Week 8 Assignments

Alan Donahue

7/31/2021

# Assignment 6

## Set the working directory to the root of your DSC 520 directory  
setwd("C:/Users/Alan Donahue/Documents/data science masters/DSC 520 Stats/GIT/dsc520")  
  
## Load the `data/r4ds/heights.csv` to  
heights\_df <- read.csv("data/r4ds/heights.csv")  
  
## Load the ggplot2 library  
library(ggplot2)  
  
## Fit a linear model using the `age` variable as the predictor and `earn` as the outcome  
age\_lm <- lm(earn ~ age, data = heights\_df)  
  
## View the summary of your model using `summary()`  
summary(age\_lm)

##   
## Call:  
## lm(formula = earn ~ age, data = heights\_df)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -25098 -12622 -3667 6883 177579   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 19041.53 1571.26 12.119 < 2e-16 \*\*\*  
## age 99.41 35.46 2.804 0.00514 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 19420 on 1190 degrees of freedom  
## Multiple R-squared: 0.006561, Adjusted R-squared: 0.005727   
## F-statistic: 7.86 on 1 and 1190 DF, p-value: 0.005137

## Creating predictions using `predict()`  
age\_predict\_df <- data.frame(earn = predict(age\_lm, heights\_df), age=heights\_df$age)  
  
## Plot the predictions against the original data  
## NEED TO FIX!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!  
#ggplot(data = heights\_df, aes(y = age\_predict\_df, x = age\_lm)) +  
 #geom\_point(color='blue') +  
 #geom\_line(color='red',data = heights\_df, aes(y=age\_predict\_df, x=age\_lm))  
  
mean\_earn <- mean(heights\_df$earn)  
## Corrected Sum of Squares Total  
sst <- sum((mean\_earn - heights\_df$earn)^2)  
## Corrected Sum of Squares for Model  
ssm <- sum((mean\_earn - age\_predict\_df$earn)^2)  
## Residuals  
residuals <- heights\_df$earn - age\_predict\_df$earn  
## Sum of Squares for Error  
sse <- sum(residuals^2)  
## R Squared R^2 = SSM\SST  
r\_squared <- ssm / sst  
  
## Number of observations  
n <- 1191  
## Number of regression parameters  
p <- 2  
## Corrected Degrees of Freedom for Model (p-1)  
dfm <- p - 1  
## Degrees of Freedom for Error (n-p)  
dfe <- n - p  
## Corrected Degrees of Freedom Total: DFT = n - 1  
dft <- n - 1  
  
## Mean of Squares for Model: MSM = SSM / DFM  
msm <- ssm / dfm  
## Mean of Squares for Error: MSE = SSE / DFE  
mse <- sse / dfe  
## Mean of Squares Total: MST = SST / DFT  
mst <- sst / dft  
## F Statistic F = MSM/MSE  
f\_score <- msm / mse  
  
## Adjusted R Squared R2 = 1 - (1 - R2)(n - 1) / (n - p)  
adjusted\_r\_squared <- 1 - (1 - r\_squared)\*(n - 1) / (n - p)  
  
## Calculate the p-value from the F distribution  
p\_value <- pf(f\_score, dfm, dft, lower.tail=F)

# Assignment 7

## Set the working directory to the root of your DSC 520 directory  
setwd("C:/Users/Alan Donahue/Documents/data science masters/DSC 520 Stats/GIT/dsc520")  
  
## Load the `data/r4ds/heights.csv` to  
heights\_df <- read.csv("data/r4ds/heights.csv")  
  
# Fit a linear model  
earn\_lm <- lm(earn ~ age + height + sex + ed + race, data=heights\_df)  
  
# View the summary of your model  
summary(earn\_lm)

##   
## Call:  
## lm(formula = earn ~ age + height + sex + ed + race, data = heights\_df)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -39423 -9827 -2208 6157 158723   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -41478.4 12409.4 -3.342 0.000856 \*\*\*  
## age 178.3 32.2 5.537 3.78e-08 \*\*\*  
## height 202.5 185.6 1.091 0.275420   
## sexmale 10325.6 1424.5 7.249 7.57e-13 \*\*\*  
## ed 2768.4 209.9 13.190 < 2e-16 \*\*\*  
## racehispanic -1414.3 2685.2 -0.527 0.598507   
## raceother 371.0 3837.0 0.097 0.922983   
## racewhite 2432.5 1723.9 1.411 0.158489   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 17250 on 1184 degrees of freedom  
## Multiple R-squared: 0.2199, Adjusted R-squared: 0.2153   
## F-statistic: 47.68 on 7 and 1184 DF, p-value: < 2.2e-16

