---

title: "ASSIGNMENT 7"

author: "Rajasekhar Reddy Karna"

date: '2020-10-18'

output:

word\_document: default

html\_document: default

pdf\_document: default

bibliography: bibliography.bib

---

Work individually on this assignment. You are encouraged to collaborate on ideas and strategies pertinent to this assignment. Data for this assignment is focused on real estate transactions recorded from 1964 to 2016 and can be found in Week 6 Housing.xlsx. Using your skills in statistical correlation, multiple regression and R programming, you are interested in the following variables: Sale Price and several other possible predictors.

Using your ‘clean’ data set from the previous week complete the following:

##a. Explain why you chose to remove data points from your ‘clean’ dataset.

##Response:

Please refer announcement for week.

```{r include=FALSE}

library(ggplot2)

library(readxl)

setwd("C:/Users/vahin/Documents/GitHub/dsc520/")

housing\_data <- read\_xlsx("data/week-7-housing.xlsx")

head(housing\_data)

```

Output:



##b. Create two variables; one that will contain the variables Sale Price and Square Foot of Lot (same variables used from previous assignment on simple regression) and one that will contain Sale Price and several additional predictors of your choice. Explain the basis for your additional predictor selections.

##Response:

As per outcome, all variables are positively correlated. Building grade and square feet of living room share aroudn 20% and 15% variation in determining sales price. So picked Building grade and Square feet of living are chosen as predictors for the model over year\_built and bedrooms.

```{r echo=FALSE}

cor(housing\_data$'Sale Price', housing\_data$square\_feet\_total\_living)^2 \* 100

cor.test(housing\_data$'Sale Price', housing\_data$square\_feet\_total\_living)

cor(housing\_data$'Sale Price', housing\_data$building\_grade)^2 \* 100

cor.test(housing\_data$'Sale Price', housing\_data$building\_grade)

cor(housing\_data$'Sale Price', housing\_data$year\_built)^2 \* 100

cor.test(housing\_data$'Sale Price', housing\_data$year\_built)

cor(housing\_data$'Sale Price', housing\_data$bedrooms)^2 \* 100

cor.test(housing\_data$'Sale Price', housing\_data$bedrooms)

sales\_price\_with\_sq\_ft\_lot <- lm(housing\_data$'Sale Price' ~ housing\_data$sq\_ft\_lot, data = housing\_data)

sales\_price\_with\_others <- lm(housing\_data$'Sale Price' ~ housing\_data$sq\_ft\_lot + housing\_data$square\_feet\_total\_living + housing\_data$building\_grade, data = housing\_data)

```

Output:

> cor(housing\_data$'Sale Price', housing\_data$square\_feet\_total\_living)^2 \* 100

[1] 20.66499

> cor.test(housing\_data$'Sale Price', housing\_data$square\_feet\_total\_living)

Pearson's product-moment correlation

data: housing\_data$"Sale Price" and housing\_data$square\_feet\_total\_living

t = 57.884, df = 12863, p-value < 2.2e-16

alternative hypothesis: true correlation is not equal to 0

95 percent confidence interval:

0.4407697 0.4681900

sample estimates:

cor

0.4545876

> cor(housing\_data$'Sale Price', housing\_data$building\_grade)^2 \* 100

[1] 15.30602

> cor.test(housing\_data$'Sale Price', housing\_data$building\_grade)

Pearson's product-moment correlation

data: housing\_data$"Sale Price" and housing\_data$building\_grade

t = 48.214, df = 12863, p-value < 2.2e-16

alternative hypothesis: true correlation is not equal to 0

95 percent confidence interval:

0.3764941 0.4057662

sample estimates:

cor

0.3912291

> cor(housing\_data$'Sale Price', housing\_data$year\_built)^2 \* 100

[1] 5.888934

> cor.test(housing\_data$'Sale Price', housing\_data$year\_built)

Pearson's product-moment correlation

data: housing\_data$"Sale Price" and housing\_data$year\_built

t = 28.371, df = 12863, p-value < 2.2e-16

alternative hypothesis: true correlation is not equal to 0

95 percent confidence interval:

0.2263401 0.2588660

sample estimates:

cor

0.2426713

> cor(housing\_data$'Sale Price', housing\_data$bedrooms)^2 \* 100

[1] 5.083558

> cor.test(housing\_data$'Sale Price', housing\_data$bedrooms)

Pearson's product-moment correlation

data: housing\_data$"Sale Price" and housing\_data$bedrooms

t = 26.247, df = 12863, p-value < 2.2e-16

alternative hypothesis: true correlation is not equal to 0

95 percent confidence interval:

0.2090015 0.2418056

sample estimates:

cor

0.2254675

##c. Execute a summary() function on two variables defined in the previous step to compare the model results. What are the R2 and Adjusted R2 statistics? Explain what these results tell you about the overall model. Did the inclusion of the additional predictors help explain any large variations found in Sale Price?

