6.1 Assignment: Housing Data

Scott Breitbach

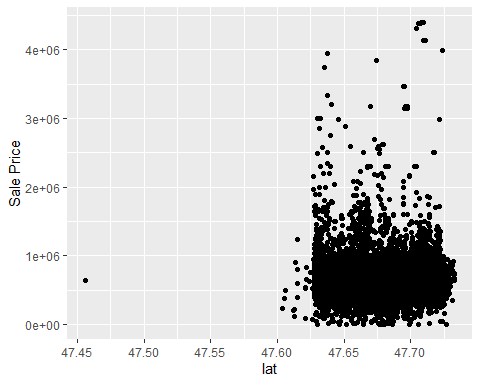
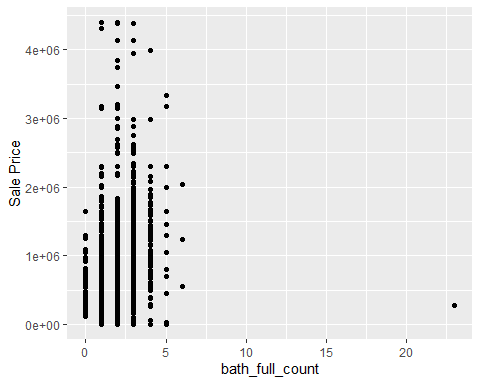
7/8/2020

## Multiple Regression

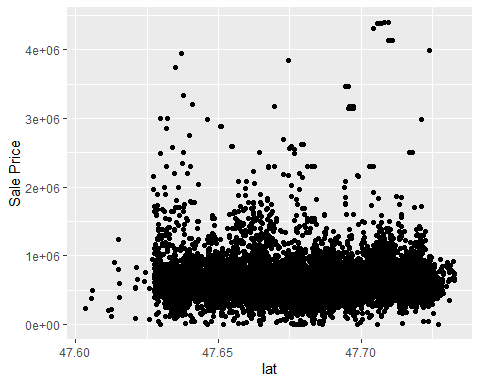
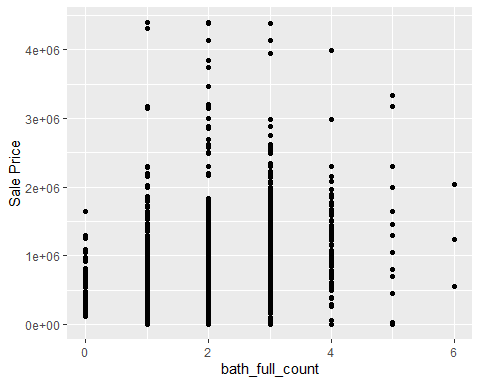
### Data for this assignment is focused on real estate transactions recorded from 1964 to 2016.

### You are interested in the following variables: Sale Price and several other possible predictors.

week\_6\_housing <- read\_excel("completed/Week6/week-6-housing.xlsx")  
# head(week\_6\_housing)



week\_6\_housing$bath\_full\_count[week\_6\_housing$bath\_full\_count==23] <- 4  
week\_6\_housing <- week\_6\_housing[!week\_6\_housing$lat < 47.5, ]



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

Having created histograms, boxplots, and scatterplots of each variable against Sale Price, I noticed a couple of points that seemed way outside of the rest.

One of the properties in the data set was recorded as having 23 full bathrooms. This seemed unusual even for a mansion and I figured it was probably a typo, so I looked up the address on Zillow.com and found that the house is listed as having 4 bathrooms. As a result, I changed the value from 23 to 4.

The other value I removed because the latitude was significantly outside of the others. With a quick search on Google Maps I found that the address matched the lat/lon correctly, but home was in Renton instead of Redmond and was therefore probably a typo. I removed this home from the data set.

### 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.

priceSqFt <- lm(`Sale Price` ~ sq\_ft\_lot, data = week\_6\_housing)  
housing\_lm <- lm(`Sale Price` ~ sq\_ft\_lot + square\_feet\_total\_living + building\_grade + year\_built, data = week\_6\_housing)

I created scatter plots of each variable with a linear model line added and visually observed the slope of each. I also performed simple regressions on each variable against the Sale Price and looked at the R squared values. Several of the variables (square feet of living space, bedrooms, and bathrooms) seemed like they would largely explain the same variance (because a larger house will tend to have more bedrooms, etc) so I chose square\_feet\_total\_living because it had the largest R2 (of those variables and largest overall at 0.2).