predicted\_df <- data.frame(  
 earn = predict(earn\_lm, heights\_df),  
 ed=heights\_df$ed, race=heights\_df$race, height=heights\_df$height,  
 age=heights\_df$age, sex=heights\_df$sex  
 )  
  
## Compute deviation (i.e. residuals)  
mean\_earn <- mean(heights\_df$earn)  
## Corrected Sum of Squares Total  
sst <- sum((mean\_earn - heights\_df$earn)^2)  
## Corrected Sum of Squares for Model  
ssm <- sum((mean\_earn - predicted\_df$earn)^2)  
## Residuals  
residuals <- heights\_df$earn - predicted\_df$earn  
## Sum of Squares for Error  
sse <- sum(residuals^2)  
## R Squared  
r\_squared <- ssm / sst  
  
## Number of observations  
n <- 1191  
## Number of regression paramaters  
p <- 8  
## Corrected Degrees of Freedom for Model  
dfm <- p - 1  
## Degrees of Freedom for Error  
dfe <- n - p  
## Corrected Degrees of Freedom Total: DFT = n - 1  
dft <- n - 1  
  
## Mean of Squares for Model: MSM = SSM / DFM  
msm <- ssm / dfm  
## Mean of Squares for Error: MSE = SSE / DFE  
mse <- sse / dfe  
## Mean of Squares Total: MST = SST / DFT  
mst <- sst / dft  
## F Statistic  
f\_score <- msm / mse  
  
## Adjusted R Squared R2 = 1 - (1 - R2)(n - 1) / (n - p)  
adjusted\_r\_squared <- 1 - (1-r\_squared)\*(n - 1) / (n - p)

# Housing Data

## Question 1

library(car)

## Loading required package: carData

#setting the working directory  
setwd("C:/Users/Alan Donahue/Documents/data science masters/DSC 520 Stats/GIT/dsc520")  
  
#load the data  
housing\_df <- read.csv("data/week-6-housing.csv")  
  
summary(housing\_df)

## Sale.Date Sale.Price sale\_reason sale\_instrument   
## Length:12865 Min. : 698 Min. : 0.00 Min. : 0.000   
## Class :character 1st Qu.: 460000 1st Qu.: 1.00 1st Qu.: 3.000   
## Mode :character Median : 593000 Median : 1.00 Median : 3.000   
## Mean : 660738 Mean : 1.55 Mean : 3.678   
## 3rd Qu.: 750000 3rd Qu.: 1.00 3rd Qu.: 3.000   
## Max. :4400000 Max. :19.00 Max. :27.000   
## sale\_warning sitetype addr\_full zip5   
## Length:12865 Length:12865 Length:12865 Min. :98052   
## Class :character Class :character Class :character 1st Qu.:98052   
## Mode :character Mode :character Mode :character Median :98052   
## Mean :98053   
## 3rd Qu.:98053   
## Max. :98074   
## ctyname postalctyn lon lat   
## Length:12865 Length:12865 Min. :-122.2 Min. :47.46   
## Class :character Class :character 1st Qu.:-122.1 1st Qu.:47.67   
## Mode :character Mode :character Median :-122.1 Median :47.69   
## Mean :-122.1 Mean :47.68   
## 3rd Qu.:-122.0 3rd Qu.:47.70   
## Max. :-121.9 Max. :47.73   
## building\_grade square\_feet\_total\_living bedrooms bath\_full\_count   
## Min. : 2.00 Min. : 240 Min. : 0.000 Min. : 0.000   
## 1st Qu.: 8.00 1st Qu.: 1820 1st Qu.: 3.000 1st Qu.: 1.000   
## Median : 8.00 Median : 2420 Median : 4.000 Median : 2.000   
## Mean : 8.24 Mean : 2540 Mean : 3.479 Mean : 1.798   
## 3rd Qu.: 9.00 3rd Qu.: 3110 3rd Qu.: 4.000 3rd Qu.: 2.000   
## Max. :13.00 Max. :13540 Max. :11.000 Max. :23.000   
## bath\_half\_count bath\_3qtr\_count year\_built year\_renovated   
## Min. :0.0000 Min. :0.000 Min. :1900 Min. : 0.00   
## 1st Qu.:0.0000 1st Qu.:0.000 1st Qu.:1979 1st Qu.: 0.00   
## Median :1.0000 Median :0.000 Median :1998 Median : 0.00   
## Mean :0.6134 Mean :0.494 Mean :1993 Mean : 26.24   
## 3rd Qu.:1.0000 3rd Qu.:1.000 3rd Qu.:2007 3rd Qu.: 0.00   
## Max. :8.0000 Max. :8.000 Max. :2016 Max. :2016.00   
## current\_zoning sq\_ft\_lot prop\_type present\_use   
## Length:12865 Min. : 785 Length:12865 Min. : 0.000   
## Class :character 1st Qu.: 5355 Class :character 1st Qu.: 2.000   
## Mode :character Median : 7965 Mode :character Median : 2.000   
## Mean : 22229 Mean : 6.598   
## 3rd Qu.: 12632 3rd Qu.: 2.000   
## Max. :1631322 Max. :300.000