##Response:

For the first model R-Squared value is 0.014, which is 1.4% of variation in sale price. For the other two predictors, value increases to 0.213, or 21.3% of the variance in sale price. So a large amount of variation in sale price. R-squared and adjusted r-squared for the model shows difference almost ~0% (0.2131 - 0.2129).

```{r echo=FALSE}

summary(sales\_price\_with\_sq\_ft\_lot)$r.squared

summary(sales\_price\_with\_others)$r.squared

summary(sales\_price\_with\_sq\_ft\_lot)$adj.r.squared

summary(sales\_price\_with\_others)$adj.r.squared

summary(sales\_price\_with\_sq\_ft\_lot)

summary(sales\_price\_with\_others)

```

Output:

> summary(sales\_price\_with\_sq\_ft\_lot)$r.squared

[1] 0.01435497

> summary(sales\_price\_with\_others)$r.squared

[1] 0.2131732

> summary(sales\_price\_with\_sq\_ft\_lot)$adj.r.squared

[1] 0.01427835

> summary(sales\_price\_with\_others)$adj.r.squared

[1] 0.2129897

> summary(sales\_price\_with\_sq\_ft\_lot)

Call:

lm(formula = housing\_data$"Sale Price" ~ housing\_data$sq\_ft\_lot,

data = housing\_data)

Residuals:

Min 1Q Median 3Q Max

-2016064 -194842 -63293 91565 3735109

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) 6.418e+05 3.800e+03 168.90 <2e-16 \*\*\*

housing\_data$sq\_ft\_lot 8.510e-01 6.217e-02 13.69 <2e-16 \*\*\*

---

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

Residual standard error: 401500 on 12863 degrees of freedom

Multiple R-squared: 0.01435, Adjusted R-squared: 0.01428

F-statistic: 187.3 on 1 and 12863 DF, p-value: < 2.2e-16

> summary(sales\_price\_with\_others)

Call:

lm(formula = housing\_data$"Sale Price" ~ housing\_data$sq\_ft\_lot +

housing\_data$square\_feet\_total\_living + housing\_data$building\_grade,

data = housing\_data)

Residuals:

Min 1Q Median 3Q Max

-1943318 -115934 -42965 39043 3747512

Coefficients:

Estimate Std. Error t value

(Intercept) -8.142e+04 2.810e+04 -2.898

housing\_data$sq\_ft\_lot 1.330e-01 5.723e-02 2.324

housing\_data$square\_feet\_total\_living 1.475e+02 4.889e+00 30.178

housing\_data$building\_grade 4.424e+04 4.348e+03 10.175

Pr(>|t|)

(Intercept) 0.00377 \*\*

housing\_data$sq\_ft\_lot 0.02016 \*

housing\_data$square\_feet\_total\_living < 2e-16 \*\*\*

housing\_data$building\_grade < 2e-16 \*\*\*

---

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

Residual standard error: 358700 on 12861 degrees of freedom

Multiple R-squared: 0.2132, Adjusted R-squared: 0.213

F-statistic: 1161 on 3 and 12861 DF, p-value: < 2.2e-16

##d. Considering the parameters of the multiple regression model you have created. What are the standardized betas for each parameter and what do the values indicate?

##Response:

The standardized beta estimates explains the number of standard deviations by which the outcome will change as a result of one standard deviation change in the predictor. The larger the beta, the stronger the effect.

In this case, 1 standard deviation of change in Sq\_ft\_lot causes sales price to change by 0.019 standard deviation, and for square\_feet\_total\_living causes sales price to change by 0.361 standard deviation and for building\_grade can cause 0.120 standard deviation change in sale price.

```{r echo=FALSE}

lm.beta(sales\_price\_with\_others)

```

Output:

> lm.beta(sales\_price\_with\_others)

housing\_data$sq\_ft\_lot

0.01872256

housing\_data$square\_feet\_total\_living

0.36113656

housing\_data$building\_grade

0.11952538

##e. Calculate the confidence intervals for the parameters in your model and explain what the results indicate.

##Response:

Confidence interval shows that there is positive relation between all predictors and outcome. This helps to calculate how many data points may fall within a certain range when attempting to use the linear model. Assumption 90% of our data to fall within those ranges.

```{r echo=FALSE}

confint(sales\_price\_with\_others, level = 0.90)

```

> confint(sales\_price\_with\_others, level = 0.90)

5 % 95 %

(Intercept) -1.276441e+05 -3.519928e+04

housing\_data$sq\_ft\_lot 3.883638e-02 2.271258e-01

housing\_data$square\_feet\_total\_living 1.394970e+02 1.555812e+02

housing\_data$building\_grade 3.708465e+04 5.138827e+04

##f. Assess the improvement of the new model compared to your original model (simple regression model) by testing whether this change is significant by performing an analysis of variance.

