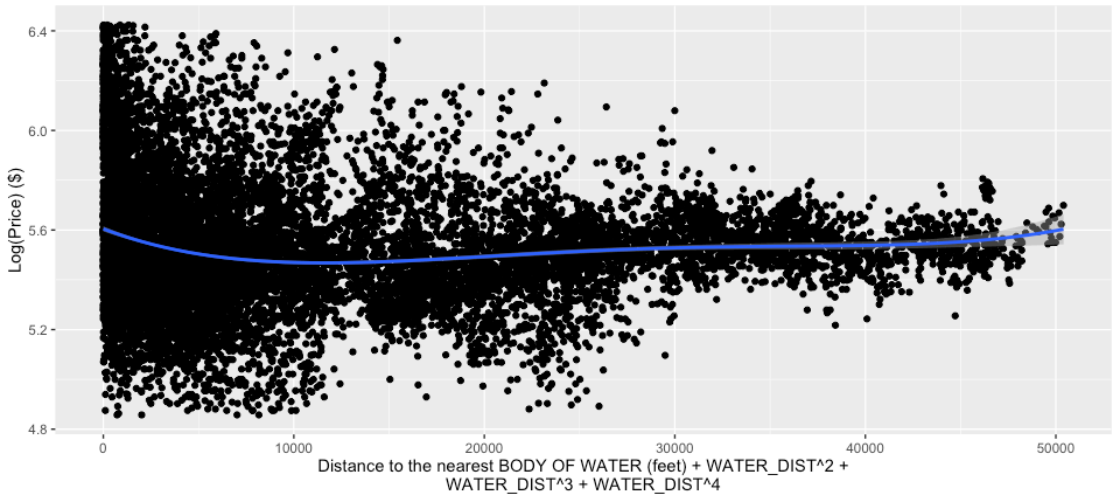
```
# log(price) vs WATER_DIST + WATER_DIST2 + WATER_DIST3 + WATER_DIST4
model.WATER_DIST4 <- lm(data=Miami_house, log10_SALE_PRC ~ WATER_DIST +
I(WATER_DIST**2) + I(WATER_DIST**3) + I(WATER_DIST**4))
summary(model.WATER_DIST4)
```

```
##
## Call:
## lm(formula = log10_SALE_PRC ~ WATER_DIST + I(WATER_DIST^2) +
##     I(WATER_DIST^3) + I(WATER_DIST^4), data = Miami_house)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.72721 -0.14831 -0.02422  0.11099  0.89577
##
## Coefficients:
##                    Estimate Std. Error  t value Pr(>|t|)
## (Intercept)      5.605e+00  5.456e-03 1027.182  < 2e-16 ***
## WATER_DIST      -2.913e-05  2.169e-06  -13.433  < 2e-16 ***
## I(WATER_DIST^2)  2.040e-09  2.196e-10    9.293  < 2e-16 ***
## I(WATER_DIST^3) -5.237e-14  7.783e-15   -6.729 1.78e-11 ***
## I(WATER_DIST^4)  4.631e-19  8.812e-20    5.255 1.50e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2436 on 13927 degrees of freedom
## Multiple R-squared:  0.02456,    Adjusted R-squared:  0.02428
## F-statistic: 87.67 on 4 and 13927 DF,  p-value: < 2.2e-16
```

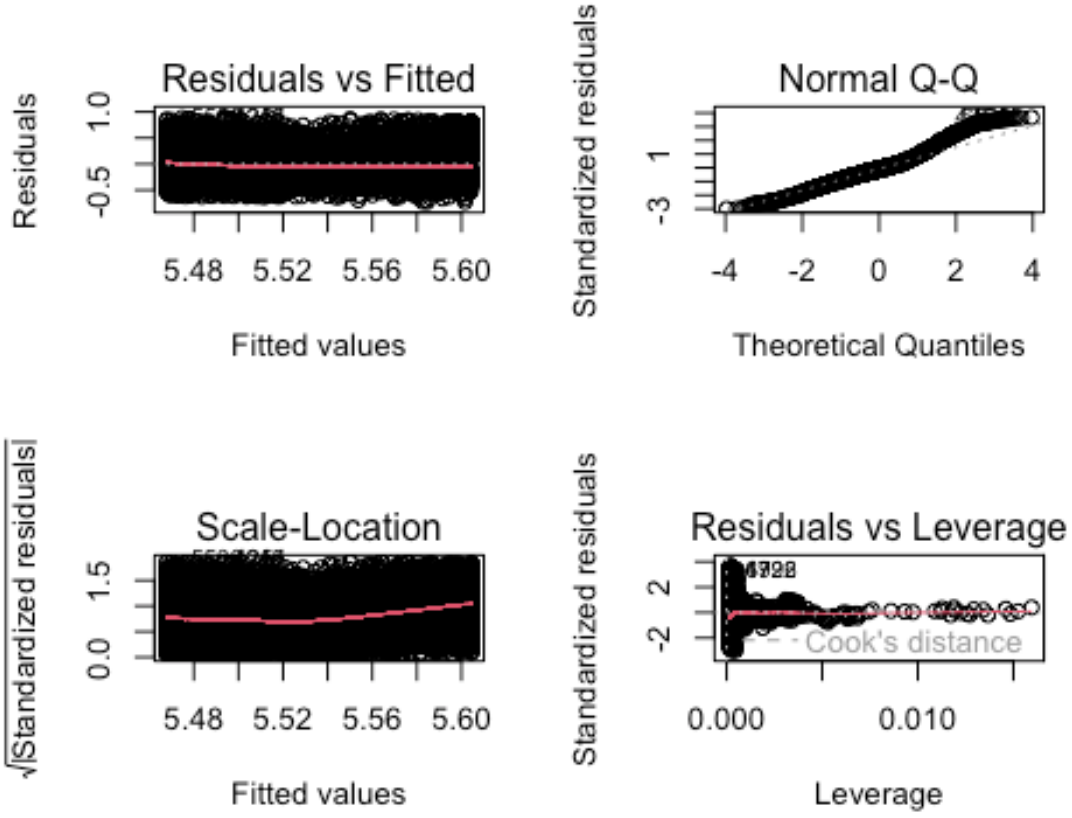
```
BIC(model.WATER_DIST4)
```

```
## [1] 237.5381
```

```
# Plotting model #
ggplot(Miami_house, aes(x = WATER_DIST, y = log10_SALE_PRC)) + geom_point() +
stat_smooth(method = "lm", formula = y ~ poly(x, 4)) +
labs(x = 'Distance to the nearest BODY OF WATER (feet) + WATER_DIST^2 +
WATER_DIST^3 + WATER_DIST^4',
y = 'Log(Price) ($)')
```



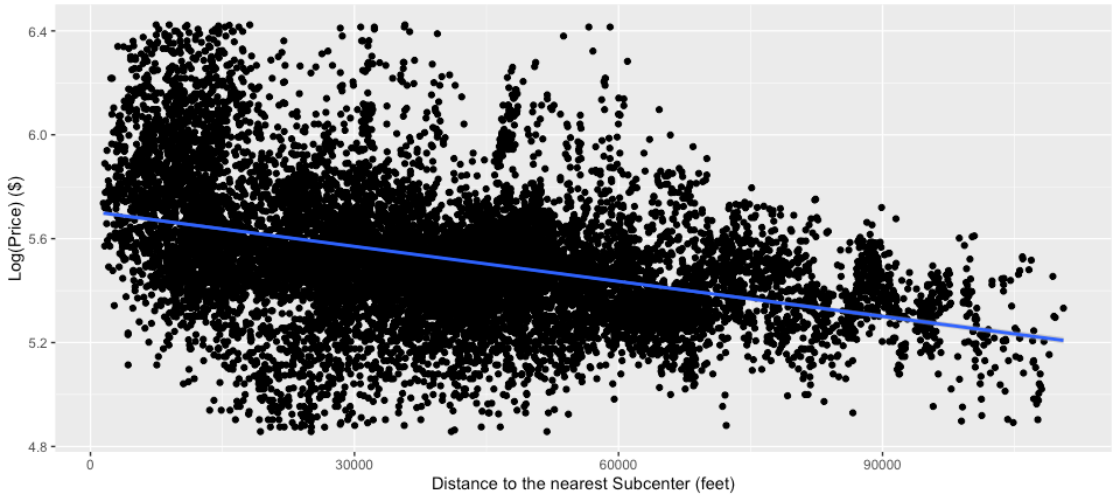
```
# Diagnostic
par(mfrow=c(2,2))
plot(model.WATER_DIST4)
```



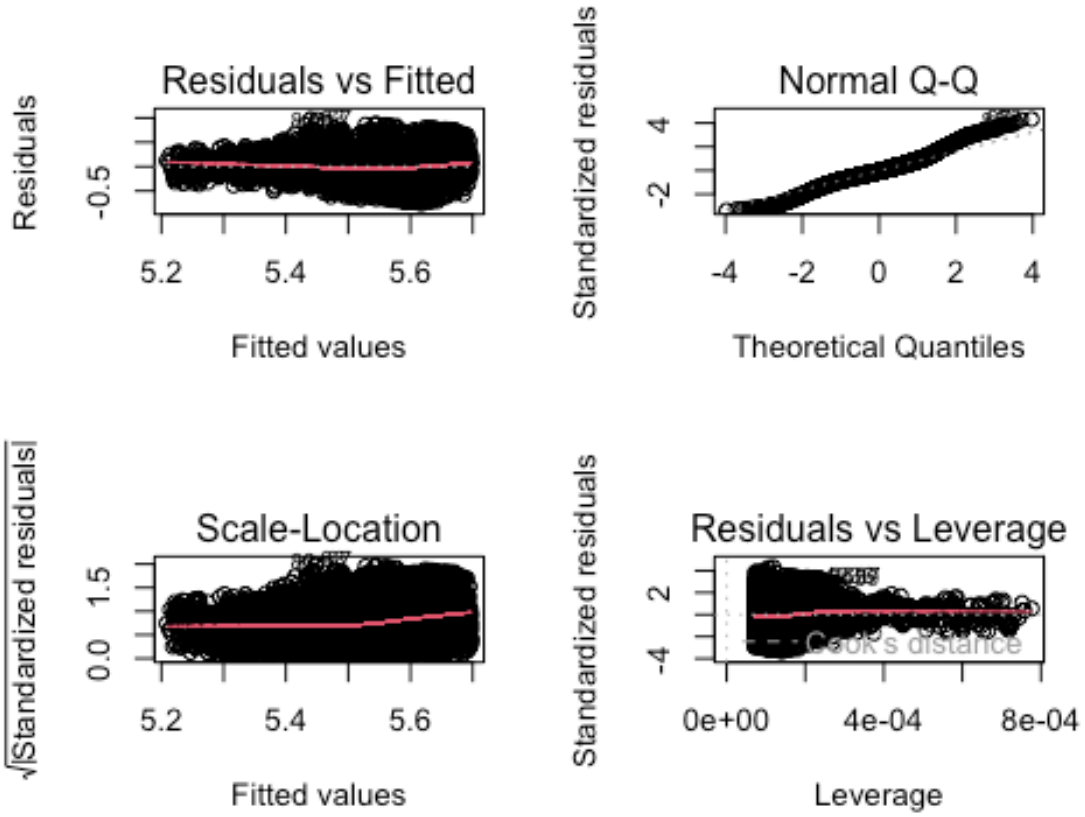

```
par(mfrow=c(1,1))

anova(model.WATER_DIST3, model.WATER_DIST4)

## Analysis of Variance Table
##
## Model 1: log10_SALE_PRC ~ WATER_DIST + I(WATER_DIST^2) + I(WATER_DIST^3)
## Model 2: log10_SALE_PRC ~ WATER_DIST + I(WATER_DIST^2) + I(WATER_DIST^3) +
##     I(WATER_DIST^4)
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1  13928 827.98
## 2  13927 826.34  1    1.6384 27.613 1.504e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

It can be seen that with having such a small P-value, the fourth-degree polynomial model has a better fit than the other models. Now even the residuals plot shows a better distribution.

```r
# log(price) vs Distance to the nearest SUBCENTER variable #
model.SUBCNTR_DI <- lm(data=Miami_house, log10_SALE_PRC ~ SUBCNTR_DI)
summary(model.SUBCNTR_DI)

##
## Call:
## lm(formula = log10_SALE_PRC ~ SUBCNTR_DI, data = Miami_house)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.76087 -0.12979 -0.01617  0.11820  0.97517
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.705e+00  4.029e-03 1416.02   <2e-16 ***
## SUBCNTR_DI  -4.491e-06  8.626e-08  -52.06   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2256 on 13930 degrees of freedom
## Multiple R-squared:  0.1629, Adjusted R-squared:  0.1628
## F-statistic:  2710 on 1 and 13930 DF,  p-value: < 2.2e-16

BIC(model.SUBCNTR_DI)

## [1] -1921.641

ggplot(Miami_house, aes(x = SUBCNTR_DI, y = log10_SALE_PRC)) +
  geom_point() + stat_smooth(method = "lm", formula = y ~ x) +
  labs(x = 'Distance to the nearest Subcenter (feet)', y = 'Log(Price) ($)')
```




```r
# Diagnostic
par(mfrow=c(2,2))
plot(model.SUBCNTR_DI)
```

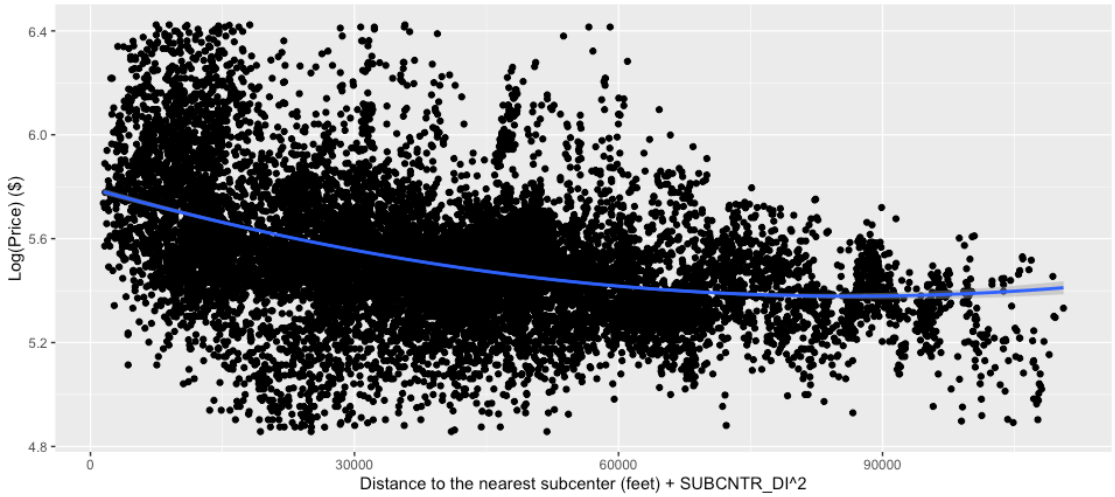