#change names for consistency  
colnames(housing\_df)[1] <- "Sale\_Date"  
colnames(housing\_df)[2] <- "Sale\_Price"  
  
head(housing\_df$sale\_warning)

## [1] "" "" "" "" "15" "18 51"

#change "" to 0 in sale\_warning  
housing\_df$sale\_warning <- sub("^$", 0, housing\_df$sale\_warning)  
head(housing\_df$sale\_warning)

## [1] "0" "0" "0" "0" "15" "18 51"

#change "" to NA in ctyname  
unique(is.na(housing\_df$ctyname))

## [1] FALSE

housing\_df$ctyname <- sub("^$", NA, housing\_df$ctyname)  
unique(is.na(housing\_df$ctyname))

## [1] FALSE TRUE

I completed a few changes to the data set. First, I converted it to a .csv file. I also changed the names of the first two columns for consistency sake. From there, I changed any "" in sale\_warning to a 0 to represent FALSE. Finally, I couldn’t find a quick way to change all the "" in ctyname and confirm that they were accurate so I changed the "" to NA.

## Question 2

#check to make sure no ""  
unique(housing\_df$building\_grade)

## [1] 9 8 7 10 6 11 12 5 4 13 2 3

unique(housing\_df$year\_built)

## [1] 2003 2006 1987 1968 1980 2005 1993 1988 1978 1976 1975 2011 1990 1972 1977  
## [16] 1986 2007 1998 1979 1966 1983 1970 1991 1999 1973 1964 2002 1963 1984 1989  
## [31] 2004 1992 1985 1981 1967 2000 2001 1952 1955 1995 1942 2008 2014 1974 1994  
## [46] 1900 1969 2015 1957 1918 1953 1982 1965 2016 1997 1996 1958 1971 2013 1954  
## [61] 2010 1959 1950 1961 1913 1951 1933 1930 1947 1914 1943 1946 1905 1948 2012  
## [76] 1929 1920 1960 1962 2009 1922 1903 1956 1941 1940 1938 1926 1927 1949 1939  
## [91] 1944 1923 1924 1925 1937 1945 1934 1935 1909 1932 1912 1931 1916 1906 1936  
## [106] 1928 1915 1919 1910

var3 <- housing\_df[housing\_df$sq\_ft\_lot=="", ]  
print(var3)

## [1] Sale\_Date Sale\_Price sale\_reason   
## [4] sale\_instrument sale\_warning sitetype   
## [7] addr\_full zip5 ctyname   
## [10] postalctyn lon lat   
## [13] building\_grade square\_feet\_total\_living bedrooms   
## [16] bath\_full\_count bath\_half\_count bath\_3qtr\_count   
## [19] year\_built year\_renovated current\_zoning   
## [22] sq\_ft\_lot prop\_type present\_use   
## <0 rows> (or 0-length row.names)