##Response:

The value in column labeled Pr(>F) is 2.2e−16 (i.e., 2.2 with the decimal

place moved 16 places to the left, or a very small value indeed); we can say that sales\_price\_with\_others significantly improved the fit of the model to the data compared to sales\_price\_with\_sq\_ft\_lot.

```{r echo=FALSE}

anova(sales\_price\_with\_sq\_ft\_lot)

aov(sales\_price\_with\_sq\_ft\_lot)

anova(sales\_price\_with\_others)

aov(sales\_price\_with\_others)

```

Output:

> anova(sales\_price\_with\_sq\_ft\_lot)

Analysis of Variance Table

Response: housing\_data$"Sale Price"

Df Sum Sq Mean Sq F value Pr(>F)

housing\_data$sq\_ft\_lot 1 3.0197e+13 3.0197e+13 187.34 < 2.2e-16 \*\*\*

Residuals 12863 2.0734e+15 1.6119e+11

---

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

> aov(sales\_price\_with\_sq\_ft\_lot)

Call:

aov(formula = sales\_price\_with\_sq\_ft\_lot)

Terms:

housing\_data$sq\_ft\_lot Residuals

Sum of Squares 3.019674e+13 2.073377e+15

Deg. of Freedom 1 12863

Residual standard error: 401483.8

Estimated effects may be unbalanced

> anova(sales\_price\_with\_others)

Analysis of Variance Table

Response: housing\_data$"Sale Price"

Df Sum Sq Mean Sq F value

housing\_data$sq\_ft\_lot 1 3.0197e+13 3.0197e+13 234.64

housing\_data$square\_feet\_total\_living 1 4.0491e+14 4.0491e+14 3146.24

housing\_data$building\_grade 1 1.3323e+13 1.3323e+13 103.53

Residuals 12861 1.6551e+15 1.2870e+11

Pr(>F)

housing\_data$sq\_ft\_lot < 2.2e-16 \*\*\*

housing\_data$square\_feet\_total\_living < 2.2e-16 \*\*\*

housing\_data$building\_grade < 2.2e-16 \*\*\*

Residuals

---

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

> aov(sales\_price\_with\_others)

Call:

aov(formula = sales\_price\_with\_others)

Terms:

housing\_data$sq\_ft\_lot

Sum of Squares 3.019674e+13

Deg. of Freedom 1

housing\_data$square\_feet\_total\_living

Sum of Squares 4.049056e+14

Deg. of Freedom 1

housing\_data$building\_grade Residuals

Sum of Squares 1.332319e+13 1.655148e+15

Deg. of Freedom 1 12861

Residual standard error: 358741

Estimated effects may be unbalanced

##g. Perform casewise diagnostics to identify outliers and/or influential cases, storing each function's output in a dataframe assigned to a unique variable name.

```{r echo=FALSE}

housing\_data$standardized\_residuals<- rstandard(sales\_price\_with\_others)

head(housing\_data$standardized\_residuals)

housing\_data$studentized\_residuals<-rstudent(sales\_price\_with\_others)

head(housing\_data$studentized\_residuals)

housing\_data$dfbeta<-dfbeta(sales\_price\_with\_others)

head(housing\_data$dfbeta)

housing\_data$dffit<-dffits(sales\_price\_with\_others)

head(housing\_data$dffit)

housing\_data$leverage<-hatvalues(sales\_price\_with\_others)

head(housing\_data$leverage)

```

Output:

> housing\_data$standardized\_residuals<- rstandard(sales\_price\_with\_others)

> head(housing\_data$standardized\_residuals)

1 2 3 4 5 6

-0.09526631 -0.25749989 -0.30602206 -0.25859457 -0.20013735 -1.83622727

> housing\_data$studentized\_residuals<-rstudent(sales\_price\_with\_others)

> head(housing\_data$studentized\_residuals)

1 2 3 4 5 6

-0.09526264 -0.25749054 -0.30601128 -0.25858519 -0.20012988 -1.83639662

> housing\_data$dfbeta<-dfbeta(sales\_price\_with\_others)

> head(housing\_data$dfbeta)

(Intercept) housing\_data$sq\_ft\_lot housing\_data$square\_feet\_total\_living

1 15.53348 1.476123e-05 0.001227376

2 38.42380 4.519048e-05 0.002074116

3 -46.69060 5.226207e-05 -0.008579373

4 18.77844 -4.558569e-06 0.012611383

5 -43.25320 2.403663e-06 0.003317644

6 -1375.07195 7.990754e-04 -0.302595677

housing\_data$building\_grade

1 -2.625498

2 -6.295360

3 7.133413

4 -7.028207

5 3.542687

6 251.748864

> housing\_data$dffit<-dffits(sales\_price\_with\_others)