```
par(mfrow=c(1,1))
```

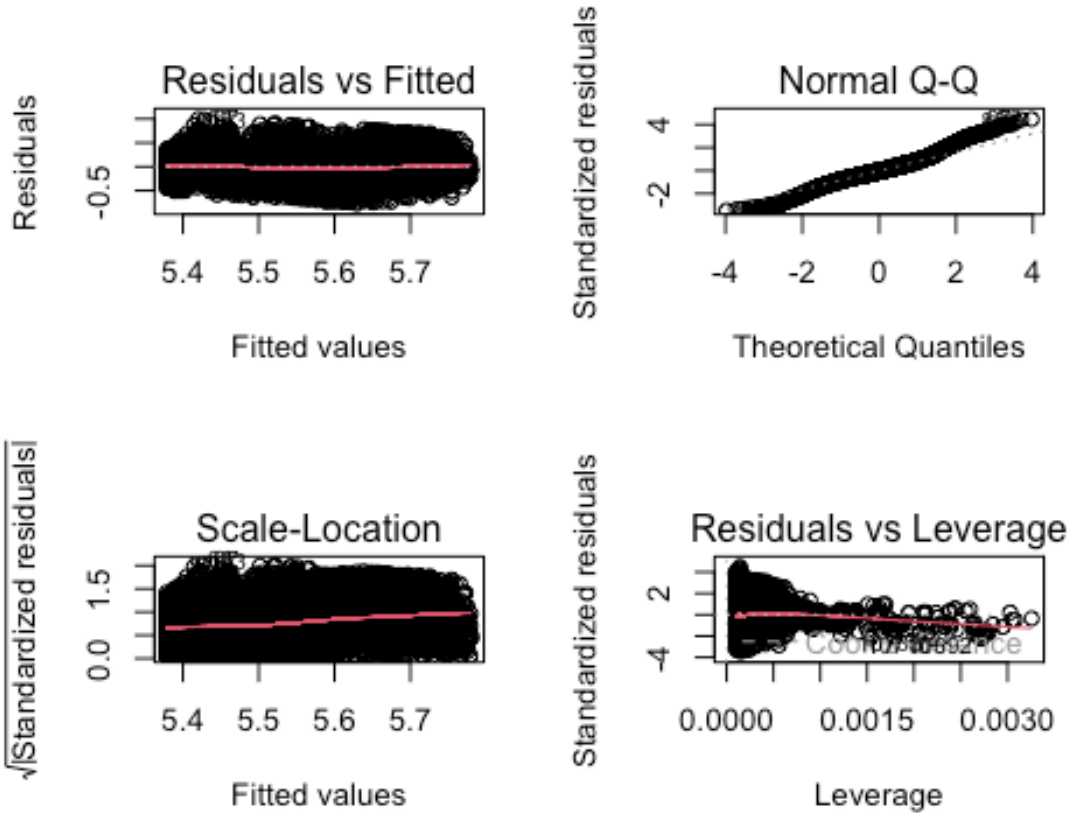
Regarding this variable also it seems that there is a relationship more complex than a simple linear model. So, the second polynomial degree has been tested and the ANOVA function has been used.

```
# log(price) vs SUBCNTR_DI2 #
model.SUBCNTR_DI2 <- lm(data=Miami_house, log10_SALE_PRC ~ SUBCNTR_DI + I(SUB
CNTR_DI**2))
summary(model.SUBCNTR_DI2)

##
## Call:
## lm(formula = log10_SALE_PRC ~ SUBCNTR_DI + I(SUBCNTR_DI^2), data = Miami_h
ouse)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.77222 -0.12695 -0.01088  0.11645  0.99471
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      5.795e+00  6.340e-03  914.09   <2e-16 ***
```

```
## SUBCNTR_DI      -9.640e-06  2.948e-07  -32.70   <2e-16 ***
## I(SUBCNTR_DI^2)  5.579e-11  3.058e-12   18.24   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.223 on 13929 degrees of freedom
## Multiple R-squared:  0.1824, Adjusted R-squared:  0.1823
## F-statistic:  1554 on 2 and 13929 DF,  p-value: < 2.2e-16

BIC(model.SUBCNTR_DI2)

## [1] -2241.112

ggplot(Miami_house, aes(x = SUBCNTR_DI, y = log10_SALE_PRC)) +
  geom_point() +
  stat_smooth(method = "lm", formula = y ~ poly(x, 2)) +
  labs(x = 'Distance to the nearest subcenter (feet) + SUBCNTR_DI^2',
  y = 'Log(Price) ($)')
```




```
# Diagnostic
par(mfrow=c(2,2))
plot(model.SUBCNTR_DI2)
```




```
anova(model.SUBCNTR_DI, model.SUBCNTR_DI2)
```

```
## Analysis of Variance Table
##
## Model 1: log10_SALE_PRC ~ SUBCNTR_DI
## Model 2: log10_SALE_PRC ~ SUBCNTR_DI + I(SUBCNTR_DI^2)
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1  13930 709.16
## 2  13929 692.61  1    16.551 332.86 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Adding a higher degree polynomial is helping the model to fit the data and from what the ANOVA function shows, it's obvious that the model including SUBCNTR_DI^2 fits better than the simple regression model. The residuals plot also shows a better random distribution of the residuals in comparison to the simple model, therefor the 3rd degree polynomial for the variable SUBCNTR_DI has been checked.

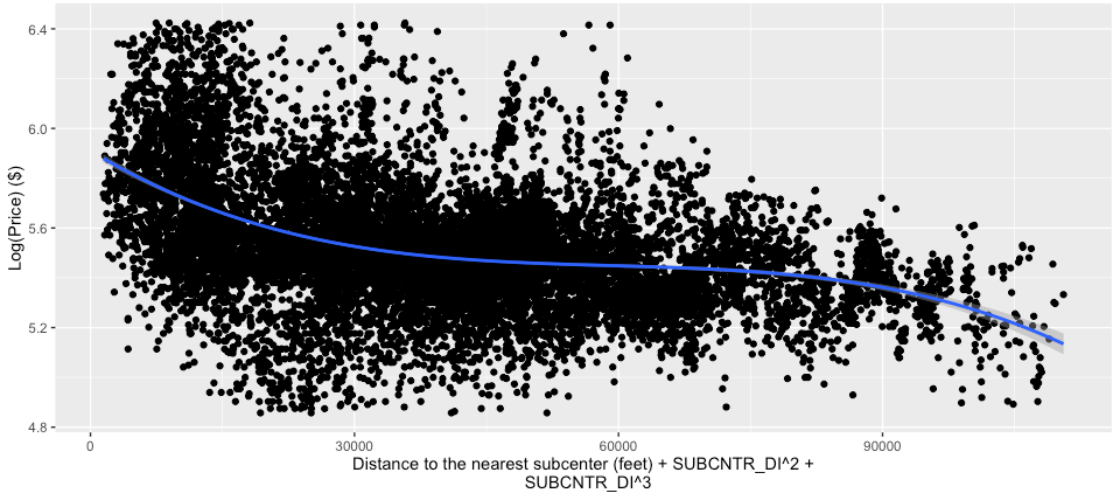
```
#log(price) vs SUBCNTR_DI2 + SUBCNTR_DI2 + SUBCNTR_DI3 #
model.SUBCNTR_DI3 <- lm(data=Miami_house, log10_SALE_PRC ~ SUBCNTR_DI +
I(SUBCNTR_DI**2) + I(SUBCNTR_DI**3))
summary(model.SUBCNTR_DI3)
```

```
##
## Call:
## lm(formula = log10_SALE_PRC ~ SUBCNTR_DI + I(SUBCNTR_DI^2) +
##     I(SUBCNTR_DI^3), data = Miami_house)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.75402 -0.12965 -0.00913  0.11408  0.96593
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)      5.909e+00  9.505e-03  621.74   <2e-16 ***
## SUBCNTR_DI      -2.136e-05  7.861e-07  -27.17   <2e-16 ***
## I(SUBCNTR_DI^2)  3.441e-10  1.821e-11   18.90   <2e-16 ***
## I(SUBCNTR_DI^3) -1.939e-15  1.207e-16  -16.06   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.221 on 13928 degrees of freedom
## Multiple R-squared:  0.1973, Adjusted R-squared:  0.1971
## F-statistic:  1141 on 3 and 13928 DF,  p-value: < 2.2e-16
```

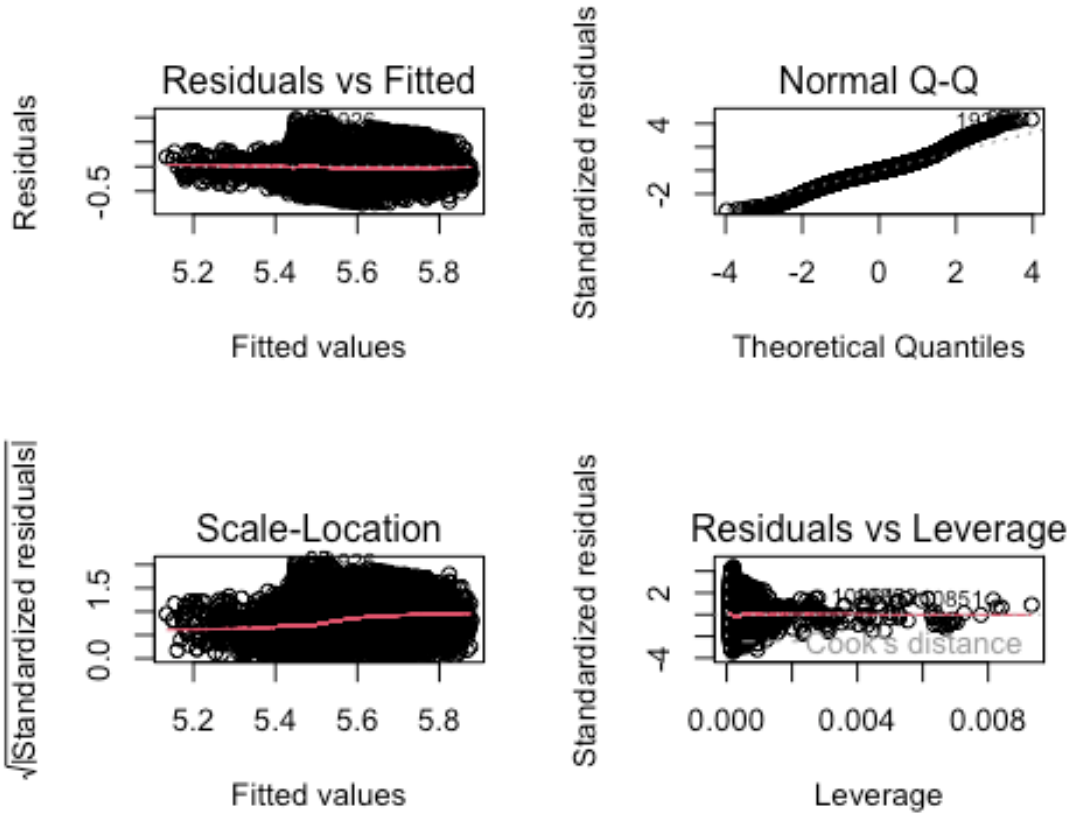
```
BIC(model.SUBCNTR_DI3)
```

```
## [1] -2487.086
```

```
ggplot(Miami_house, aes(x = SUBCNTR_DI, y = log10_SALE_PRC)) +
  geom_point() +
  stat_smooth(method = "lm", formula = y ~ poly(x, 3)) +
  labs(x = 'Distance to the nearest subcenter (feet) + SUBCNTR_DI^2 +
  SUBCNTR_DI^3', y = 'Log(Price) ($)')
```




```
# Diagnostic
par(mfrow=c(2,2))
plot(model.SUBCNTR_DI3)
```




```
par(mfrow=c(1,1))

anova(model.SUBCNTR_DI2, model.SUBCNTR_DI3)

## Analysis of Variance Table
##
## Model 1: log10_SALE_PRC ~ SUBCNTR_DI + I(SUBCNTR_DI^2)
## Model 2: log10_SALE_PRC ~ SUBCNTR_DI + I(SUBCNTR_DI^2) + I(SUBCNTR_DI^3)
##   Res.Df    RSS Df Sum of Sq     F    Pr(>F)
## 1  13929 692.61
## 2  13928 680.02  1    12.587 257.8 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Again there is an improvement in the model and the ANOVA function has been used to compare the models, and with a small P-value, the third-degree polynomial is selected. Adding polynomial degree 3 has a better fit on the data but looking at the residuals plot shows that there is not a good distribution of the residuals and also the red line in the scale-location plot is not flat. So, the 4th polynomial has been tested.

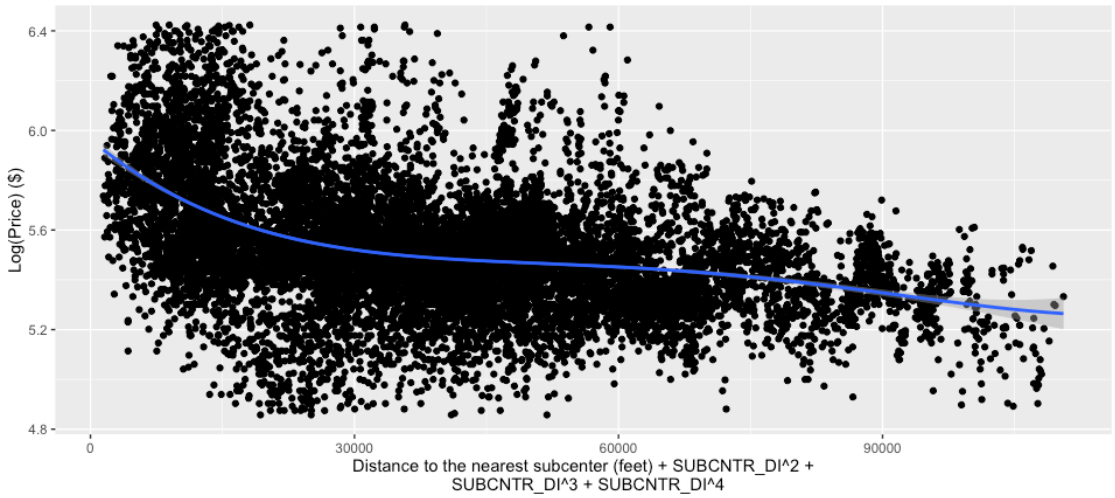
```
#log(price) vs SUBCNTR_DI2 + SUBCNTR_DI2 + SUBCNTR_DI3 + SUBCNTR_DI4#
model.SUBCNTR_DI4 <- lm(data=Miami_house, log10_SALE_PRC ~ SUBCNTR_DI +
I(SUBCNTR_DI**2) + I(SUBCNTR_DI**3) + I(SUBCNTR_DI**4))
summary(model.SUBCNTR_DI4)
```