Additionally, I chose building\_grade, which had the second largest R2 value (0.15) and year\_built, which had an R2 of 0.06, but seemed like it would be unrelated to the other variables.

### 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?

summary(priceSqFt)

##   
## Call:  
## lm(formula = `Sale Price` ~ sq\_ft\_lot, data = week\_6\_housing)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2016086 -194853 -63381 91563 3735107   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 6.418e+05 3.800e+03 168.89 <2e-16 \*\*\*  
## sq\_ft\_lot 8.510e-01 6.218e-02 13.69 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 401500 on 12862 degrees of freedom  
## Multiple R-squared: 0.01436, Adjusted R-squared: 0.01428   
## F-statistic: 187.3 on 1 and 12862 DF, p-value: < 2.2e-16

summary(housing\_lm)

##   
## Call:  
## lm(formula = `Sale Price` ~ sq\_ft\_lot + square\_feet\_total\_living +   
## building\_grade + year\_built, data = week\_6\_housing)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2130050 -118827 -42226 42623 3736746   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -5.149e+06 3.930e+05 -13.102 < 2e-16 \*\*\*  
## sq\_ft\_lot 2.962e-01 5.825e-02 5.084 3.74e-07 \*\*\*  
## square\_feet\_total\_living 1.411e+02 4.884e+00 28.889 < 2e-16 \*\*\*  
## building\_grade 3.264e+04 4.412e+03 7.399 1.46e-13 \*\*\*  
## year\_built 2.597e+03 2.009e+02 12.927 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 356400 on 12859 degrees of freedom  
## Multiple R-squared: 0.2233, Adjusted R-squared: 0.2231   
## F-statistic: 924.4 on 4 and 12859 DF, p-value: < 2.2e-16

R2 and adjusted R2 are 0.01436 and 0.01428 respectively for the simple regression and 0.2233 and 0.2231 respectively for the multiple regression. The R2 of 0.01436 for sq\_ft\_lot tells us that this variable accounts for 1.4% of the variation in Sale Price. When including the other three predictors (square\_feet\_total\_living, building\_grade, and year\_built), the R2 increases to 0.2233, explaining 22.3% of the variation in Sale Price, which means that the three added predictors account for an additional 20.9% of the variance.

The adjusted R2 roughly how well our model generalizes. If the difference between the adjusted R2 value and the R2 value is small, it means the model generalizes well. In the instance of our multiple regression model, the difference (0.2233 - 0.2231) is 0.0002, indicating if the model were derived from the general population instead of the sample, that it would account for about 0.02% less variance.

### 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?

lm.beta(housing\_lm)

## sq\_ft\_lot square\_feet\_total\_living building\_grade   
## 0.04169814 0.34529255 0.08819358   
## year\_built   
## 0.11059174

sd(week\_6\_housing$`Sale Price`)

## [1] 404396.8

sd(week\_6\_housing$sq\_ft\_lot)

## [1] 56935.31

sd(week\_6\_housing$square\_feet\_total\_living)

## [1] 989.7259

sd(week\_6\_housing$building\_grade)

## [1] 1.092556

sd(week\_6\_housing$year\_built)

## [1] 17.22061

The standardized betas tell us that for each unit change in the standard deviation of the predictor variable, the outcome will change by the number of standard deviations reported. Larger values, like for square\_feet\_total\_living (0.35) and year\_built (0.11) indicate more importance in the model than building\_grade (0.09) and sq\_ft\_lot (0.04).

This tells us, for example, that for every standard deviation unit increase in living space, ~990 sqft, you could expect a 0.345 standard deviation unit increase in Sale Price (0.345 \* $404,396 = $139,517). So for each additional 990 square feet you could expect another $139,517 increase to the Sale Price. Similarly, for each 56,935 sqft of lot space (or ~1 American football field), you could expect an extra (0.0417 \* $139,517) $5,818 on the Sale Price.