> head(housing\_data$dffit)

1 2 3 4 5

-0.001099111 -0.002935992 -0.003323269 -0.003506871 -0.002760488

6

-0.066598878

> housing\_data$leverage<-hatvalues(sales\_price\_with\_others)

> head(housing\_data$leverage)

1 2 3 4 5

0.0001331006 0.0001299962 0.0001179247 0.0001838876 0.0001902239

6

0.0013134999

##h. Calculate the standardized residuals using the appropriate command, specifying those that are +-2, storing the results of large residuals in a variable you create.

```{r echo=FALSE}

housing\_data$large\_residual <- housing\_data$studentized\_residuals > 2 | housing\_data$studentized\_residuals < -2

```

##i. Use the appropriate function to show the sum of large residuals.

```{r echo=FALSE}

sum(housing\_data$large\_residual)

```

Output:

> sum(housing\_data$large\_residual)

[1] 317

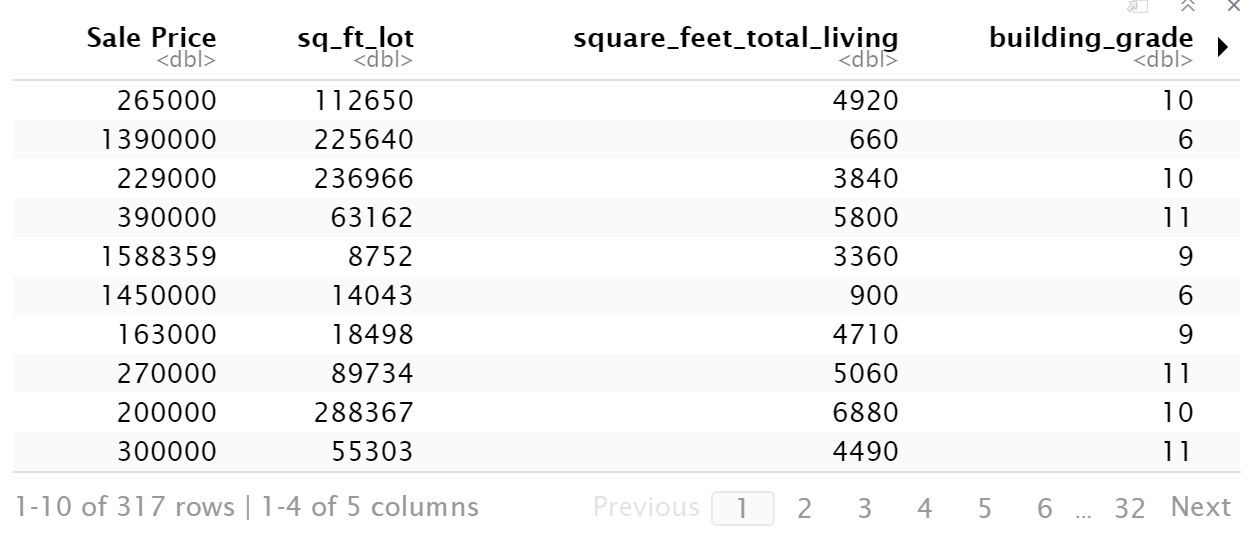
##j. Which specific variables have large residuals (only cases that evaluate as TRUE)?

```{r echo=FALSE}

housing\_data[housing\_data$large\_residual, c('Sale Price', 'sq\_ft\_lot', 'square\_feet\_total\_living', 'building\_grade', "standardized\_residuals")]

```

Output:



##k. Investigate further by calculating the leverage, cooks distance, and covariance rations. Comment on all cases that are problematics.

##Response:

Out of 317 large residuals one value has cook distance greater than 1. Hence only 1 influential observation in the model. Also couldn't find any problematic cases as per the data.

```{r echo=FALSE}

housing\_data[housing\_data$large\_residual, c("cooks\_distance", "leverage", "covariance\_ratios")]

cook\_dist <- housing\_data$cooks\_distance > 1

leverage <- housing\_data$leverage

covariance\_ratios <- housing\_data$covariance\_ratios

covariance\_ratios

sum(cook\_dist)

sum(leverage)

sum(covariance\_ratios)

```

##l. Perform the necessary calculations to assess the assumption of independence and state if the condition is met or not.

##Response:

Definition for Durbin Watson test reports:

- 2 is no autocorrelation.

- '0 to <2' is positive autocorrelation (common in time series data).