#Question 2   
sq\_ft\_lot\_lm <- lm(Sale\_Price ~ sq\_ft\_lot, data=housing\_df, na.action = na.omit)  
mult\_pred\_lm <- lm(Sale\_Price ~ sq\_ft\_lot + building\_grade + year\_built, data = housing\_df, na.action = na.omit)

I chose my additional predictors based off the theoretical importance they have in regard to buying or selling a house.I felt that the quality of the build (building\_grade) was important to the sale price because a cheaply made house should go for less money. I felt when the house was built was important to the sale price because an older house made need more repairs than a new house.

## Question 3

summary(sq\_ft\_lot\_lm)

##   
## Call:  
## lm(formula = Sale\_Price ~ sq\_ft\_lot, data = housing\_df, na.action = na.omit)  
##   
## 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 \*\*\*  
## 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(mult\_pred\_lm)

##   
## Call:  
## lm(formula = Sale\_Price ~ sq\_ft\_lot + building\_grade + year\_built,   
## data = housing\_df, na.action = na.omit)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2172605 -137008 -44312 54092 3706547   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -6.724e+06 4.016e+05 -16.74 <2e-16 \*\*\*  
## sq\_ft\_lot 6.577e-01 5.870e-02 11.20 <2e-16 \*\*\*  
## building\_grade 1.218e+05 3.251e+03 37.47 <2e-16 \*\*\*  
## year\_built 3.194e+03 2.062e+02 15.49 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 367800 on 12861 degrees of freedom  
## Multiple R-squared: 0.1729, Adjusted R-squared: 0.1727   
## F-statistic: 896.1 on 3 and 12861 DF, p-value: < 2.2e-16

The R2 and Adjusted R2 for the simple regression are .01435 and .01428 respectfully. The R2 value tells us that the sq\_ft\_lot accounts for 1.44% of the variation in sale price. The Adjusted R2 tells us how well our model generalizes. We want the value to be close to the value of R2. It tells us that there is a .007% in shrinkage which means it would generalize well.

The R2 and Adjusted R2 for the multiple regression are .1729 and .1727 respectfully. The R2 value tells us that the multple predictors account for 17.29% of variance in sale price which means it covers more compared to just sq\_ft\_lot. The Adjusted R2 tells us there is a .02% in shrinkage which means it would generalize well.

## Question 4

For the simple regression, the b0 = 6.418e+05 and b1 = 8.510e-01. This tells me that when X = 0, the sale price is going to be $641,800 and for every sq\_ft\_lot added, the price will go up by .851.

For the multiple regression, b0 = -6.724e+06, b1 = 6.577e-01, b2 = 1.218e+05, and b3 = 3.194e+03. The negative intercept is a cause for concern. It means that there there might be an issue with the assumption of linearity. The other coefficients show how much the sale price would go up if one sq\_ft\_lot or the build grade went up or there was a change in year the house was built.

## Question 5

confint(sq\_ft\_lot\_lm)

## 2.5 % 97.5 %  
## (Intercept) 6.343730e+05 6.492698e+05  
## sq\_ft\_lot 7.291208e-01 9.728641e-01

confint(mult\_pred\_lm)

## 2.5 % 97.5 %  
## (Intercept) -7.511117e+06 -5.936765e+06  
## sq\_ft\_lot 5.426072e-01 7.727464e-01  
## building\_grade 1.154567e+05 1.282031e+05  
## year\_built 2.790059e+03 3.598422e+03

The confidence intervals are stating that in 95% of these samples it will contain the b that represents the population. The confidence intervals look good because they don’t cross zero and the two ends are close to each other for each interval.

## Question 6

anova(sq\_ft\_lot\_lm, mult\_pred\_lm)

## Analysis of Variance Table  
##   
## Model 1: Sale\_Price ~ sq\_ft\_lot  
## Model 2: Sale\_Price ~ sq\_ft\_lot + building\_grade + year\_built  
## Res.Df RSS Df Sum of Sq F Pr(>F)   
## 1 12863 2.0734e+15   
## 2 12861 1.7399e+15 2 3.3349e+14 1232.5 < 2.2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

We can say it has significantly improved.