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

format(round(confint(housing\_lm), digits = 4), scientific = FALSE)

## 2.5 % 97.5 %   
## (Intercept) "-5919352.2526" "-4378741.6305"  
## sq\_ft\_lot " 0.1820" " 0.4104"  
## square\_feet\_total\_living " 131.5119" " 150.6575"  
## building\_grade " 23995.2732" " 41292.3722"  
## year\_built " 2203.2751" " 2990.8424"

Understanding that a smaller confidence interval means that the value of b in the sample is representative of the population, indicating a better model. Given that, square\_feet\_total\_living and year\_built seem to be relatively good predictors, while the confidence intervals for sq\_ft\_lot and building grade are about double from the 2.5% interval to the 97.5% interval and might not be as good of predictors.

That said, none of the confidence intervals for any of the predictors cross zero, so that is good news for the model. Crossing zero would tell us that for that predictor part of the model has a negative correlation and part positive.

### 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.

anova(priceSqFt, housing\_lm)

## Analysis of Variance Table  
##   
## Model 1: `Sale Price` ~ sq\_ft\_lot  
## Model 2: `Sale Price` ~ sq\_ft\_lot + square\_feet\_total\_living + building\_grade +   
## year\_built  
## Res.Df RSS Df Sum of Sq F Pr(>F)   
## 1 12862 2.0734e+15   
## 2 12859 1.6338e+15 3 4.3959e+14 1153.3 < 2.2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

The anova() function lets us compare the two models and gives us the Pr(>F) value, which tells us the significance in the change in R2 value from the old model to the new model. In this instance the Pr(>F) of < 2.2e-16 indicates that the change in R2 from the first model (.014) to the second model (0.223) is very significant.

### 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.

diagnostics\_df <- week\_6\_housing[, c('Sale Date', 'Sale Price', "addr\_full",   
 "building\_grade", "square\_feet\_total\_living",   
 "year\_built", "sq\_ft\_lot")]  
diagnostics\_df$residuals<-resid(housing\_lm)  
diagnostics\_df$standardized.residuals<-rstandard(housing\_lm)  
diagnostics\_df$studentized.residuals<-rstudent(housing\_lm)  
diagnostics\_df$cooks.distance<-cooks.distance(housing\_lm)  
diagnostics\_df$dfbeta<-dfbeta(housing\_lm)  
diagnostics\_df$dffit<-dffits(housing\_lm)  
diagnostics\_df$leverage<-hatvalues(housing\_lm)  
diagnostics\_df$covariance.ratios<-covratio(housing\_lm)

### 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.

diagnostics\_df$large.residual<-diagnostics\_df$standardized.residuals > 2 |   
 diagnostics\_df$standardized.residuals < -2

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

"Sum of large residuals"

## [1] "Sum of large residuals"

sum(diagnostics\_df$large.residual)

## [1] 334

"Percent of data set with large residuals"

## [1] "Percent of data set with large residuals"

sum(diagnostics\_df$large.residual)/nrow(diagnostics\_df)\*100

## [1] 2.596393

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

std\_resid\_sort <- diagnostics\_df[order(-diagnostics\_df$standardized.residuals),]  
std\_resid\_sort[std\_resid\_sort$large.residual,   
 c("sq\_ft\_lot", "square\_feet\_total\_living", "building\_grade",   
 "year\_built", "standardized.residuals")]

## # A tibble: 334 x 5  
## sq\_ft\_lot square\_feet\_total\_li~ building\_grade year\_built standardized.resid~  
## <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 425145 1670 8 1964 10.5  
## 2 4749 2450 8 2010 10.4  
## 3 4244 2440 8 2011 10.4  
## 4 4368 2550 8 2010 10.3  
## 5 1327090 2410 6 1935 10.3  
## 6 5816 2750 8 2012 10.3  
## 7 13289 2810 8 2012 10.2  
## 8 4451 2960 8 2012 10.2  
## 9 8908 3010 8 2012 10.1  
## 10 5310 3110 8 2012 10.1  
## # ... with 324 more rows

std\_resid\_sort\_rev <- diagnostics\_df[order(diagnostics\_df$standardized.residuals),]  
head(std\_resid\_sort\_rev[std\_resid\_sort\_rev$large.residual,   
 c("sq\_ft\_lot", "square\_feet\_total\_living", "building\_grade",   
 "year\_built", "standardized.residuals")])