- '>2 to 4' is negative autocorrelation (less common in time series data).

Based on the Durbin watson test we can see the DWT-statistic value for the model is 0.52. Which means there is positive autocorrelation and the condition is met.

```{r echo=FALSE}

library(car)

dwt(sales\_price\_with\_others)

dwtest(sales\_price\_with\_others)

```

Output:

> dwt(sales\_price\_with\_others)

lag Autocorrelation D-W Statistic p-value

1 0.7402678 0.5194617 0

Alternative hypothesis: rho != 0

> dwtest(sales\_price\_with\_others)

Durbin-Watson test

data: sales\_price\_with\_others

DW = 0.51946, p-value < 2.2e-16

alternative hypothesis: true autocorrelation is greater than 0

##m. Perform the necessary calculations to assess the assumption of no multicollinearity and state if the condition is met or not.

##Response:

VIF values are all well below 10 and the tolerance statistics all well above 0.2. Also, the average VIF is very close to 1. Based on these measures we can safely conclude that there is no collinearity within our data.

```{r echo=FALSE}

vif(sales\_price\_with\_others)

mean(vif(sales\_price\_with\_others))

```

Output:

> vif(sales\_price\_with\_others)

housing\_data$sq\_ft\_lot

1.061258

housing\_data$square\_feet\_total\_living

2.340696

housing\_data$building\_grade

2.255639

> mean(vif(sales\_price\_with\_others))

[1] 1.885864

##n. Visually check the assumptions related to the residuals using the plot() and hist() functions. Summarize what each graph is informing you of and if any anomalies are present.

Residuals vs fitted plot shows a fairly random pattern, which means the assumptions of linearity, randomness and homoscedasticity have been met. The Q-Q plot shows less normality but this can happen due to smaller sample than the population.