## Question 7

housing\_df$residuals <- resid(mult\_pred\_lm)  
housing\_df$standardized.residuals <- rstandard(mult\_pred\_lm)  
housing\_df$studentized.residuals <- rstudent(mult\_pred\_lm)  
housing\_df$cooks.distance <- cooks.distance(mult\_pred\_lm)  
housing\_df$dfbeta <- dfbeta(mult\_pred\_lm)  
housing\_df$dffit <- dffits(mult\_pred\_lm)  
housing\_df$leverage <- hatvalues(mult\_pred\_lm)  
housing\_df$covariance.ratios <- covratio(mult\_pred\_lm)

## Question 8

head(housing\_df$standardized.residuals > 2 | housing\_df$standardized.residuals < -2)

## [1] FALSE FALSE FALSE FALSE FALSE FALSE

housing\_df$large.residuals <- housing\_df$standardized.residuals > 2 | housing\_df$standardized.residuals < -2  
head(housing\_df$large.residuals)

## [1] FALSE FALSE FALSE FALSE FALSE FALSE

## Question 9

sum(housing\_df$large.residuals)

## [1] 327

percent.large <- (sum(housing\_df$large.residuals)/nrow(housing\_df)) \* 100  
print(head(percent.large))

## [1] 2.54178

## Question 10

first.results <- housing\_df[housing\_df$large.residuals, c("Sale\_Price", "sq\_ft\_lot", "building\_grade", "year\_built", "standardized.residuals")]  
head(first.results)

## Sale\_Price sq\_ft\_lot building\_grade year\_built standardized.residuals  
## 14 165000 278891 9 2011 -2.216749  
## 72 1900000 37017 11 1990 2.455721  
## 108 1520000 19173 9 1952 2.447062  
## 115 1390000 225640 6 1955 2.693326  
## 160 229000 236966 10 2008 -2.272395  
## 239 1588359 8752 9 2005 2.190472

## Question 11

second.results <- housing\_df[housing\_df$large.residuals, c("cooks.distance", "leverage", "covariance.ratios")]  
head(second.results)

## cooks.distance leverage covariance.ratios  
## 14 0.002343891 0.0019043048 1.0006888  
## 72 0.001038161 0.0006881257 0.9991237  
## 108 0.001164074 0.0007769857 0.9992258  
## 115 0.002968210 0.0016340540 0.9996897  
## 160 0.001761850 0.0013629149 1.0000685  
## 239 0.000165206 0.0001377053 0.9989567

#cooks distance greater than 1 = concern  
housing\_df[housing\_df$cooks.distance > 1, ]

## [1] Sale\_Date Sale\_Price sale\_reason   
## [4] sale\_instrument sale\_warning sitetype   
## [7] addr\_full zip5 ctyname   
## [10] postalctyn lon lat   
## [13] building\_grade square\_feet\_total\_living bedrooms   
## [16] bath\_full\_count bath\_half\_count bath\_3qtr\_count   
## [19] year\_built year\_renovated current\_zoning   
## [22] sq\_ft\_lot prop\_type present\_use   
## [25] residuals standardized.residuals studentized.residuals   
## [28] cooks.distance dfbeta dffit   
## [31] leverage covariance.ratios large.residuals   
## <0 rows> (or 0-length row.names)

twice.leverage = 2 \* ((3+1) / 12864)  
three.leverage = 3 \* ((3+1) / 12864)  
  
third.results <- housing\_df[housing\_df$leverage > three.leverage, ]  
head(third.results)