## # A tibble: 6 x 5  
## sq\_ft\_lot square\_feet\_total\_liv~ building\_grade year\_built standardized.resid~  
## <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 1631322 8750 12 1996 -6.18  
## 2 1127205 5830 12 1969 -4.31  
## 3 209088 7110 11 2006 -4.15  
## 4 45738 9360 10 2009 -3.88  
## 5 118483 8490 11 2008 -3.68  
## 6 288367 6880 10 2008 -3.51

334 cases (2.6% of the data set) has large standardized residuals (>|2|). We would expect around 5% of the data set to have residuals over 2. I would say that this isn’t necessarily cause for concern, however some of the values are quite large, as high as 10.5.

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

#### Leverage:

leverage\_df <- diagnostics\_df  
avgLeverage <- mean(leverage\_df$leverage)  
avgLeverage

## [1] 0.0003886816

avgLeverage \* 2

## [1] 0.0007773632

leverage\_df$leverageX2<-leverage\_df$leverage > (avgLeverage \* 2)  
sum(leverage\_df$leverageX2)

## [1] 783

sum(leverage\_df$large.residual & leverage\_df$leverageX2)

## [1] 110

avgLeverage \* 3

## [1] 0.001166045

leverage\_df$leverageX3<-leverage\_df$leverage > (avgLeverage \* 3)  
sum(leverage\_df$leverageX3)

## [1] 447

sum(leverage\_df$large.residual & leverage\_df$leverageX3)

## [1] 83

leverageSort <- leverage\_df[order(-leverage\_df$leverage),]  
leverageSort[leverageSort$large.residual,   
 c("sq\_ft\_lot", "square\_feet\_total\_living", "building\_grade",   
 "year\_built", "leverage", "standardized.residuals")]

## # A tibble: 334 x 6  
## sq\_ft\_lot square\_feet\_tot~ building\_grade year\_built leverage  
## <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 1631322 8750 12 1996 0.0638   
## 2 1327090 2410 6 1935 0.0440   
## 3 1166246 1230 9 1999 0.0374   
## 4 1166246 3690 11 1999 0.0348   
## 5 1127205 5830 12 1969 0.0301   
## 6 1008414 5270 9 2015 0.0254   
## 7 963702 5150 11 1997 0.0223   
## 8 307752 13540 12 1999 0.0142   
## 9 657816 5790 11 1999 0.0100   
## 10 574992 2700 7 2003 0.00870  
## # ... with 324 more rows, and 1 more variable: standardized.residuals <dbl>

Of the 334 cases with large residuals, 110 have leverage values over 2X the average and 83 with over 3X the average. These cases with high hat values have significantly more weight in influencing the outcome variables over the predicted values from the model than the majority of the cases and should probably be reviewed more closely.

If we wanted to see how much influence each of these cases was having on the parameters of the model, we could rerun the analysis with each case excluded and look at the change in the b coefficients.

#### Cook’s Distance:

cooksSort <- diagnostics\_df[order(-diagnostics\_df$cooks.distance),]  
cooksSort[cooksSort$large.residual,  
 c("sq\_ft\_lot", "square\_feet\_total\_living", "building\_grade",   
 "year\_built", "cooks.distance", "standardized.residuals")]

## # A tibble: 334 x 6  
## sq\_ft\_lot square\_feet\_tot~ building\_grade year\_built cooks.distance  
## <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 1327090 2410 6 1935 0.979   
## 2 1631322 8750 12 1996 0.520   
## 3 657816 5790 11 1999 0.144   
## 4 1127205 5830 12 1969 0.115   
## 5 1166246 1230 9 1999 0.109   
## 6 425145 1670 8 1964 0.102   
## 7 1166246 3690 11 1999 0.0473  
## 8 1008414 5270 9 2015 0.0455  
## 9 139392 11810 13 2000 0.0444  
## 10 307752 13540 12 1999 0.0298  
## # ... with 324 more rows, and 1 more variable: standardized.residuals <dbl>

Cook’s distance will give us information as to h ow much influence a case has on the entire model and all the cases that the model predicts. The text suggests that values over 1 may be cause for concern and, while there is a 0.52 value in there the only value that comes close to 1 is 0.98. This isn’t over 1, but it’s close enough that it should probably be investigated, and perhaps also the 0.52 value because their standardized residuals are quite large as well at 10.3 and -6.2 respectively.