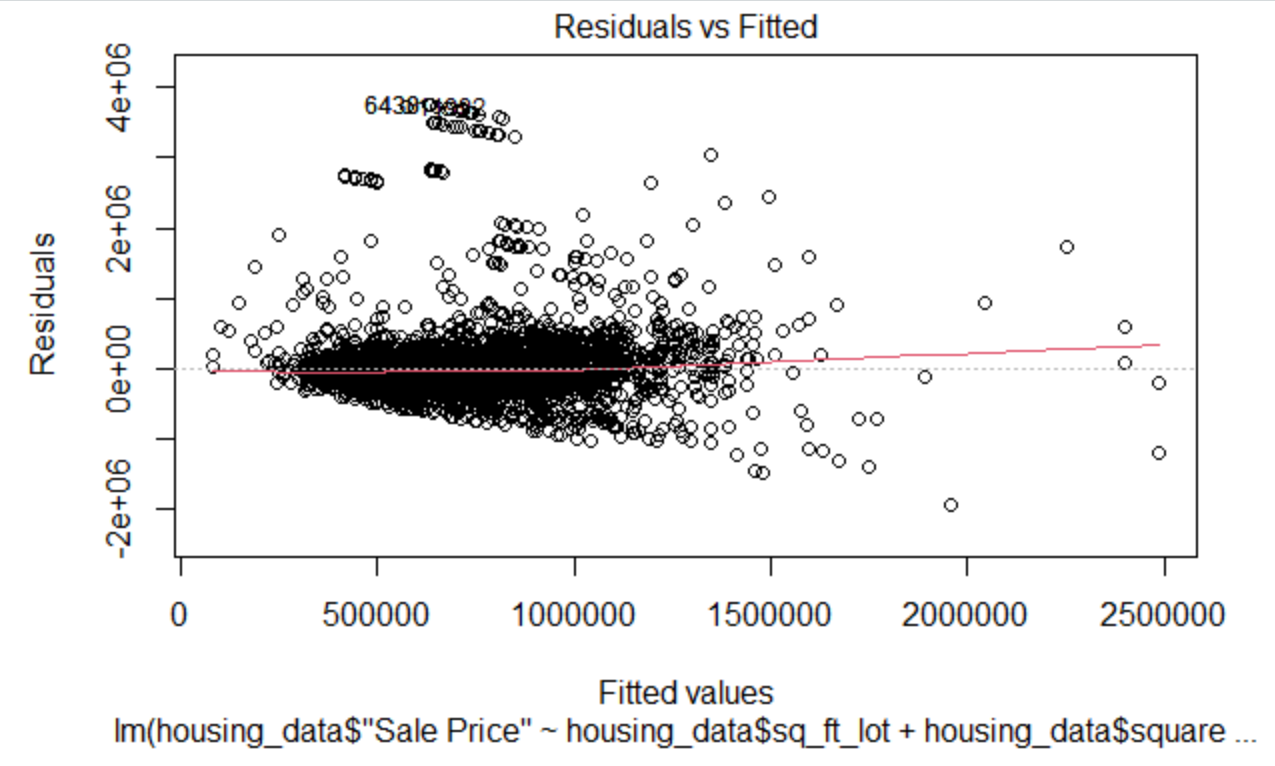
```{r echo=FALSE}

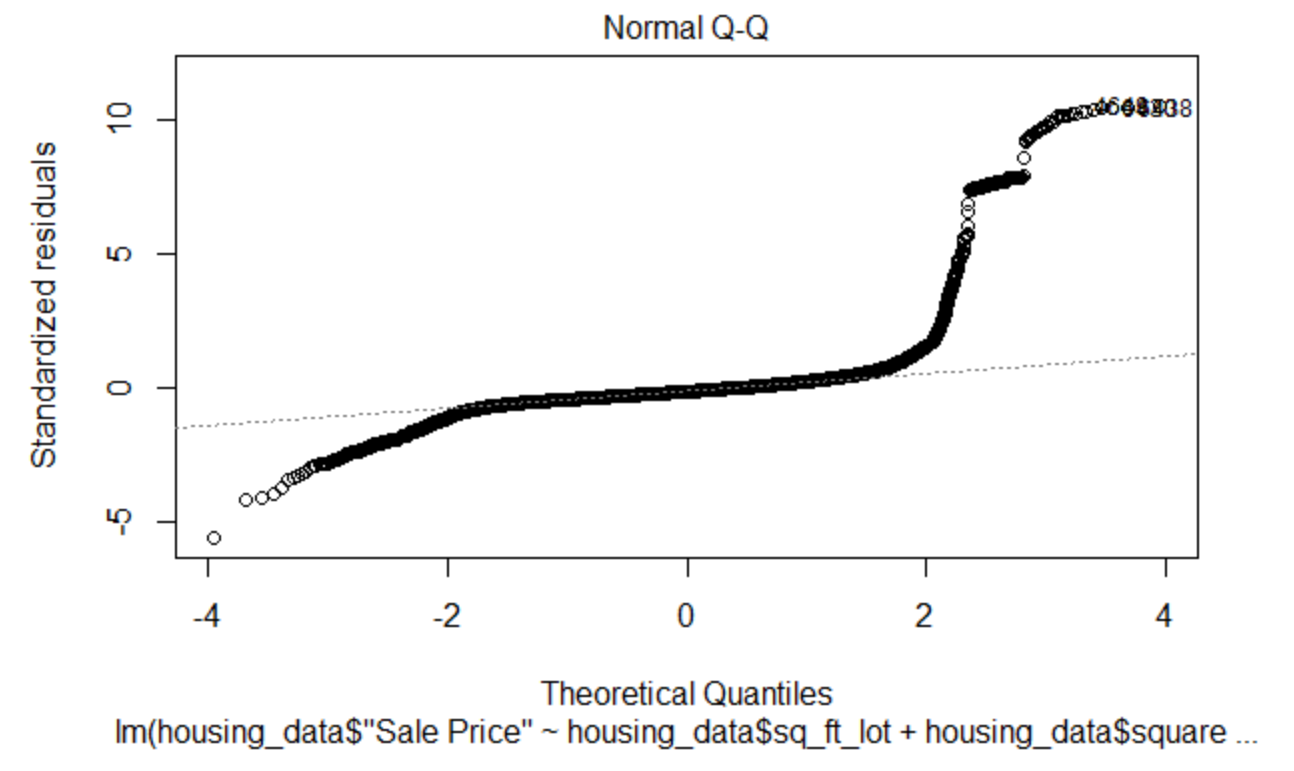
plot(sales\_price\_with\_others)

