Exercise\_12\_Housing\_Data\_NoskyChristopher

Nosky, Christopher

1/31/2021

knitr::opts\_chunk$set(echo = TRUE)  
  
setwd('C:/Users/cwnos/Documents/DSC520/Git repo/DSC520/dsc520')  
  
library(readxl)  
library(Rcmdr)

## Warning: package 'Rcmdr' was built under R version 4.0.3

## Loading required package: splines

## Loading required package: RcmdrMisc

## Warning: package 'RcmdrMisc' was built under R version 4.0.3

## Loading required package: car

## Warning: package 'car' was built under R version 4.0.3

## Loading required package: carData

## Warning: package 'carData' was built under R version 4.0.3

## Loading required package: sandwich

## Warning: package 'sandwich' was built under R version 4.0.3

## Loading required package: effects

## Warning: package 'effects' was built under R version 4.0.3

## Registered S3 methods overwritten by 'lme4':  
## method from  
## cooks.distance.influence.merMod car   
## influence.merMod car   
## dfbeta.influence.merMod car   
## dfbetas.influence.merMod car

## lattice theme set by effectsTheme()  
## See ?effectsTheme for details.

## The Commander GUI is launched only in interactive sessions

##   
## Attaching package: 'Rcmdr'

## The following object is masked from 'package:base':  
##   
## errorCondition

library(ggplot2)

## Warning: package 'ggplot2' was built under R version 4.0.3

library(pastecs)

## Warning: package 'pastecs' was built under R version 4.0.3

library(MASS, pos = 18)  
library(car)  
week\_7\_housing <- read\_excel("data/week-6-housing.xlsx")  
  
week\_7\_housing\_orginial <- read\_excel("data/week-6-housing.xlsx")  
  
week\_7\_housing <- within(week\_7\_housing, {  
 addr\_full <- NULL  
 building\_grade <- NULL  
 ctyname <- NULL  
 lat <- NULL  
 lon <- NULL  
 sale\_instrument <- NULL   
 sale\_reason <- NULL  
 sale\_warning <- NULL  
 sitetype <- NULL  
 year\_renovated <- NULL   
})  
  
week\_7\_housing\_Updated <- na.omit(week\_7\_housing)  
sale\_price <- week\_7\_housing\_Updated$`Sale Price`  
bathrooms <- subset(week\_7\_housing\_Updated,   
 select = c('bath\_full\_count', 'bath\_half\_count',  
 'bath\_3qtr\_count'), drop = FALSE)  
  
total\_bathrooms <- (bathrooms$bath\_full\_count \* 1) +   
 (bathrooms$bath\_half\_count \* 0.5) +   
 (bathrooms$bath\_3qtr\_count \* 0.75)  
  
week\_7\_housing\_Updated$total\_bathrooms <- total\_bathrooms  
  
week\_7\_housing\_Updated <- subset(week\_7\_housing\_Updated,   
 select = c('Sale Price',   
 'square\_feet\_total\_living',  
 'bedrooms',   
 'sq\_ft\_lot',   
 'total\_bathrooms'))   
  
str(week\_7\_housing\_Updated)

## tibble [12,865 x 5] (S3: tbl\_df/tbl/data.frame)  
## $ Sale Price : num [1:12865] 698000 649990 572500 420000 369900 ...  
## $ square\_feet\_total\_living: num [1:12865] 2810 2880 2770 1620 1440 4160 3960 3720 4160 2760 ...  
## $ bedrooms : num [1:12865] 4 4 4 3 3 4 5 4 4 4 ...  
## $ sq\_ft\_lot : num [1:12865] 6635 5570 8444 9600 7526 ...  
## $ total\_bathrooms : num [1:12865] 2.5 2.75 2.25 1.75 1.75 3.25 3.75 2.5 3.25 1.75 ...

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:

Explain why you chose to remove data from your data set.