```
##
## Call:
## lm(formula = log10_SALE_PRC ~ SUBCNTR_DI + I(SUBCNTR_DI^2) +
##     I(SUBCNTR_DI^3) + I(SUBCNTR_DI^4), data = Miami_house)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.74277 -0.12825 -0.00948  0.11347  0.96171
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)      5.964e+00  1.385e-02 430.530  < 2e-16 ***
## SUBCNTR_DI      -2.943e-05  1.686e-06 -17.457  < 2e-16 ***
## I(SUBCNTR_DI^2)  6.706e-10  6.302e-11  10.642  < 2e-16 ***
## I(SUBCNTR_DI^3) -6.798e-15  9.060e-16  -7.503 6.62e-14 ***
## I(SUBCNTR_DI^4)  2.372e-20  4.383e-21   5.411 6.35e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2207 on 13927 degrees of freedom
## Multiple R-squared:  0.199,  Adjusted R-squared:  0.1987
## F-statistic: 864.8 on 4 and 13927 DF,  p-value: < 2.2e-16
```

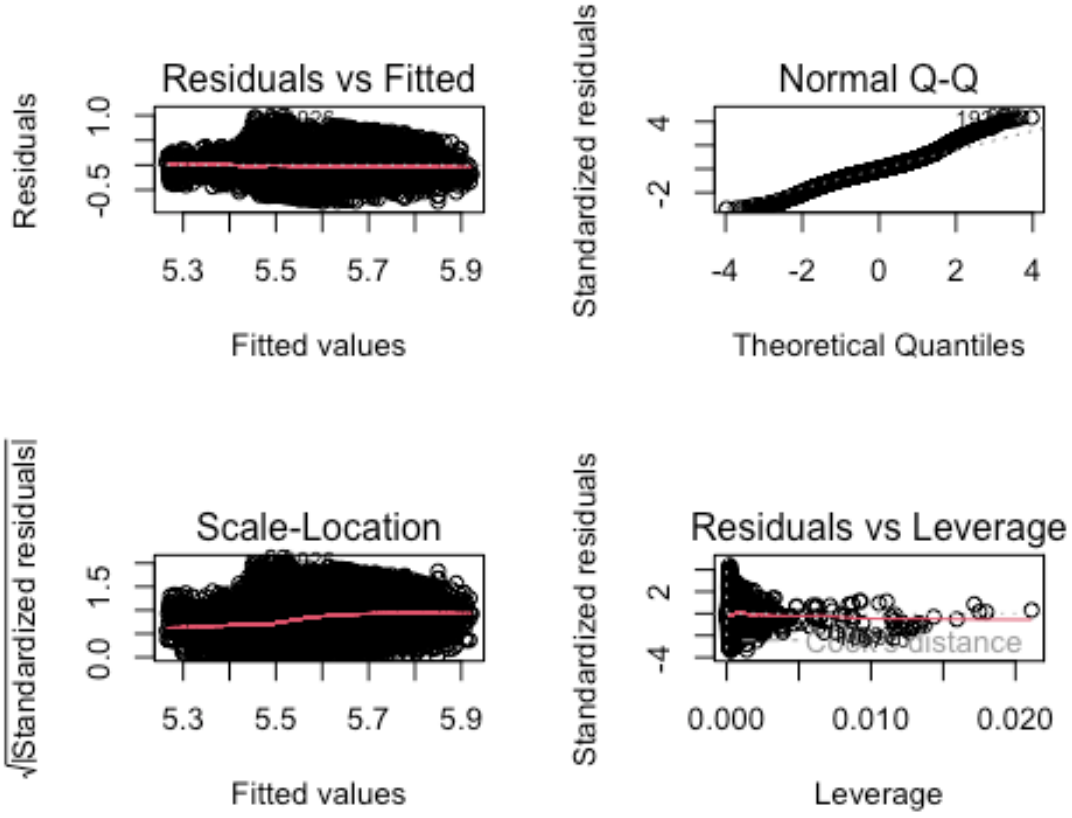
```
BIC(model.SUBCNTR_DI4)
```

```
## [1] -2506.808
```

```
# Plotting model #
ggplot(Miami_house, aes(x = SUBCNTR_DI, y = log10_SALE_PRC)) +
  geom_point() +
  stat_smooth(method = "lm", formula = y ~ poly(x, 4)) +
  labs(x = 'Distance to the nearest subcenter (feet) + SUBCNTR_DI^2 +
  SUBCNTR_DI^3 + SUBCNTR_DI^4', y = 'Log(Price) ($)')
```



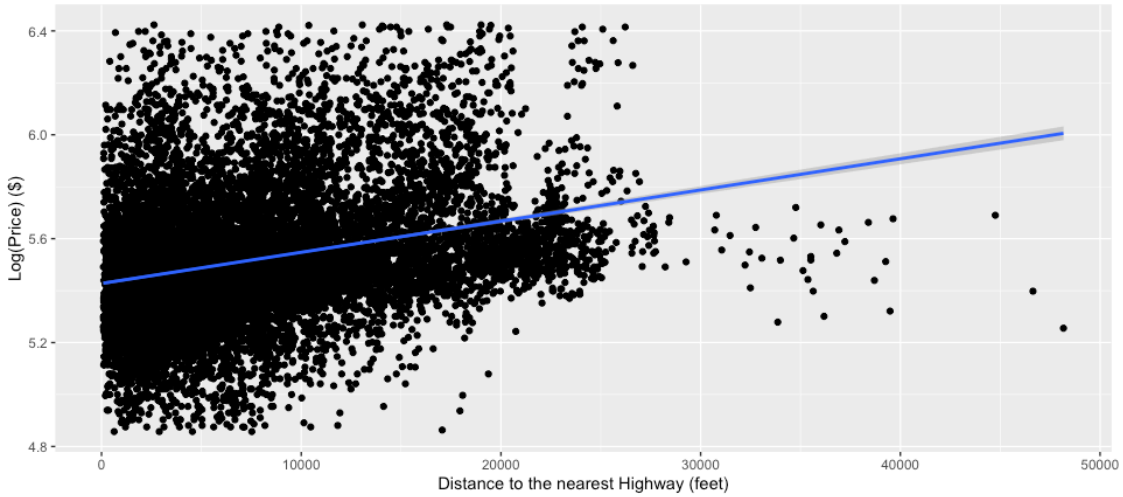
```
# Diagnostic
par(mfrow=c(2,2))
plot(model.SUBCNTR_DI4)
```



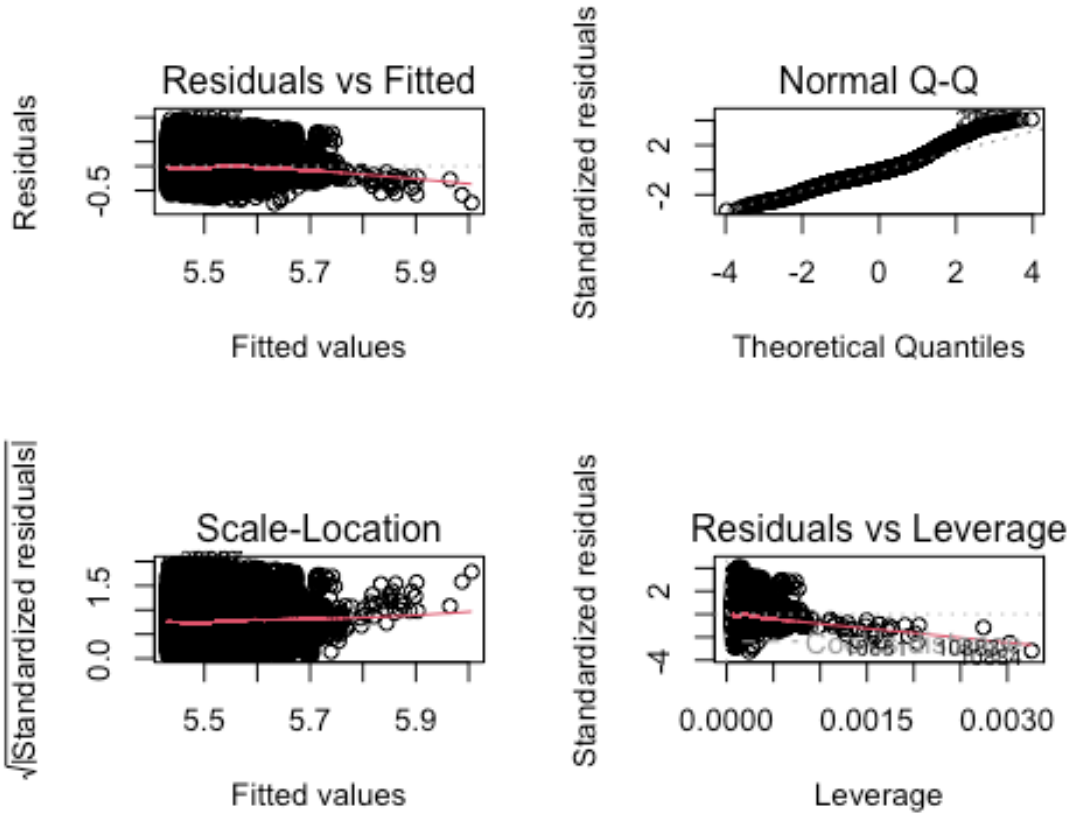

```
par(mfrow=c(1,1))

anova(model.SUBCNTR_DI3, model.SUBCNTR_DI4)

## Analysis of Variance Table
##
## Model 1: log10_SALE_PRC ~ SUBCNTR_DI + I(SUBCNTR_DI^2) + I(SUBCNTR_DI^3)
## Model 2: log10_SALE_PRC ~ SUBCNTR_DI + I(SUBCNTR_DI^2) + I(SUBCNTR_DI^3) +
##     I(SUBCNTR_DI^4)
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1  13928 680.02
## 2  13927 678.60  1    1.4269 29.284 6.354e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The 4th degree polynomial of variable SUBCNTR_DI fits better the data and the residuals plot shows the better distribution of the residuals and the scale-location plot also shows that the red line is flatter in comparison to the degree polynomial equal to 3. A small P-value in the ANOVA function shows SUBCNTR_DI^4 provides a better fit.

```
# log(price) vs Distance to the nearest HIGHWAY variable #
model.HWY_DIST <- lm(data=Miami_house, log10_SALE_PRC ~ HWY_DIST)
summary(model.HWY_DIST)

##
## Call:
## lm(formula = log10_SALE_PRC ~ HWY_DIST, data = Miami_house)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.76902 -0.14845 -0.03357  0.10620  0.96410
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 5.428e+00  3.231e-03 1679.94   <2e-16 ***
## HWY_DIST    1.200e-05  3.289e-07   36.48   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2356 on 13930 degrees of freedom
## Multiple R-squared:  0.08722,    Adjusted R-squared:  0.08716
## F-statistic:  1331 on 1 and 13930 DF,  p-value: < 2.2e-16

BIC(model.HWY_DIST)

## [1] -716.1222

# Plotting model #
ggplot(Miami_house, aes(x = HWY_DIST, y = log10_SALE_PRC)) +
geom_point() + stat_smooth(method = "lm", formula = y ~ x) +
labs(x = 'Distance to the nearest Highway (feet)', y = 'Log(Price) ($)')
```




```
# Diagnostic
par(mfrow=c(2,2))
plot(model.HWY_DIST)
```