#### Covariance Ratios:

cvrHi <- 1 + (3 \* (4 + 1) / nrow(diagnostics\_df))  
cvrHi

## [1] 1.001166

cvrLo <- 1 - (3 \* (4 + 1) / nrow(diagnostics\_df))  
cvrLo

## [1] 0.998834

covrSort <- diagnostics\_df[order(-diagnostics\_df$covariance.ratios),]  
  
covrSort$covariance.boundaries<-covrSort$covariance.ratios > cvrHi |   
 covrSort$covariance.ratios < cvrLo  
  
sum(covrSort$large.residual & covrSort$covariance.boundaries)

## [1] 264

covrSort[covrSort$large.residual & covrSort$covariance.boundaries,  
 c("sq\_ft\_lot", "year\_built", "covariance.ratios", "standardized.residuals",   
 "leverage", "cooks.distance")]

## # A tibble: 264 x 6  
## sq\_ft\_lot year\_built covariance.rati~ standardized.re~ leverage  
## <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 1631322 1996 1.05 -6.18 0.0638   
## 2 1166246 1999 1.03 2.56 0.0348   
## 3 1166246 1999 1.03 3.75 0.0374   
## 4 1127205 1969 1.02 -4.31 0.0301   
## 5 1008414 2015 1.02 -2.96 0.0254   
## 6 963702 1997 1.02 2.08 0.0223   
## 7 307752 1999 1.01 -3.21 0.0142   
## 8 574992 2003 1.01 -2.09 0.00870  
## 9 544199 2007 1.00 -3.16 0.00760  
## 10 207781 2003 1.00 2.76 0.00668  
## # ... with 254 more rows, and 1 more variable: cooks.distance <dbl>

Looking at the upper (1.001166) and lower (0.998834) covariance ratio boundaries, it looks like of our 334 cases with large residuals, 264 have values above or below those limits. The case with the highest covariance ratio also has the second-highest Cook’s distance, so it may be something to look at, but overall the other Cook’s distances (apart from the 0.98) are fairly low and probably not much cause for concern since they’re not having much influence on the model.

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

durbinWatsonTest(housing\_lm)

## lag Autocorrelation D-W Statistic p-value  
## 1 0.7292876 0.5414148 0  
## Alternative hypothesis: rho != 0

The Durbin-Watson test tells us whether the residuals between adjacent residuals are correlated, either positively or negatively. A value of 2 indicates no correlation, which is good, and values above 3 (with a negative autocorrelation) or below 1 (positive autocorrelation) are concerning. So, this value of 0.54 raises some alarms, especially since the p-value is significant at 0.

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

vif(housing\_lm)

## sq\_ft\_lot square\_feet\_total\_living building\_grade   
## 1.113588 2.365294 2.352617   
## year\_built   
## 1.211685

mean(vif(housing\_lm))

## [1] 1.760796

1/vif(housing\_lm)

## sq\_ft\_lot square\_feet\_total\_living building\_grade   
## 0.8979978 0.4227804 0.4250585   
## year\_built   
## 0.8252967