## Sale\_Date Sale\_Price sale\_reason sale\_instrument sale\_warning sitetype  
## 14 1/4/2006 165000 1 3 0 R1  
## 65 1/26/2006 446400 8 3 12 R1  
## 115 2/15/2006 1390000 1 3 0 R1  
## 116 2/15/2006 1390000 1 3 0 R1  
## 131 2/21/2006 650000 1 3 0 R1  
## 160 2/27/2006 229000 18 3 13 R1  
## addr\_full zip5 ctyname postalctyn lon lat  
## 14 2921 288TH AVE NE 98053 <NA> REDMOND -121.9577 47.63382  
## 65 28616 NE 47TH PL 98053 <NA> REDMOND -121.9569 47.65066  
## 115 19656 NE REDMOND RD 98053 <NA> REDMOND -122.0772 47.69595  
## 116 19656 NE REDMOND RD 98053 <NA> REDMOND -122.0772 47.69595  
## 131 26608 NE 15TH ST 98053 <NA> REDMOND -121.9831 47.62150  
## 160 28527 NE 47TH PL 98053 <NA> REDMOND -121.9580 47.64833  
## building\_grade square\_feet\_total\_living bedrooms bath\_full\_count  
## 14 9 1850 3 2  
## 65 7 1770 3 3  
## 115 6 660 0 1  
## 116 10 3280 3 2  
## 131 9 3960 4 2  
## 160 10 3840 0 0  
## bath\_half\_count bath\_3qtr\_count year\_built year\_renovated current\_zoning  
## 14 0 0 2011 0 RA5  
## 65 0 0 1984 0 RA5  
## 115 0 0 1955 0 RA5  
## 116 0 1 2000 0 RA5  
## 131 1 2 1995 0 RA5  
## 160 0 0 2008 0 RA5  
## sq\_ft\_lot prop\_type present\_use residuals standardized.residuals  
## 14 278891 R 2 -814566.1 -2.2167494  
## 65 220654 R 2 -164960.7 -0.4487772  
## 115 225640 R 2 989822.9 2.6933263  
## 116 225640 R 2 358762.6 0.9759815  
## 131 217800 R 2 -238280.2 -0.6481659  
## 160 236966 R 300 -835240.2 -2.2723953  
## studentized.residuals cooks.distance dfbeta.(Intercept) dfbeta.sq\_ft\_lot  
## 14 -2.2170868 2.343891e-03 1.404118e+04 -5.422020e-03  
## 65 -0.4487632 6.353258e-05 8.454823e+02 -8.668139e-04  
## 115 2.6939814 2.968210e-03 9.156804e+03 5.097091e-03  
## 116 0.9759797 2.832488e-04 -1.969672e+03 1.725497e-03  
## 131 -0.6481513 1.074077e-04 1.277533e+03 -1.145039e-03  
## 160 -2.2727633 1.761850e-03 8.995790e+03 -4.379147e-03  
## dfbeta.building\_grade dfbeta.year\_built dffit leverage  
## 14 3.970472e+01 -7.180759e+00 -0.09684224 0.001904305  
## 65 2.252595e+01 -5.141369e-01 -0.01594097 0.001260223  
## 115 -1.588384e+02 -3.955913e+00 0.10898906 0.001634054  
## 116 2.365842e+01 8.852379e-01 0.03365993 0.001188033  
## 131 1.009564e-01 -6.379585e-01 -0.02072707 0.001021596  
## 160 -3.951887e+01 -4.334068e+00 -0.08396238 0.001362915  
## covariance.ratios large.residuals  
## 14 1.0006888 TRUE  
## 65 1.0015105 FALSE  
## 115 0.9996897 TRUE  
## 116 1.0012042 FALSE  
## 131 1.0012032 FALSE  
## 160 1.0000685 TRUE

CVR.upper <- 1 + (3 \* (3+1)/12864)  
CVR.lower <- 1 - (3 \* (3+1)/12864)  
  
sum(housing\_df$covariance.ratios > CVR.upper | housing\_df$covariance.ratios < CVR.lower)

## [1] 755

There are no issues with Cook’s Distance. However, there are well over 430 cases that the average leverage is above three times the average leverage. Additionally, there are 755 cases where the covariance ratios are outside of the upper and lower limits. This is cause for major concern.

## Question 12

durbinWatsonTest(mult\_pred\_lm)

## lag Autocorrelation D-W Statistic p-value  
## 1 0.6982173 0.6035601 0  
## Alternative hypothesis: rho != 0

The assumption of independence has not been met after running the durbinWatsonTest() function. Anything less than 1 or greater than three is a cause for concern. The ideal scenario is to be as close to 2 as possible. The function returns 0.6035601 which is below 1.