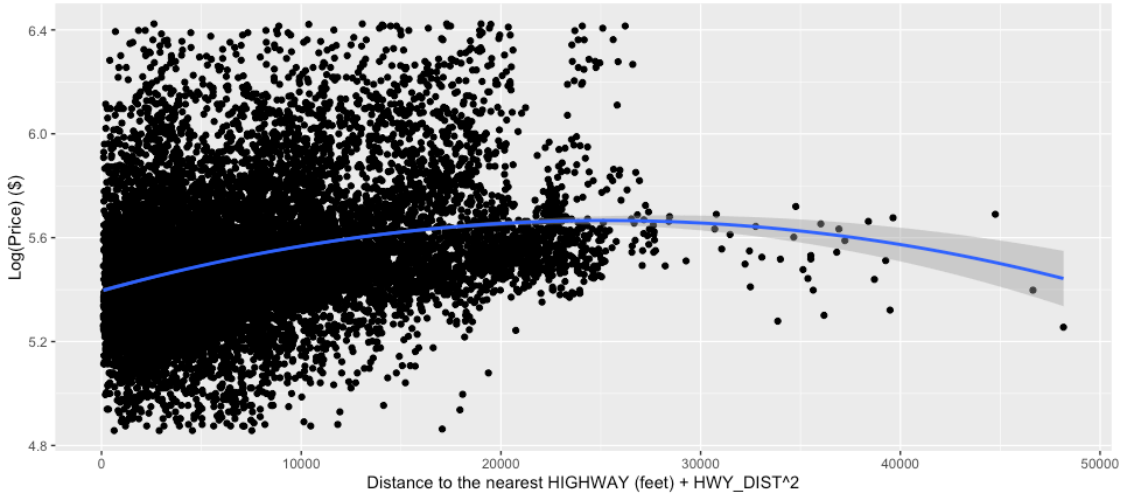



```
par(mfrow=c(1,1))
```

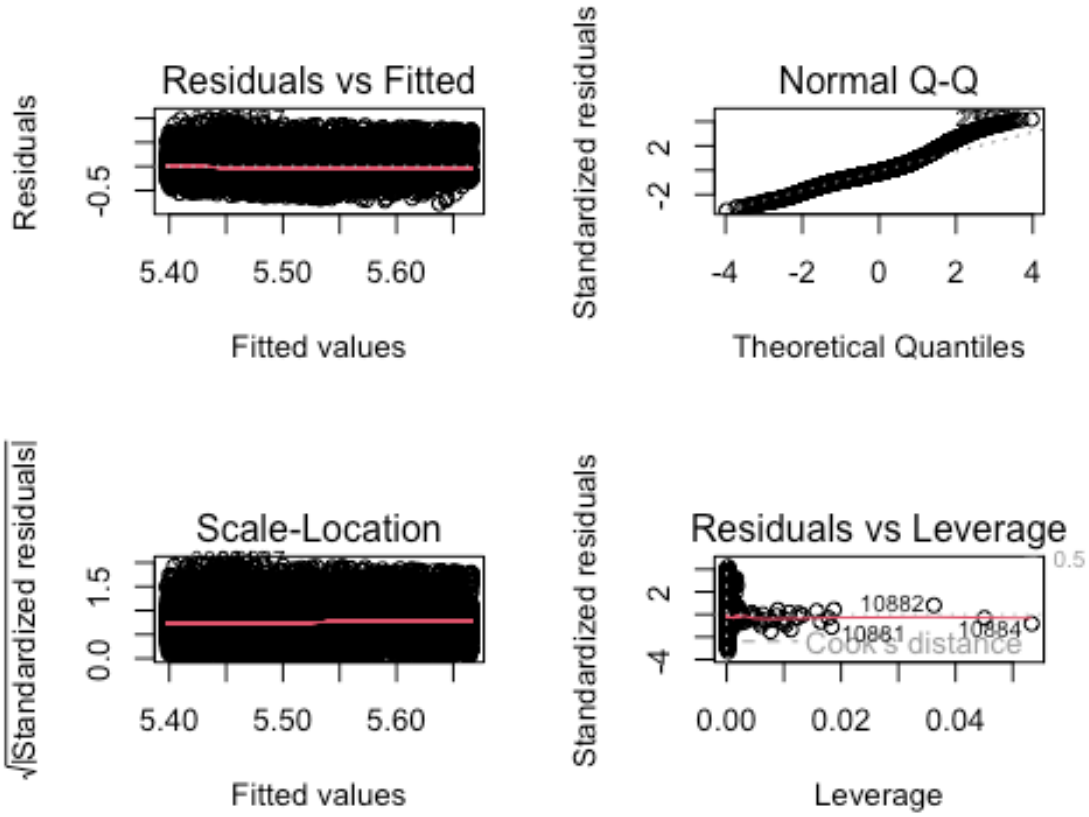
The simple model is not able to fit the data and from what the residuals plot shows the distribution of the residuals is not satisfying. So, it shows that a more complex model is needed to fit the data. The second-degree polynomial for this variable has been added to the model.

```
# Degree 2 for HIGHWAY Distance #
model.HWY_DIST2 <- lm(data=Miami_house, log10_SALE_PRC ~ HWY_DIST +
I(HWY_DIST**2))
summary(model.HWY_DIST2)

##
## Call:
## lm(formula = log10_SALE_PRC ~ HWY_DIST + I(HWY_DIST^2), data = Miami_house
)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.77450 -0.14630 -0.03377  0.10774  0.98319
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)    5.396e+00  4.384e-03 1230.84   <2e-16 ***
## HWY_DIST       2.144e-05  9.394e-07   22.82   <2e-16 ***
## I(HWY_DIST^2) -4.248e-10  3.962e-11  -10.72   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2346 on 13929 degrees of freedom
## Multiple R-squared:  0.0947, Adjusted R-squared:  0.09457
## F-statistic: 728.5 on 2 and 13929 DF,  p-value: < 2.2e-16

BIC(model.HWY_DIST2)

## [1] -821.1056

# Plotting model #
ggplot(Miami_house, aes(x = HWY_DIST, y = log10_SALE_PRC)) +
  geom_point() +
  stat_smooth(method = "lm", formula = y ~ poly(x, 2)) +
  labs(x = 'Distance to the nearest HIGHWAY (feet) + HWY_DIST^2',
  y = 'Log(Price) ($)')
```




```
# Diagnostic
par(mfrow=c(2,2))
plot(model.HWY_DIST2)
```




```
par(mfrow=c(1,1))

anova(model.HWY_DIST, model.HWY_DIST2)

## Analysis of Variance Table
##
## Model 1: log10_SALE_PRC ~ HWY_DIST
## Model 2: log10_SALE_PRC ~ HWY_DIST + I(HWY_DIST^2)
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1  13930 773.26
## 2  13929 766.93  1    6.3304 114.97 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The second-degree fits better the data and the residuals plot shows the better distribution of the residuals and also scale-location plot shows that the horizontal red line is flatter. The small P-value ANOVA function shows that HWY_DIST^2 provides a significantly better fit than the model without. Since there was an improvement to the model by adding a higher degree polynomial the 3rd degree has been also checked.

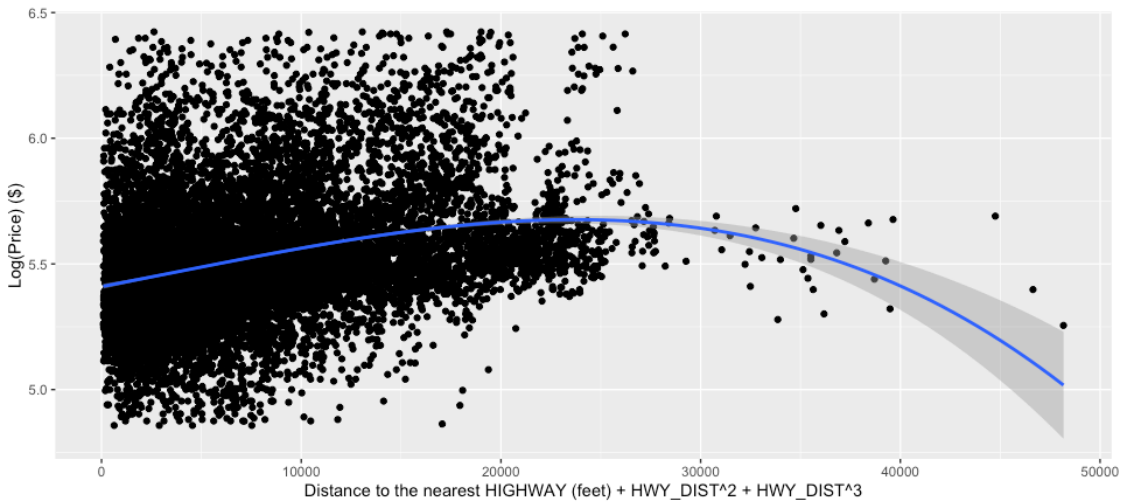
```
# log(price) vs HWY_DIST + HWY_DIST2 + HWY_DIST3 #
model.HWY_DIST3 <- lm(data=Miami_house, log10_SALE_PRC ~ HWY_DIST +
I(HWY_DIST**2) + I(HWY_DIST**3))
summary(model.HWY_DIST3)
```

```
##
## Call:
## lm(formula = log10_SALE_PRC ~ HWY_DIST + I(HWY_DIST^2) + I(HWY_DIST^3),
##     data = Miami_house)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.78208 -0.14694 -0.03331  0.10700  0.97316
##
## Coefficients:
##                  Estimate Std. Error  t value Pr(>|t|)
## (Intercept)    5.410e+00  5.383e-03 1004.899  < 2e-16 ***
## HWY_DIST       1.513e-05  1.682e-06    9.001  < 2e-16 ***
## I(HWY_DIST^2)  1.446e-10  1.321e-10    1.095    0.274
## I(HWY_DIST^3) -1.303e-14  2.884e-15   -4.520 6.24e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2345 on 13928 degrees of freedom
## Multiple R-squared:  0.09602,    Adjusted R-squared:  0.09583
## F-statistic: 493.2 on 3 and 13928 DF,  p-value: < 2.2e-16
```

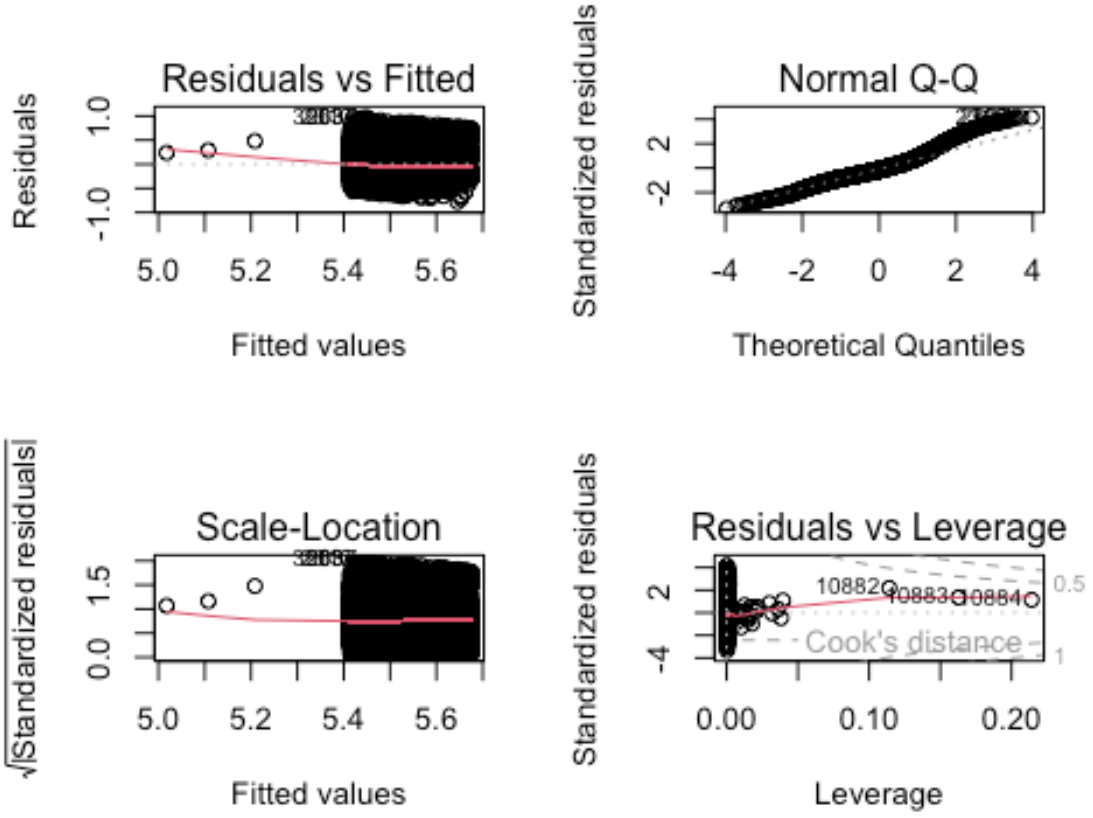
```
BIC(model.HWY_DIST3)
```

```
## [1] -831.9845
```

```
# Plotting model #
ggplot(Miami_house, aes(x = HWY_DIST, y = log10_SALE_PRC)) +
  geom_point() +
  stat_smooth(method = "lm", formula = y ~ poly(x, 3)) +
  labs(x = 'Distance to the nearest HIGHWAY (feet) + HWY_DIST^2 + HWY_DIST^3'
,
  y = 'Log(Price) ($)')
```



```
# Diagnostic
par(mfrow=c(2,2))
plot(model.HWY_DIST3)
```




```
par(mfrow=c(1,1))

anova(model.HWY_DIST2, model.HWY_DIST3)

## Analysis of Variance Table
##
## Model 1: log10_SALE_PRC ~ HWY_DIST + I(HWY_DIST^2)
## Model 2: log10_SALE_PRC ~ HWY_DIST + I(HWY_DIST^2) + I(HWY_DIST^3)
##   Res.Df    RSS Df Sum of Sq     F    Pr(>F)
## 1  13929 766.93
## 2  13928 765.80  1    1.1233 20.43 6.237e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

From what can be seen from the results there is not a big improvement in interpolating the data by adding the 3rd degree polynomial of the variable HWY_DIST and also the residuals plot shows that the results are not as good as the ones received by the 2nd-degree polynomial. So, the second-degree polynomial model for the variable HWY_DIST is selected.