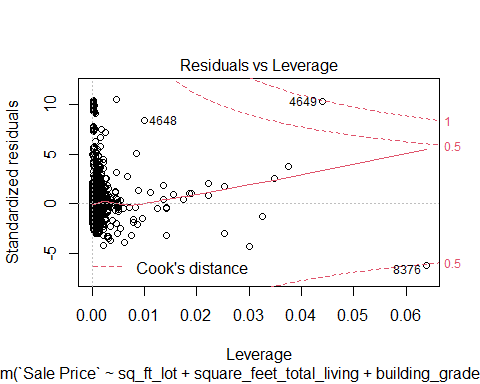
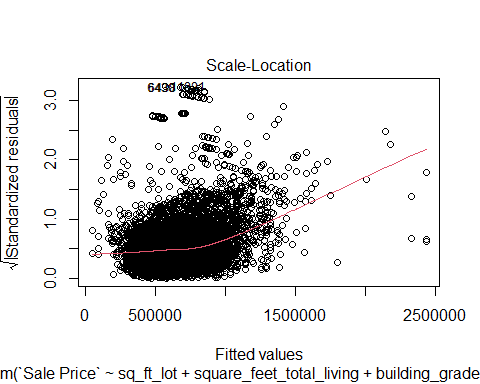
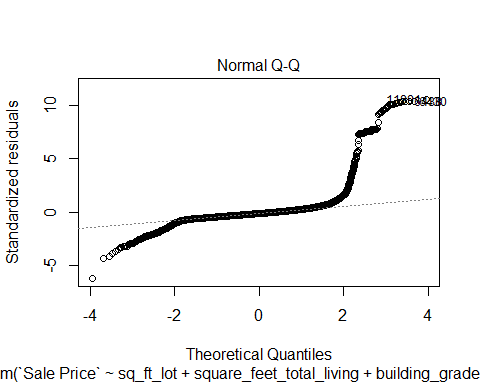
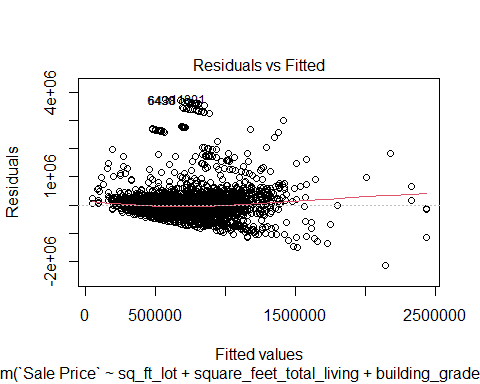
The largest Variance inflation factor (VIF) in my model is square\_feet\_total\_living at 2.37 (with building\_grade coming in at a close second with 2.35). Since these values are all well below 10, there is little cause for concern.

The average VIF for my model comes out to 1.76. While this values is above 1, I don’t know that it would be considered significantly above 1 so it is probably okay.

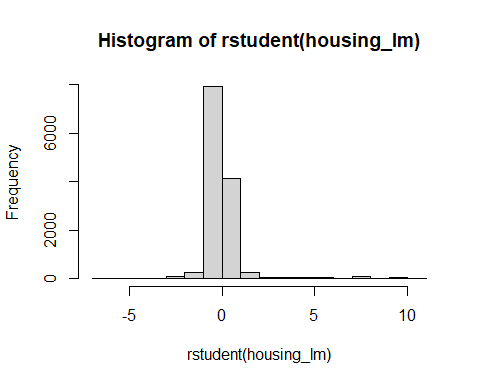
Finally for tolerance, values below 0.2 start to raise flags and less than 0.1 is problematic. Fortunately, the lowest value in my model is 0.42.

### 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.

plot(housing\_lm)



hist(rstudent(housing\_lm))



Looking at the Residuals vs Fitted values plot, the data appears pretty normal/linear overall, although there are a few clusters of dots with relatively high residuals, leading to perhaps some heteroscedasticity in the data indicating variance across the residuals. The Scale-Location plot shows largely the same information except the Fitted values are plotted against standardized residuals, which also makes all the values positive.

The second plot, Normal Q-Q, helps us to visualize the deviations from normality. We can see that the majority of the points around the middle fit the line fairly well, but that they deviate at the negative end and deviate sharply at the positive end indicating that the data set is not entirely normally distributed. The histogram confirms this, showing a distribution with positive kurtosis and long tails.

The final plot, Residuals vs Leverage shows that the vast majority of cases cluster toward zero leverage, which is good, though there are a handful which are quite high. In addition, most of them fall within a reasonable Cook’s distance with only a couple of exceptions.

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

Of the predictive variables I chose to add to my model, the square feet of living space seemed like the best predictor. Year built and building grade also seemed to be somewhat important, though less so. The square feet of the lot was the worst predictor and I would probably remove it from my model.

After removing a couple of outliers, the data set was still quite large, at 12864 cases. With such a large sample size, the model is going to generalize better with the overall population. Given the large data set, even with 334 cases that have high standardized residuals the overall percentage of problematic data points was relatively quite small. That said, there were a handful of data points that did not fit well in the model, though for the most part they held little influence on the model. It would be good, however, to look at the worst offenders more closely to see if there might be a justifiable reason to exclude them from the data set / model.