## Question 13

vif(mult\_pred\_lm)

## sq\_ft\_lot building\_grade year\_built   
## 1.062196 1.200062 1.198887

1/vif(mult\_pred\_lm)

## sq\_ft\_lot building\_grade year\_built   
## 0.9414461 0.8332905 0.8341072

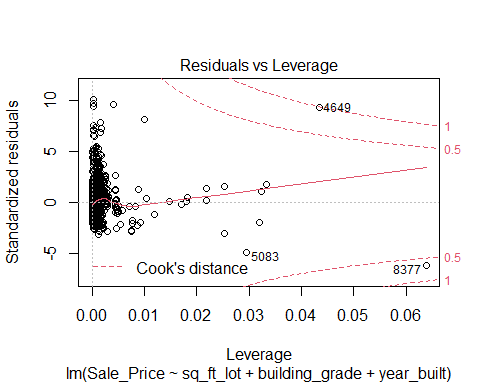
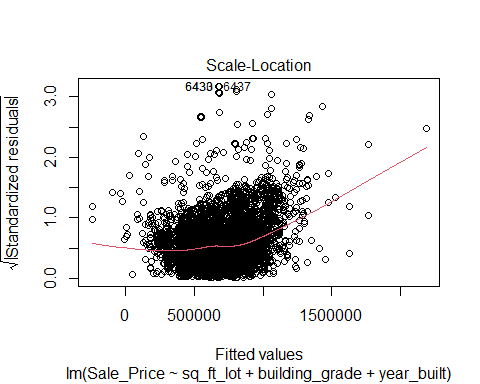
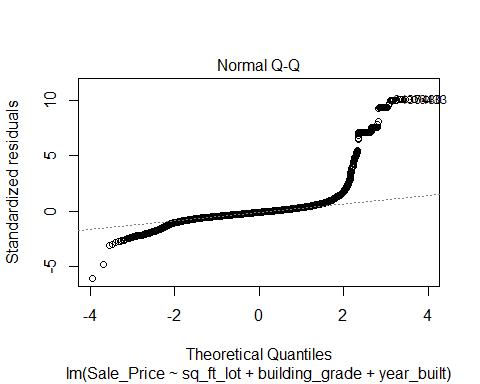
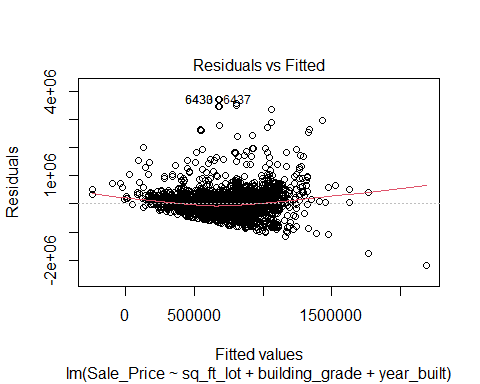
mean(vif(mult\_pred\_lm))

## [1] 1.153715

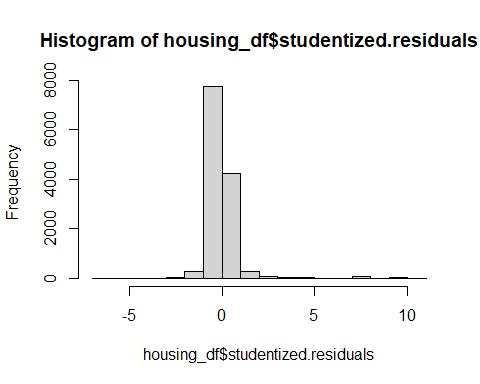
The assumption of no multicollinearity has been met because the largest VIF is less than 10, the average VIF is not substantially greater than 1, and the tolerance is above .2.

## Question 14

plot(mult\_pred\_lm)



hist(housing\_df$studentized.residuals)



The Residuals vs Fitted chart shows that the dots are all clustered and then fan out which means that there is heteroscedasticity in the data.The Q-Q plot shows deviations in the line at the extremes which means the data is not normalized and has a skew.

## Question 15

The issues with leverage, covariance ratios, and the assumption of independence not being met means that the regression model is biased and not good for generalizing the population.