After having done this analysis there are 5 models which are as follows:

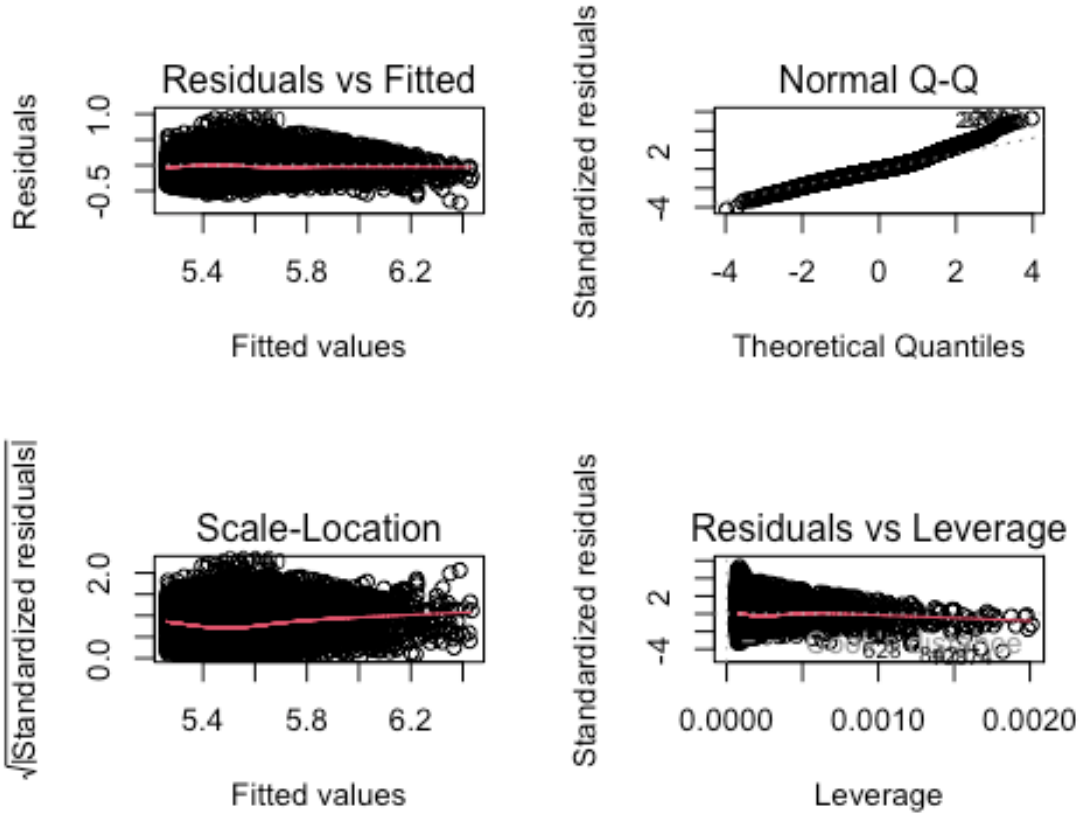
1. A simple model with only the first-degree component of the variable TOT_LVG_AREA.
2. A simple model with a first and second-degree component of the variable TOT_LVG_AREA.
3. A model with all the 6 variables with first-degree:
    1. TOT_LVG_AREA
    2. SUBCNTR_DI
    3. SPEC_FEAT_VAL
    4. HWY_DIST
    5. WATER_DIST
    6. RAIL_DIST
4. A model with all the 6 variables:
    1. TOT_LVG_AREA, with second-degree polynomial
    2. SUBCNTR_DI
    3. SPEC_FEAT_VAL
    4. HWY_DIST
    5. WATER_DIST
    6. RAIL_DIST
5. A model with all the 6 variables:
    1. TOT_LVG_AREA, with second-degree polynomial
    2. SUBCNTR_DI, with 4^{th}-degree polynomial
    3. SPEC_FEAT_VAL, with second-degree polynomial
    4. HWY_DIST, with second-degree polynomial
    5. WATER_DIST, with 4^{th}-degree polynomial
    6. RAIL_DIST

In this step, all of the numerical models have been compared to each other.

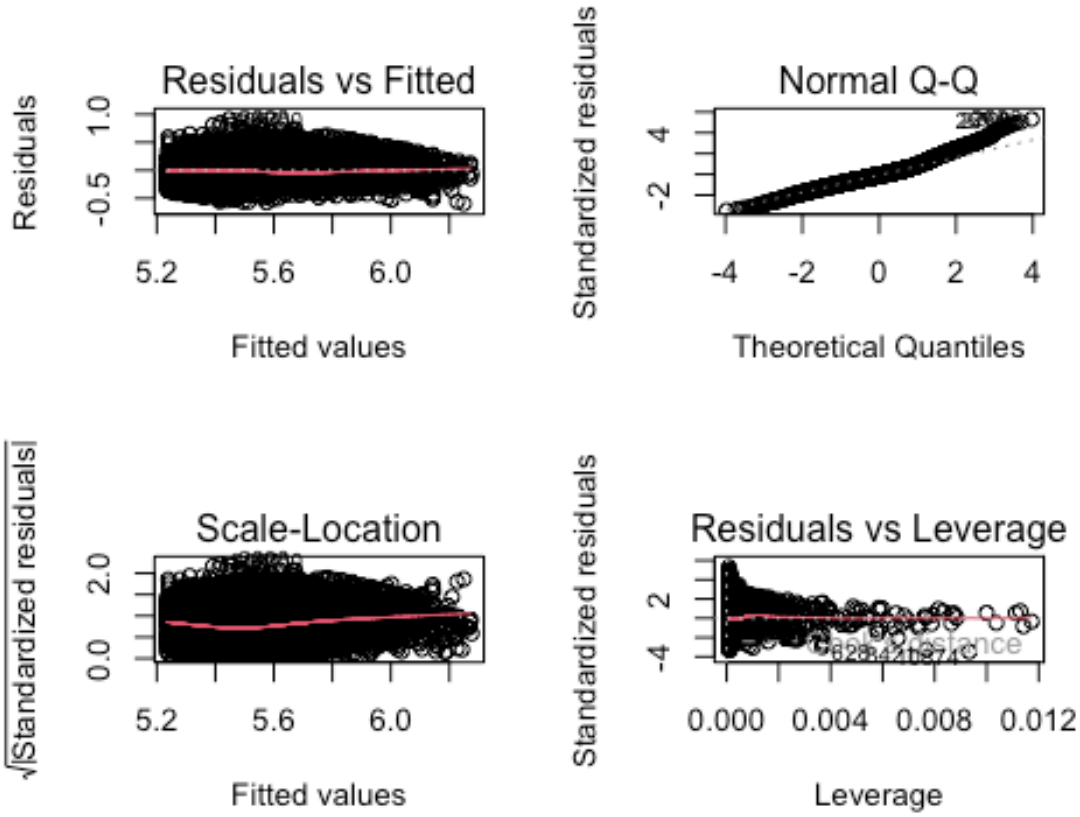
```
# The first numerical model with one variable, (TOT_LVG_AREA) #
summary(model.num1)
```

```
##
## Call:
## lm(formula = log10_SALE_PRC ~ TOT_LVG_AREA, data = Miami_house)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.73501 -0.10746 -0.01606  0.08546  0.91663
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  5.076e+00  3.985e-03    1274   <2e-16 ***
## TOT_LVG_AREA 2.161e-04  1.801e-06     120   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1729 on 13930 degrees of freedom
## Multiple R-squared:  0.5083, Adjusted R-squared:  0.5083
## F-statistic: 1.44e+04 on 1 and 13930 DF,  p-value: < 2.2e-16
```

```
# Diagnostic
par(mfrow=c(2,2))
plot(model.num1)
```

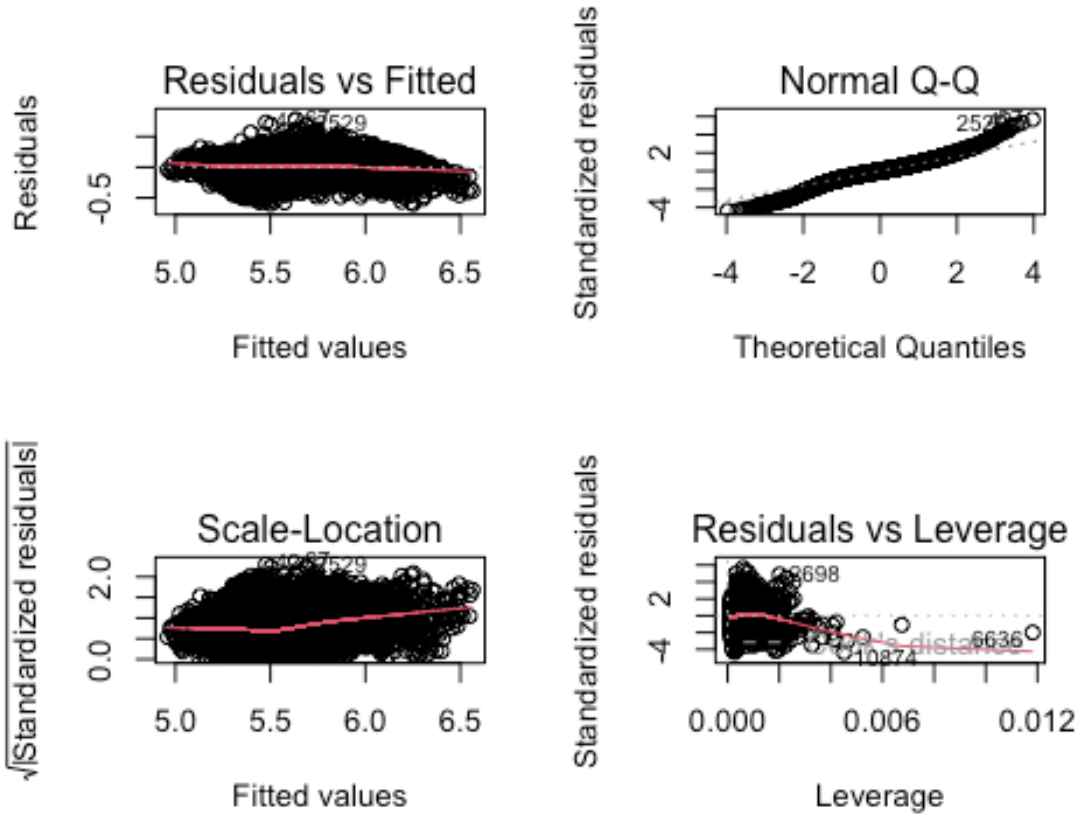


```r
par(mfrow=c(1,1))

# The second numerical model with 2 variables, (TOT_LVG_AREA) and (TOT_LVG_AR
EA**2)#
summary(model.num2)

##
## Call:
## lm(formula = log10_SALE_PRC ~ TOT_LVG_AREA + I(TOT_LVG_AREA^2),
##     data = Miami_house)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.59598 -0.10963 -0.01754  0.08647  0.91463
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         5.002e+00  8.815e-03 567.480   <2e-16 ***
## TOT_LVG_AREA        2.816e-04  7.261e-06  38.779   <2e-16 ***
## I(TOT_LVG_AREA^2) -1.256e-08  1.350e-09  -9.306   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1724 on 13929 degrees of freedom
## Multiple R-squared:  0.5114, Adjusted R-squared:  0.5113
## F-statistic:  7289 on 2 and 13929 DF,  p-value: < 2.2e-16

# Diagnostic
par(mfrow=c(2,2))
plot(model.num2)
```



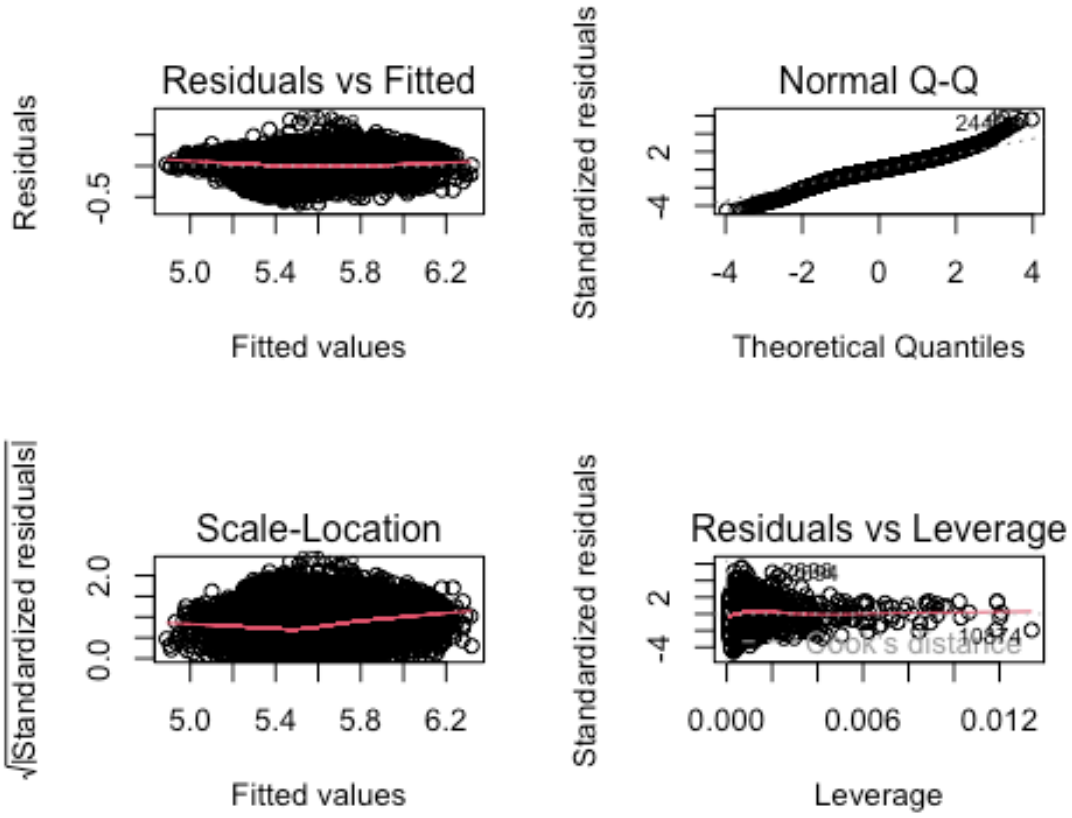

```
par(mfrow=c(1,1))

# The third numerical model with 6 variables, all the 6 variables with degree
1 #
model.poly1 <- lm(log10_SALE_PRC ~ TOT_LVG_AREA + SPEC_FEAT_VAL +
RAIL_DIST + WATER_DIST + SUBCNTR_DI + HWY_DIST, data=Miami_house)
summary(model.poly1)

##
## Call:
## lm(formula = log10_SALE_PRC ~ TOT_LVG_AREA + SPEC_FEAT_VAL +
##     RAIL_DIST + WATER_DIST + SUBCNTR_DI + HWY_DIST, data = Miami_house)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.59955 -0.07172  0.00180  0.07691  0.78157
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    5.234e+00  4.104e-03 1275.54   <2e-16 ***
## TOT_LVG_AREA   1.851e-04  1.707e-06  108.39   <2e-16 ***
## SPEC_FEAT_VAL  2.133e-06  9.873e-08   21.61   <2e-16 ***
## RAIL_DIST      3.854e-06  2.193e-07   17.58   <2e-16 ***
```

```
## WATER_DIST    -2.759e-06  1.114e-07  -24.75   <2e-16 ***
## SUBCNTR_DI    -4.042e-06  6.196e-08  -65.24   <2e-16 ***
## HWY_DIST       6.703e-06  2.190e-07   30.61   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1377 on 13925 degrees of freedom
## Multiple R-squared:  0.6884, Adjusted R-squared:  0.6882
## F-statistic:  5127 on 6 and 13925 DF,  p-value: < 2.2e-16

# Diagnostic
par(mfrow=c(2,2))
plot(model.poly1)
```