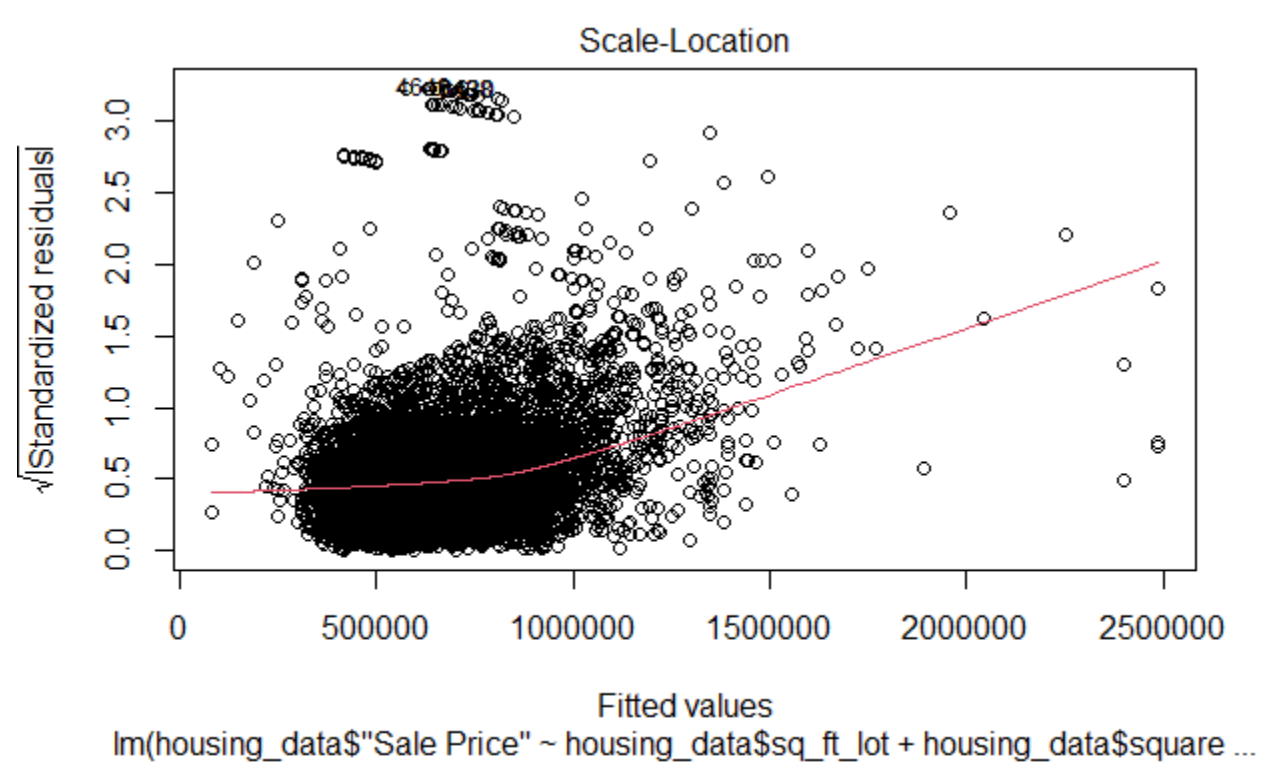
hist(housing\_data$studentized\_residuals)

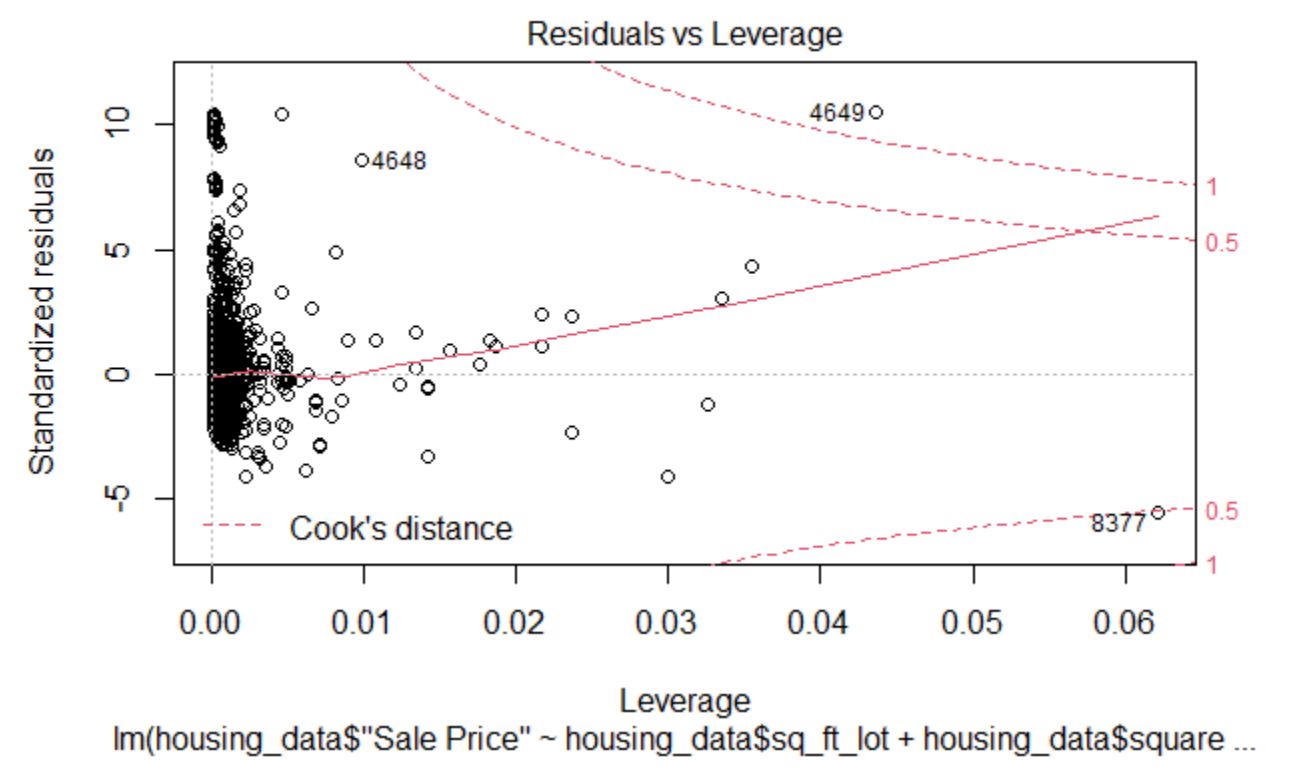
```

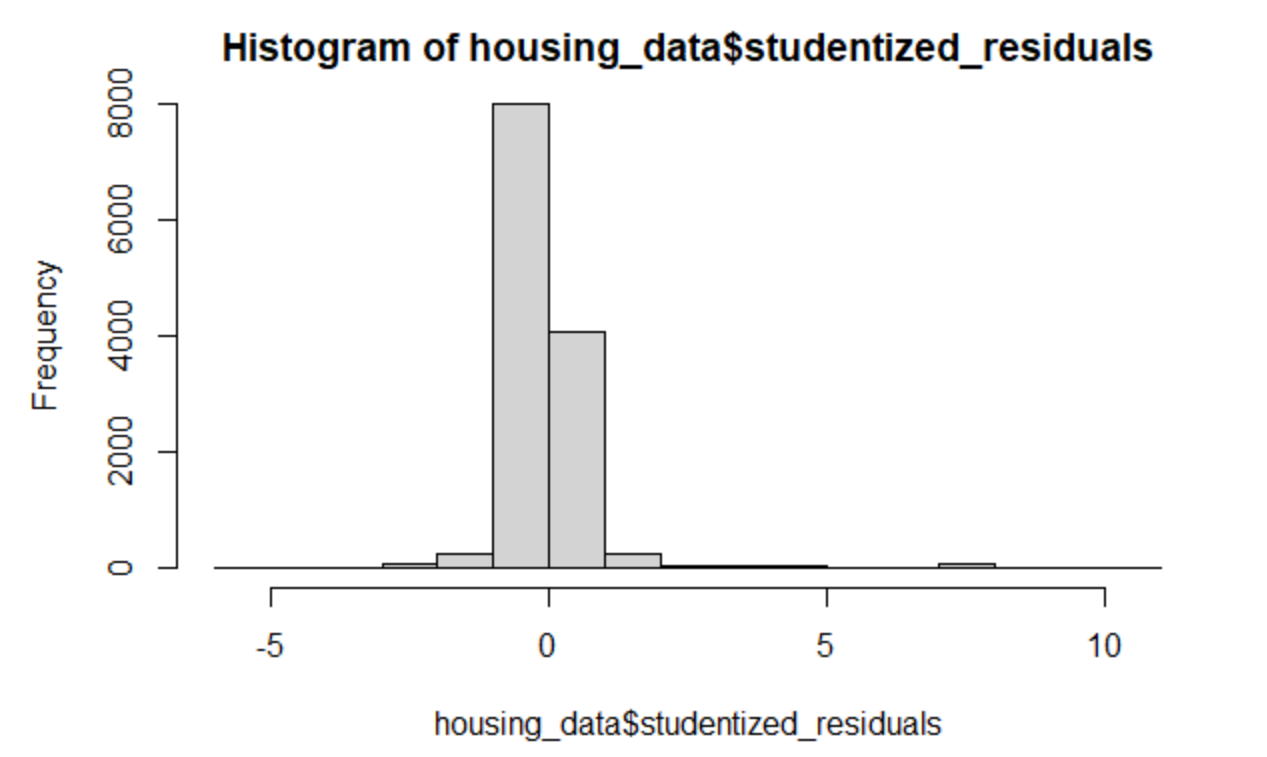
Output:











##0. Overall, is this regression model unbiased? If an unbiased regression model, what does this tell us about the sample vs. the entire population model?

##Response:

Based on the above plots looks like a normal distribution, we can say that the model is both accurate and generalized to the population. Overall relatively unbiased.

## References

Discovering Statistics Using R

R Markdown basics

R for Everyone. Pearson Education, 2017

https://rstudio.com/wp-content/uploads/2016/03/rmarkdown-cheatsheet-2.0.pdf?\_ga=2.247973831.1388722509.1600630414-1304384236.1598891840

Slake for students feedback and reference comments.

Few points captured from the GitHub files of classmates.