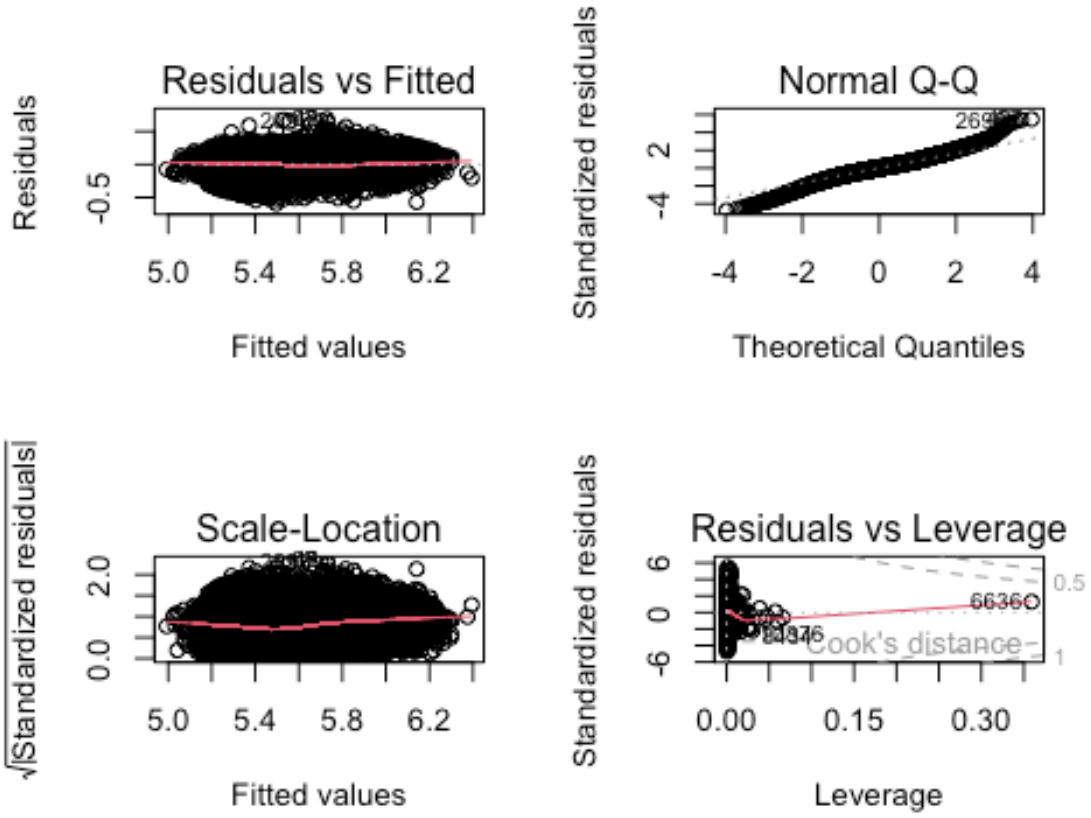

```
par(mfrow=c(1,1))

# The 4th numerical model with all the variables with degree 1 except TOT_LVG
_AREA #
model.poly2 <- lm(log10_SALE_PRC ~ poly(TOT_LVG_AREA,2) +
SPEC_FEAT_VAL + RAIL_DIST + WATER_DIST + SUBCNTR_DI +
HWY_DIST, data=Miami_house)
summary(model.poly2)
```

```
##
## Call:
## lm(formula = log10_SALE_PRC ~ poly(TOT_LVG_AREA, 2) + SPEC_FEAT_VAL +
##     RAIL_DIST + WATER_DIST + SUBCNTR_DI + HWY_DIST, data = Miami_house)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.60766 -0.07559  0.00163  0.07945  0.75287
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)              5.628e+00  3.273e-03 1719.54   <2e-16 ***
## poly(TOT_LVG_AREA, 2)1  1.778e+01  1.598e-01  111.25   <2e-16 ***
## poly(TOT_LVG_AREA, 2)2 -3.738e+00  1.390e-01  -26.90   <2e-16 ***
## SPEC_FEAT_VAL            2.287e-06  9.643e-08   23.72   <2e-16 ***
## RAIL_DIST                3.604e-06  2.140e-07   16.84   <2e-16 ***
## WATER_DIST              -3.232e-06  1.101e-07  -29.36   <2e-16 ***
## SUBCNTR_DI              -4.161e-06  6.057e-08  -68.69   <2e-16 ***
## HWY_DIST                 6.446e-06  2.137e-07   30.16   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1343 on 13924 degrees of freedom
## Multiple R-squared:  0.7038, Adjusted R-squared:  0.7036
## F-statistic:  4726 on 7 and 13924 DF,  p-value: < 2.2e-16

# Diagnostic
par(mfrow=c(2,2))
plot(model.poly2)
```




```
par(mfrow=c(1,1))

# The fifth numerical model with 6 variables, all 6 variables with their best
degree #
model.poly3 <- lm(log10_SALE_PRC ~ poly(TOT_LVG_AREA,2) +
poly(SPEC_FEAT_VAL,2) + RAIL_DIST + poly(WATER_DIST, 4) +
poly(SUBCNTR_DI, 4) + poly(HWY_DIST, 2), data=Miami_house)
summary(model.poly3)

##
## Call:
## lm(formula = log10_SALE_PRC ~ poly(TOT_LVG_AREA, 2) + poly(SPEC_FEAT_VAL,
##     2) + RAIL_DIST + poly(WATER_DIST, 4) + poly(SUBCNTR_DI, 4) +
##     poly(HWY_DIST, 2), data = Miami_house)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.59371 -0.06832  0.00176  0.07102  0.68575
##
## Coefficients:
##                             Estimate Std. Error  t value Pr(>|t|)
## (Intercept)                5.491e+00  2.119e-03 2591.338  < 2e-16 ***
## poly(TOT_LVG_AREA, 2)1     1.715e+01  1.511e-01  113.550  < 2e-16 ***
```

```
## poly(TOT_LVG_AREA, 2)2  -3.603e+00  1.332e-01  -27.055  < 2e-16 ***
## poly(SPEC_FEAT_VAL, 2)1  3.646e+00  1.481e-01   24.624  < 2e-16 ***
## poly(SPEC_FEAT_VAL, 2)2 -5.333e-01  1.288e-01   -4.140 3.49e-05 ***
## RAIL_DIST                3.453e-06  2.198e-07   15.711  < 2e-16 ***
## poly(WATER_DIST, 4)1    -3.492e+00  1.549e-01  -22.549  < 2e-16 ***
## poly(WATER_DIST, 4)2     1.196e+00  1.466e-01    8.164 3.52e-16 ***
## poly(WATER_DIST, 4)3    -2.912e+00  1.278e-01  -22.792  < 2e-16 ***
## poly(WATER_DIST, 4)4     2.709e+00  1.369e-01   19.787  < 2e-16 ***
## poly(SUBCNTR_DI, 4)1    -1.146e+01  1.587e-01  -72.181  < 2e-16 ***
## poly(SUBCNTR_DI, 4)2     3.312e+00  1.346e-01   24.612  < 2e-16 ***
## poly(SUBCNTR_DI, 4)3    -2.520e+00  1.374e-01  -18.344  < 2e-16 ***
## poly(SUBCNTR_DI, 4)4     1.412e+00  1.400e-01   10.083  < 2e-16 ***
## poly(HWY_DIST, 2)1       4.664e+00  1.596e-01   29.224  < 2e-16 ***
## poly(HWY_DIST, 2)2      -6.340e-01  1.386e-01   -4.576 4.79e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1251 on 13916 degrees of freedom
## Multiple R-squared:  0.7427, Adjusted R-squared:  0.7425
## F-statistic:  2678 on 15 and 13916 DF,  p-value: < 2.2e-16

# Diagnostic
par(mfrow=c(2,2))
plot(model.poly3)
```




```
par(mfrow=c(1,1))
# Comparing BIC #

BIC(model.num1)

## [1] -9335.795

BIC(model.num2)

## [1] -9412.609

BIC(model.poly1)

## [1] -15640.74

BIC(model.poly2)

## [1] -16336.92

BIC(model.poly3)

## [1] -18225.49
```

This process shows that the fifth model is the best one among other models with a BIC of -18225.49. But all of these models will be checked with each other again when the categorical variables merged with the numerical ones.

3.2. Categorical Variables

In this part, all the categorical variables in the dataset have been processed. The idea is to study the full model at first and then by removing the variables which are not significant and at the end a model will be selected that has the least number of variables with R2 that is not so different from the R2 which the full model has.

```
# CATEGORICAL VARIABLE SELECTION #
model.cat1 <- lm(log10_SALE_PRC ~ avno60plus + month_sold +
structure_quality + has_SPECFEAT + has_BODYOFWATER +
age, data=Miami_house)
summary(model.cat1)

##
## Call:
## lm(formula = log10_SALE_PRC ~ avno60plus + month_sold + structure_quality
+
##     has_SPECFEAT + has_BODYOFWATER + age, data = Miami_house)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.79538 -0.12065 -0.01773  0.10454  0.93028
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          5.204e+00  1.810e-02 287.452  < 2e-16 ***
## avno60plus1         -9.565e-02  1.455e-02  -6.572 5.15e-11 ***
## month_sold2          1.038e-02  9.815e-03   1.058 0.290245
## month_sold3          2.136e-02  9.239e-03   2.312 0.020803 *
## month_sold4          2.151e-02  9.225e-03   2.331 0.019756 *
## month_sold5          4.442e-02  9.207e-03   4.824 1.42e-06 ***
## month_sold6          4.118e-02  9.019e-03   4.566 5.01e-06 ***
## month_sold7          3.120e-02  9.263e-03   3.368 0.000759 ***
## month_sold8          2.892e-02  9.166e-03   3.155 0.001610 **
## month_sold9          3.350e-02  9.252e-03   3.621 0.000294 ***
## month_sold10         3.904e-02  9.537e-03   4.093 4.27e-05 ***
## month_sold11         3.583e-02  9.346e-03   3.833 0.000127 ***
## month_sold12         2.612e-02  9.343e-03   2.795 0.005191 **
## structure_quality2   1.563e-01  1.605e-02   9.743  < 2e-16 ***
## structure_quality3   9.660e-01  5.385e-02  17.940  < 2e-16 ***
## structure_quality4   3.142e-01  1.584e-02  19.840  < 2e-16 ***
## structure_quality5   5.159e-01  1.641e-02  31.440  < 2e-16 ***
## has_SPECFEATTRUE     7.050e-02  4.726e-03  14.919  < 2e-16 ***
## has_BODYOFWATERTRUE  2.068e-01  1.996e-02  10.360  < 2e-16 ***
## age                 -2.113e-03  8.513e-05 -24.822  < 2e-16 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2056 on 13912 degrees of freedom
## Multiple R-squared:  0.3056, Adjusted R-squared:  0.3046
## F-statistic: 322.2 on 19 and 13912 DF,  p-value: < 2.2e-16

BIC(model.cat1)

## [1] -4353.619
```

The month_sold variable is removed since it's not significant.

```
# without month_sold #
model.cat2 <- lm(log10_SALE_PRC ~ avno60plus +
structure_quality + has_SPECFEAT +has_BODYOFWATER +
age, data=Miami_house)
summary(model.cat2)

##
## Call:
## lm(formula = log10_SALE_PRC ~ avno60plus + structure_quality +
##     has_SPECFEAT + has_BODYOFWATER + age, data = Miami_house)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.77917 -0.12056 -0.01844  0.10543  0.93094
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          5.234e+00  1.673e-02 312.877  < 2e-16 ***
## avno60plus1         -9.764e-02  1.456e-02  -6.705  2.1e-11 ***
## structure_quality2   1.563e-01  1.606e-02   9.733  < 2e-16 ***
## structure_quality3   9.656e-01  5.390e-02  17.917  < 2e-16 ***
## structure_quality4   3.142e-01  1.585e-02  19.822  < 2e-16 ***
## structure_quality5   5.154e-01  1.642e-02  31.378  < 2e-16 ***
## has_SPECFEATTRUE     7.031e-02  4.730e-03  14.865  < 2e-16 ***
## has_BODYOFWATERTRUE  2.072e-01  1.998e-02  10.373  < 2e-16 ***
## age                 -2.130e-03  8.515e-05 -25.019  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2059 on 13923 degrees of freedom
## Multiple R-squared:  0.3034, Adjusted R-squared:  0.303
## F-statistic:   758 on 8 and 13923 DF,  p-value: < 2.2e-16

BIC(model.cat2)

## [1] -4414.547
```

From now on variables are being removed one by one and their influence on the model is being checked. In the end, the model with the least number of variables and with an adjusted R2 that is not much different from the adjusted R2 of the full model will be selected.

```
# without month_sold and avno60plus #
model.cat3 <- lm(log10_SALE_PRC ~ structure_quality + has_SPECFEAT +
has_BODYOFWATER + age, data=Miami_house)
summary(model.cat3)

##
## Call:
## lm(formula = log10_SALE_PRC ~ structure_quality + has_SPECFEAT +
##     has_BODYOFWATER + age, data = Miami_house)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.77384 -0.12184 -0.01837  0.10505  0.93531
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          5.238e+00  1.674e-02  312.91   <2e-16 ***
## structure_quality2   1.537e-01  1.608e-02    9.56   <2e-16 ***
## structure_quality3   9.623e-01  5.398e-02   17.83   <2e-16 ***
## structure_quality4   3.105e-01  1.587e-02   19.57   <2e-16 ***
## structure_quality5   5.088e-01  1.642e-02   30.98   <2e-16 ***
## has_SPECFEATTRUE     7.002e-02  4.737e-03   14.78   <2e-16 ***
## has_BODYOFWATERTRUE  2.087e-01  2.001e-02   10.43   <2e-16 ***
## age                 -2.195e-03  8.473e-05  -25.91   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2062 on 13924 degrees of freedom
## Multiple R-squared:  0.3011, Adjusted R-squared:  0.3008
## F-statistic: 857.1 on 7 and 13924 DF,  p-value: < 2.2e-16

BIC(model.cat3)

## [1] -4379.179

# without month_sold and avno60plus and structure #
model.cat4 <- lm(log10_SALE_PRC ~ has_SPECFEAT +
has_BODYOFWATER + age, data=Miami_house)
summary(model.cat4)

##
## Call:
## lm(formula = log10_SALE_PRC ~ has_SPECFEAT + has_BODYOFWATER +
##     age, data = Miami_house)
##
## Residuals:
```

```
##      Min       1Q   Median       3Q      Max
## -0.68271 -0.15034 -0.03108  0.10630  0.96059
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          5.510e+00  5.664e-03  972.84   <2e-16 ***
## has_SPECFEATTRUE     9.606e-02  5.442e-03   17.65   <2e-16 ***
## has_BODYOFWATERTRUE  3.077e-01  2.299e-02   13.38   <2e-16 ***
## age                 -2.372e-03  9.537e-05  -24.87   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2377 on 13928 degrees of freedom
## Multiple R-squared:  0.07083,    Adjusted R-squared:  0.07063
## F-statistic: 353.9 on 3 and 13928 DF,  p-value: < 2.2e-16

BIC(model.cat4)

## [1] -448.9834
```

Seems that the variable structure_quality is a very important variable for the model. So removing it has an adverse effect.

```
# without month_sold and has feature and avno60plus #
model.cat5 <- lm(log10_SALE_PRC ~ structure_quality  +
has_BODYOFWATER + age, data=Miami_house)
summary(model.cat5)

##
## Call:
## lm(formula = log10_SALE_PRC ~ structure_quality + has_BODYOFWATER +
##     age, data = Miami_house)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.76694 -0.12417 -0.02085  0.10567  0.94378
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          5.288e+00  1.652e-02 320.010   <2e-16 ***
## structure_quality2   1.580e-01  1.620e-02   9.752   <2e-16 ***
## structure_quality3   9.641e-01  5.440e-02  17.723   <2e-16 ***
## structure_quality4   3.179e-01  1.598e-02  19.893   <2e-16 ***
## structure_quality5   5.195e-01  1.653e-02  31.420   <2e-16 ***
## has_BODYOFWATERTRUE  2.147e-01  2.016e-02  10.650   <2e-16 ***
## age                 -2.139e-03  8.531e-05 -25.073   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2078 on 13925 degrees of freedom
```

```
## Multiple R-squared:  0.2902, Adjusted R-squared:  0.2899
## F-statistic: 948.7 on 6 and 13925 DF,  p-value: < 2.2e-16

BIC(model.cat5)

## [1] -4171.789

# without month_sold and has water and avno60plus and has feature #
model.cat6 <- lm(log10_SALE_PRC ~ structure_quality + age, data=Miami_house)
summary(model.cat6)

##
## Call:
## lm(formula = log10_SALE_PRC ~ structure_quality + age, data = Miami_house)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.77293 -0.12390 -0.02053  0.10558  0.97084
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         5.285e+00  1.659e-02 318.611   <2e-16 ***
## structure_quality2  1.599e-01  1.627e-02   9.829   <2e-16 ***
## structure_quality3  9.660e-01  5.462e-02  17.686   <2e-16 ***
## structure_quality4  3.206e-01  1.604e-02  19.980   <2e-16 ***
## structure_quality5  5.262e-01  1.659e-02  31.722   <2e-16 ***
## age                -2.102e-03  8.558e-05 -24.565   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2086 on 13926 degrees of freedom
## Multiple R-squared:  0.2844, Adjusted R-squared:  0.2841
## F-statistic:  1107 on 5 and 13926 DF,  p-value: < 2.2e-16

BIC(model.cat6)

## [1] -4068.306

# without month_sold and has water and avno60plus and has feature and age #
model.cat7 <- lm(log10_SALE_PRC ~ structure_quality, data=Miami_house)
summary(model.cat7)

##
## Call:
## lm(formula = log10_SALE_PRC ~ structure_quality, data = Miami_house)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.80911 -0.12368 -0.01266  0.11456  0.96745
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)         5.14650    0.01593  323.10   <2e-16 ***
## structure_quality2  0.24091    0.01627   14.81   <2e-16 ***
## structure_quality3  1.07328    0.05561   19.30   <2e-16 ***
## structure_quality4  0.39216    0.01611   24.34   <2e-16 ***
## structure_quality5  0.60462    0.01663   36.37   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2131 on 13927 degrees of freedom
## Multiple R-squared:  0.2534, Adjusted R-squared:  0.2532
## F-statistic:  1182 on 4 and 13927 DF,  p-value: < 2.2e-16

BIC(model.cat7)

## [1] -3486.87

# Checking on the R2 of these models #
summary(model.cat1)$adj.r.squared

## [1] 0.3046346

summary(model.cat2)$adj.r.squared

## [1] 0.3029845

summary(model.cat3)$adj.r.squared

## [1] 0.3007842

summary(model.cat4)$adj.r.squared

## [1] 0.07062701

summary(model.cat5)$adj.r.squared

## [1] 0.2898627

summary(model.cat6)$adj.r.squared

## [1] 0.2841296

summary(model.cat7)$adj.r.squared

## [1] 0.2531637
```

The analysis showed that the variables structure_quality and age are important variables for the model and the model containing only these two variables has an R2 value of 0.2841 in comparison to the full model which was 0.3040. There is not a specific difference between these two considering that the full model contained 6 variables but the model.cat6 only contains 2 variables. So the selected model as the best categorical model in this study is model cat6.

Final Model

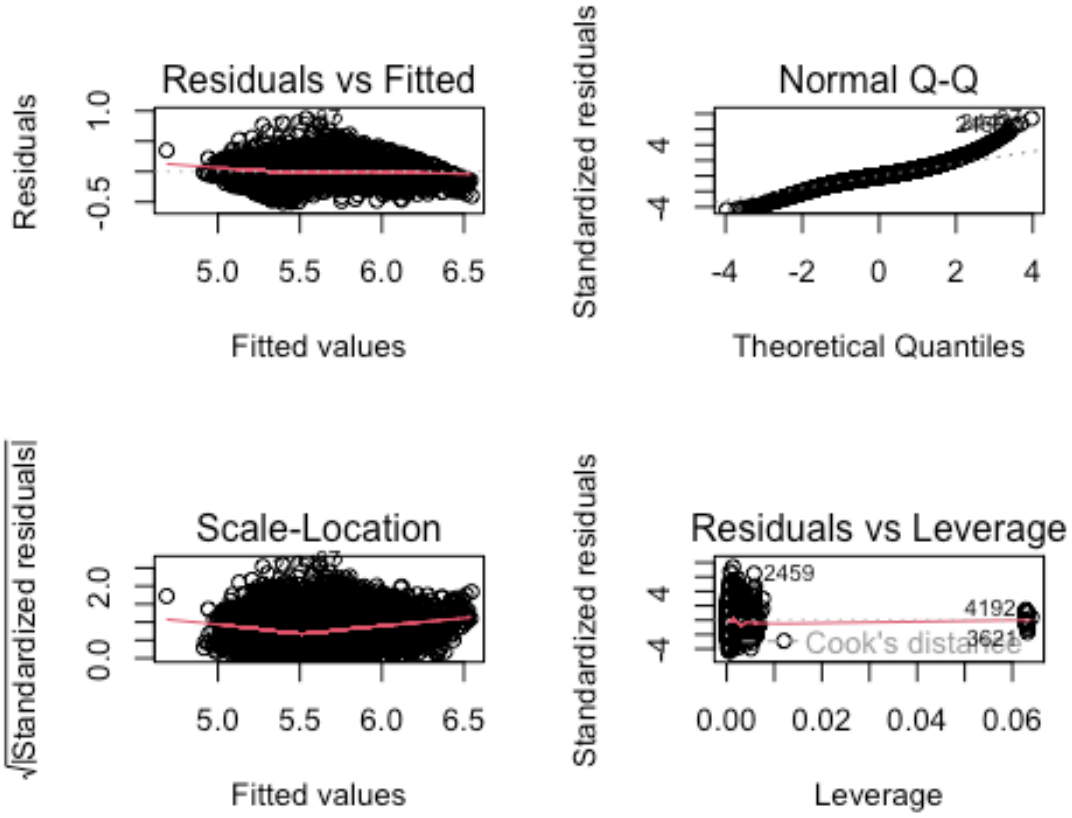
4.1. Final Model selection

In this part, the numerical models which contain numerical variables (TOT_LVG_AREA, SPEC_FEAT_VAL, RAIL_DIST, WATER_DIST, SUBCNTR_DI, and HWY_DIST) and, the categorical model which contains categorical variables (structure_quality and age) will be combined. There are 3 models which are as follows:

1. Model Poly1 + Cat6
2. Model Poly2 + Cat6
3. Model Poly3 + Cat6

```
# Poly1 + CAT6 #
model.Poly1Cat6 <- lm(log10_SALE_PRC ~ TOT_LVG_AREA +
SPEC_FEAT_VAL + RAIL_DIST + WATER_DIST + SUBCNTR_DI +
HWY_DIST + structure_quality + age, data=Miami_house)
summary(model.Poly1Cat6)

##
## Call:
## lm(formula = log10_SALE_PRC ~ TOT_LVG_AREA + SPEC_FEAT_VAL +
##     RAIL_DIST + WATER_DIST + SUBCNTR_DI + HWY_DIST + structure_quality +
##     age, data = Miami_house)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.51647 -0.06455 -0.00386  0.06207  0.87817
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         5.122e+00  1.050e-02  487.75   <2e-16 ***
## TOT_LVG_AREA        1.584e-04  1.571e-06  100.86   <2e-16 ***
## SPEC_FEAT_VAL       1.807e-06  8.542e-08   21.15   <2e-16 ***
## RAIL_DIST           3.209e-06  1.896e-07   16.93   <2e-16 ***
## WATER_DIST         -2.449e-06  1.024e-07  -23.92   <2e-16 ***
## SUBCNTR_DI         -4.162e-06  5.844e-08  -71.21   <2e-16 ***
## HWY_DIST            5.114e-06  1.918e-07   26.67   <2e-16 ***
## structure_quality2  1.741e-01  9.284e-03   18.75   <2e-16 ***
## structure_quality3  5.067e-01  3.123e-02   16.22   <2e-16 ***
## structure_quality4  2.472e-01  9.162e-03   26.98   <2e-16 ***
## structure_quality5  3.600e-01  9.535e-03   37.76   <2e-16 ***
## age                -1.617e-03  5.850e-05  -27.63   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1187 on 13920 degrees of freedom
```

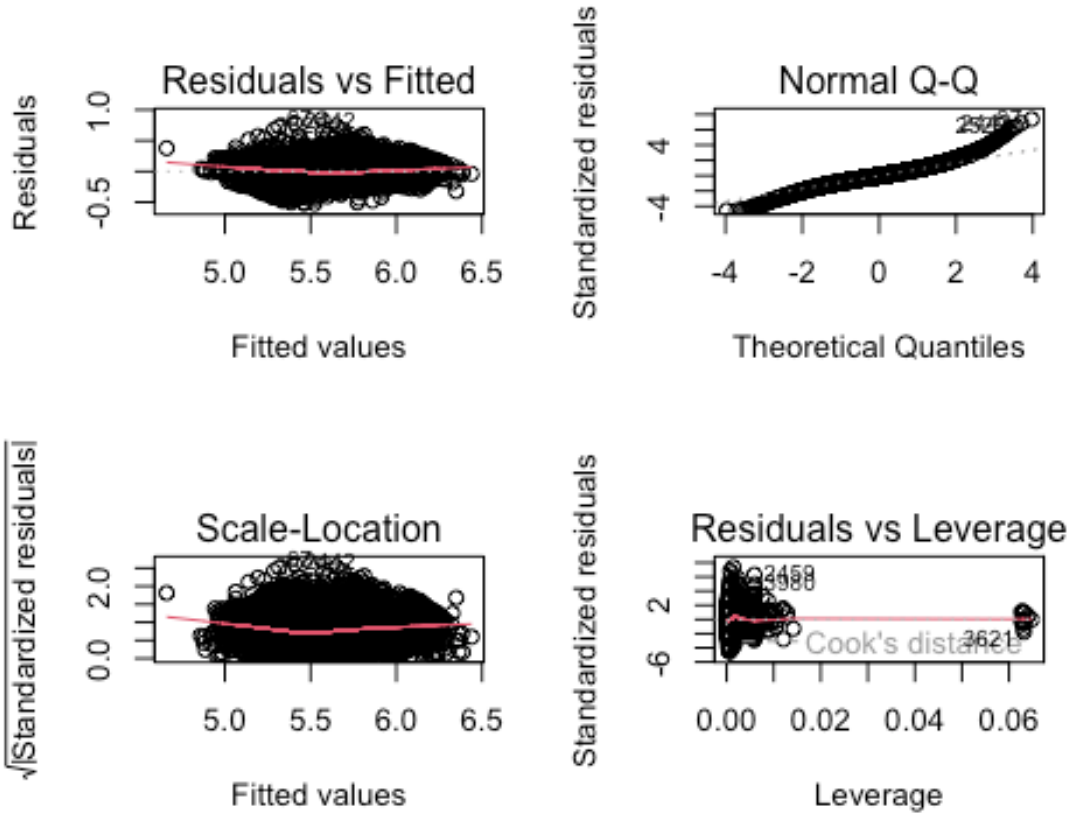
```
## Multiple R-squared:  0.7683, Adjusted R-squared:  0.7682
## F-statistic:  4197 on 11 and 13920 DF,  p-value: < 2.2e-16

# Diagnostic
par(mfrow=c(2,2))
plot(model.Poly1Cat6)
```




```
BIC(model.Poly1Cat6)

## [1] -19724.21

par(mfrow=c(1,1))

# Poly2 + CAT6 #
model.Poly2Cat6 <- lm(log10_SALE_PRC ~ poly(TOT_LVG_AREA,2) +
SPEC_FEAT_VAL + RAIL_DIST + WATER_DIST + SUBCNTR_DI +
HWY_DIST + structure_quality + age, data=Miami_house)
summary(model.Poly2Cat6)

##
## Call:
## lm(formula = log10_SALE_PRC ~ poly(TOT_LVG_AREA, 2) + SPEC_FEAT_VAL +
##     RAIL_DIST + WATER_DIST + SUBCNTR_DI + HWY_DIST + structure_quality +
##     age, data = Miami_house)
```

```
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.52349 -0.06821 -0.00416  0.06272  0.84894
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)              5.468e+00  9.745e-03  561.11   <2e-16 ***
## poly(TOT_LVG_AREA, 2)1   1.528e+01  1.465e-01  104.31   <2e-16 ***
## poly(TOT_LVG_AREA, 2)2 -3.515e+00  1.208e-01  -29.11   <2e-16 ***
## SPEC_FEAT_VAL            1.950e-06  8.308e-08   23.47   <2e-16 ***
## RAIL_DIST                3.002e-06  1.842e-07   16.30   <2e-16 ***
## WATER_DIST              -2.767e-06  9.999e-08  -27.67   <2e-16 ***
## SUBCNTR_DI              -4.251e-06  5.683e-08  -74.81   <2e-16 ***
## HWY_DIST                 4.921e-06  1.863e-07   26.41   <2e-16 ***
## structure_quality2       1.614e-01  9.025e-03   17.89   <2e-16 ***
## structure_quality3       5.045e-01  3.033e-02   16.64   <2e-16 ***
## structure_quality4       2.290e-01  8.917e-03   25.68   <2e-16 ***
## structure_quality5       3.513e-01  9.263e-03   37.93   <2e-16 ***
## age                     -1.488e-03  5.698e-05  -26.11   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1153 on 13919 degrees of freedom
## Multiple R-squared:  0.7816, Adjusted R-squared:  0.7814
## F-statistic:  4152 on 12 and 13919 DF,  p-value: < 2.2e-16
```

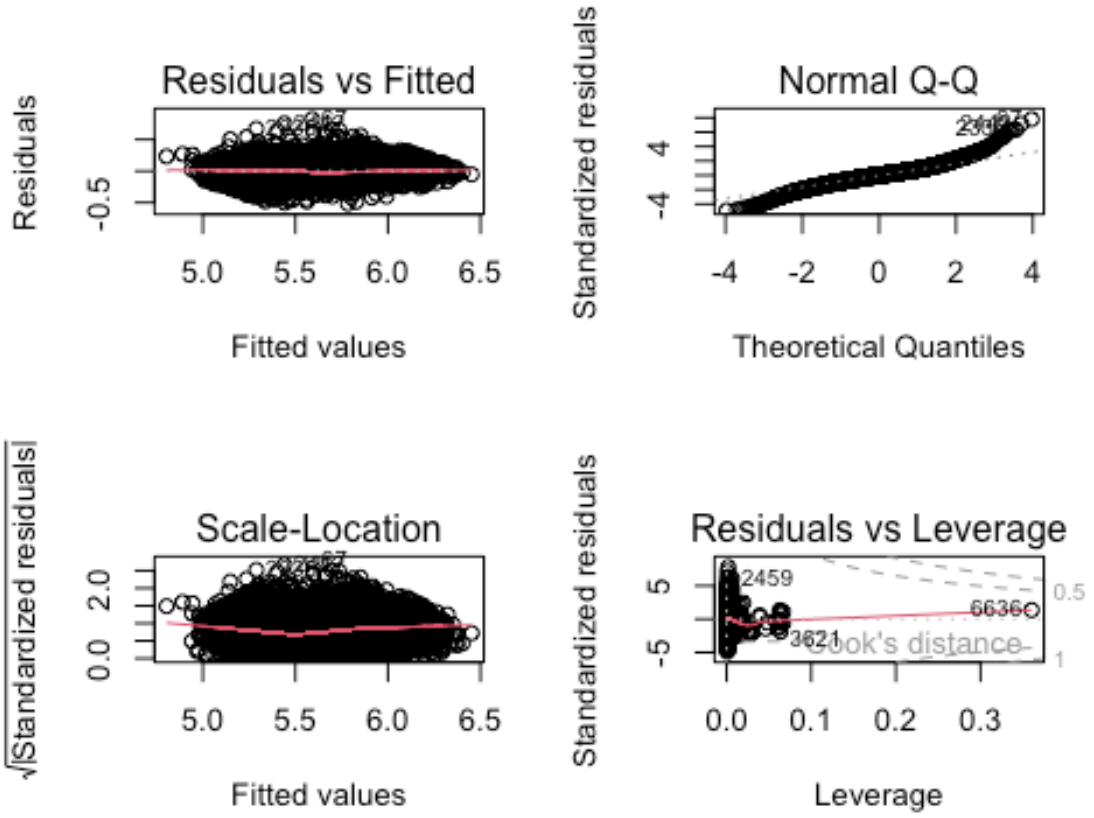
```
# Diagnostic
par(mfrow=c(2,2))
plot(model.Poly2Cat6)
```




```
BIC(model.Poly2Cat6)

## [1] -20537.78

par(mfrow=c(1,1))

# Poly3 + CAT6 #
model.Poly3Cat6 <- lm(log10_SALE_PRC ~ poly(TOT_LVG_AREA,2) +
poly(SPEC_FEAT_VAL,2) + RAIL_DIST + poly(WATER_DIST, 4) +
poly(SUBCNTR_DI, 4) + poly(HWY_DIST, 2) + structure_quality +
  age, data=Miami_house)
summary(model.Poly3Cat6)

##
## Call:
## lm(formula = log10_SALE_PRC ~ poly(TOT_LVG_AREA, 2) + poly(SPEC_FEAT_VAL,
##     2) + RAIL_DIST + poly(WATER_DIST, 4) + poly(SUBCNTR_DI, 4) +
##     poly(HWY_DIST, 2) + structure_quality + age, data = Miami_house)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.51662 -0.05955  0.00088  0.05562  0.82368
##
## Coefficients:
```

```
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 5.345e+00  8.580e-03 623.048  < 2e-16 ***
## poly(TOT_LVG_AREA, 2)1      1.450e+01  1.357e-01 106.784  < 2e-16 ***
## poly(TOT_LVG_AREA, 2)2     -3.331e+00  1.139e-01 -29.254  < 2e-16 ***
## poly(SPEC_FEAT_VAL, 2)1     3.133e+00  1.255e-01  24.965  < 2e-16 ***
## poly(SPEC_FEAT_VAL, 2)2    -6.604e-01  1.091e-01  -6.051 1.47e-09 ***
## RAIL_DIST                   2.848e-06  1.860e-07  15.306  < 2e-16 ***
## poly(WATER_DIST, 4)1       -3.556e+00  1.370e-01 -25.948  < 2e-16 ***
## poly(WATER_DIST, 4)2        9.271e-01  1.267e-01   7.317 2.68e-13 ***
## poly(WATER_DIST, 4)3       -2.361e+00  1.101e-01 -21.440  < 2e-16 ***
## poly(WATER_DIST, 4)4        2.282e+00  1.162e-01  19.628  < 2e-16 ***
## poly(SUBCNTR_DI, 4)1       -1.182e+01  1.446e-01 -81.726  < 2e-16 ***
## poly(SUBCNTR_DI, 4)2        4.200e+00  1.151e-01  36.482  < 2e-16 ***
## poly(SUBCNTR_DI, 4)3       -1.125e+00  1.178e-01  -9.553  < 2e-16 ***
## poly(SUBCNTR_DI, 4)4        3.882e-01  1.197e-01   3.244  0.00118 **
## poly(HWY_DIST, 2)1          3.993e+00  1.353e-01  29.517  < 2e-16 ***
## poly(HWY_DIST, 2)2         -1.510e-01  1.173e-01  -1.287  0.19802
## structure_quality2          1.428e-01  8.304e-03  17.202  < 2e-16 ***
## structure_quality3          4.251e-01  2.788e-02  15.249  < 2e-16 ***
## structure_quality4          2.157e-01  8.218e-03  26.251  < 2e-16 ***
## structure_quality5          3.244e-01  8.567e-03  37.871  < 2e-16 ***
## age                        -1.834e-03  5.348e-05 -34.289  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1057 on 13911 degrees of freedom
## Multiple R-squared:  0.8164, Adjusted R-squared:  0.8161
## F-statistic:  3092 on 20 and 13911 DF,  p-value: < 2.2e-16
```

```r
# Diagnostic
par(mfrow=c(2,2))
plot(model.Poly3Cat6)
```




```
BIC(model.Poly3Cat6)

## [1] -22875.5
```

Considering all the three final models which are the combination of numerical models and the categorical model. The first one has the adjusted R2 as 0.7682 and BIC as -19724.21 but the second model has adjusted R2 as 0.7814 and BIC as -20537.78, for the third model also the adjusted R2 is 0.8161 and BIC is -22875.5, which in comparison to the first two models has better results. Considering the residual plot for the third model it's obvious that the residuals are normally distributed around zero also the scale-location plot shows a good distribution and the horizontal line is flat. The distances in the QQ plot on both ends are acceptable.

4.2. Cross-Validation of the Final model

In this part, the mean squared error and the 10-fold cross-validation error have been calculated to check the performance of the final best model in this study.

Where Mean Squared Error is:

$$MSE = \frac{1}{n}\Sigma(y_i - \overline{y_i})^2$$

and Cross-validation is:

$$CV_k = \Sigma \frac{n_k}{n} MSE_k$$

```
finalbest.fit <- lm(log10_SALE_PRC ~ poly(TOT_LVG_AREA,2) +
poly(SPEC_FEAT_VAL, 2) + RAIL_DIST + poly(WATER_DIST, 4) +
poly(SUBCNTR_DI, 4) + poly(HWY_DIST, 2) + structure_quality +
age, data=Miami_house)

model.glm <- glm(log10_SALE_PRC ~ poly(TOT_LVG_AREA,2) +
poly(SPEC_FEAT_VAL, 2) + RAIL_DIST + poly(WATER_DIST, 4) +
poly(SUBCNTR_DI, 4) + poly(HWY_DIST, 2) + structure_quality +
age, data=Miami_house)
summary(model.glm)

##
## Call:
## glm(formula = log10_SALE_PRC ~ poly(TOT_LVG_AREA, 2) + poly(SPEC_FEAT_VAL,
##     2) + RAIL_DIST + poly(WATER_DIST, 4) + poly(SUBCNTR_DI, 4) +
##     poly(HWY_DIST, 2) + structure_quality + age, data = Miami_house)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -0.51662  -0.05955   0.00088   0.05562   0.82368
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)                5.345e+00  8.580e-03 623.048  < 2e-16 ***
## poly(TOT_LVG_AREA, 2)1     1.450e+01  1.357e-01 106.784  < 2e-16 ***
## poly(TOT_LVG_AREA, 2)2    -3.331e+00  1.139e-01 -29.254  < 2e-16 ***
## poly(SPEC_FEAT_VAL, 2)1    3.133e+00  1.255e-01  24.965  < 2e-16 ***
## poly(SPEC_FEAT_VAL, 2)2   -6.604e-01  1.091e-01  -6.051 1.47e-09 ***
## RAIL_DIST                  2.848e-06  1.860e-07  15.306  < 2e-16 ***
## poly(WATER_DIST, 4)1      -3.556e+00  1.370e-01 -25.948  < 2e-16 ***
## poly(WATER_DIST, 4)2       9.271e-01  1.267e-01   7.317 2.68e-13 ***
## poly(WATER_DIST, 4)3      -2.361e+00  1.101e-01 -21.440  < 2e-16 ***
## poly(WATER_DIST, 4)4       2.282e+00  1.162e-01  19.628  < 2e-16 ***
## poly(SUBCNTR_DI, 4)1      -1.182e+01  1.446e-01 -81.726  < 2e-16 ***
## poly(SUBCNTR_DI, 4)2       4.200e+00  1.151e-01  36.482  < 2e-16 ***
## poly(SUBCNTR_DI, 4)3      -1.125e+00  1.178e-01  -9.553  < 2e-16 ***
## poly(SUBCNTR_DI, 4)4       3.882e-01  1.197e-01   3.244  0.00118 **
## poly(HWY_DIST, 2)1         3.993e+00  1.353e-01  29.517  < 2e-16 ***
## poly(HWY_DIST, 2)2        -1.510e-01  1.173e-01  -1.287  0.19802
## structure_quality2         1.428e-01  8.304e-03  17.202  < 2e-16 ***
## structure_quality3         4.251e-01  2.788e-02  15.249  < 2e-16 ***
## structure_quality4         2.157e-01  8.218e-03  26.251  < 2e-16 ***
## structure_quality5         3.244e-01  8.567e-03  37.871  < 2e-16 ***
## age                       -1.834e-03  5.348e-05 -34.289  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## (Dispersion parameter for gaussian family taken to be 0.01118283)
##
##     Null deviance: 847.15  on 13931  degrees of freedom
## Residual deviance: 155.56  on 13911  degrees of freedom
## AIC: -23041
##
## Number of Fisher Scoring iterations: 2
```

```
# Mean Squared Error for final best model #
mse(Miami_house$log10_SALE_PRC, fitted.values(finalbest.fit))
```

```
## [1] 0.01116597
```

```
cv.error <- cv.glm(data = Miami_house, model.glm, K=10)
cv.error$delta[1]
```

```
## [1] 0.01120079
```

The mean squared error for the actual values of the response variable and the fitted values using the best final model is 0.01116597 and the 10-fold cross-validation error using the cv.glm function is 0.01120079, which confirms that the best final model that has been obtained in this study is qualified. The final best model has 8 explanatory variables which have been chosen through the analysis in part 3 among 17 variables inside the dataset. This model has an adjusted R2 of about 81.6% which is an indicator of high accuracy.