* Starting with the removed data points outlined in Tech Fort this week, I considered each point and came to the same conclusions indicated and removed them from the start.
* As indicated I also reworked the three(3) separate bathroom variables into a single value, without knowing for certain I was using the same methodology this proved to be an amusing endeavor first arriving at a sum of all the bathroom values for the entire data set. After adding the total\_bathrooms subset back into week\_7\_housing\_Updated I created a new subset week\_7\_housing\_Updated with 5 variables.
* The final variable I removed was **building\_grade** because after considering what those values might indicate it opened up a ‘rabbit-hole’ of questions I wasn’t certain I could get the answers for. For example, the biggest question I had after considering that it was a ranking metric for the quality of the build I now question how the ranking might be normalized considering the range of the data set. Given that building code is updated regularly and construction methods and materials are often improved year after year, are we ranking the building\_grade based on the year it was built or did we base it off of current standards? If we were to know which method was used to get the ranking we could possibly through a bit of work write a function that normalizes the metric around a single standard.

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.

RegModel.1 <- lm(sale\_price~square\_feet\_total\_living, data = week\_7\_housing\_Updated)

RegModel.2 <- lm(sale\_price~square\_feet\_total\_living+total\_bathrooms +bedrooms+sq\_ft\_lot,   
 data = week\_7\_housing\_Updated)

* Here I have created two models the first contains Sale price with sq\_ft\_lot as it’s predictor variable, and the second uses Sale price with predictors sq\_ft\_lot, bedrooms, total\_bathrooms, square\_feet\_total\_living.
* My inclusion of the predictors bedrooms, total\_bathrooms and square\_feet\_total\_living seemed a logical choice as I didn’t know the effect each may have on the sale price. My intention was to start with all three and remove them one at a time to see if we could get a better fit.

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(RegModel.1)

##   
## Call:  
## lm(formula = sale\_price ~ square\_feet\_total\_living, data = week\_7\_housing\_Updated)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1800136 -120257 -41547 44028 3811745   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.891e+05 8.745e+03 21.62 <2e-16 \*\*\*  
## square\_feet\_total\_living 1.857e+02 3.208e+00 57.88 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 360200 on 12863 degrees of freedom  
## Multiple R-squared: 0.2066, Adjusted R-squared: 0.2066   
## F-statistic: 3351 on 1 and 12863 DF, p-value: < 2.2e-16

summary(RegModel.2)

##   
## Call:  
## lm(formula = sale\_price ~ square\_feet\_total\_living + total\_bathrooms +   
## bedrooms + sq\_ft\_lot, data = week\_7\_housing\_Updated)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1975012 -117703 -40353 44570 3787149   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 2.158e+05 1.451e+04 14.869 < 2e-16 \*\*\*  
## square\_feet\_total\_living 1.828e+02 5.268e+00 34.698 < 2e-16 \*\*\*  
## total\_bathrooms 2.979e+04 6.969e+03 4.275 1.92e-05 \*\*\*  
## bedrooms -2.734e+04 4.506e+03 -6.068 1.33e-09 \*\*\*  
## sq\_ft\_lot 9.441e-02 5.800e-02 1.628 0.104   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 359500 on 12860 degrees of freedom  
## Multiple R-squared: 0.2098, Adjusted R-squared: 0.2096   
## F-statistic: 853.6 on 4 and 12860 DF, p-value: < 2.2e-16

* From our reading, we know we can look at the R-squared and Adjusted R-squared values like a percentage of variability that our predictor variables account for. Looking at the RegModel.1 values it would appear that sq\_ft\_lot accounts for about 1.4% of the variability in sale price, sadly this doesn’t help us out all that much.
* Our second model (I was hoping would do better) shows R-squared and Adjusted R-squared values are 0.2096 & 0.2095 respectively and can account for about 21% of the variability in our sale price. What strikes me as odd as I look at the rest of this summary is the bedrooms variable appears to impact the sale price negatively, that is if I’m reading this correctly. As I understand it, for each additional bedroom the sale price drops by about $28,000.
* At this point, I would say it may be worth revisiting the included data points I am using to see if we can find additional factors that may be able to explain this variability.

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?

compareCoefs(RegModel.1, RegModel.2)

## Calls:  
## 1: lm(formula = sale\_price ~ square\_feet\_total\_living, data =   
## week\_7\_housing\_Updated)  
## 2: lm(formula = sale\_price ~ square\_feet\_total\_living + total\_bathrooms +   
## bedrooms + sq\_ft\_lot, data = week\_7\_housing\_Updated)  
##   
## Model 1 Model 2  
## (Intercept) 189107 215765  
## SE 8745 14511  
##   
## square\_feet\_total\_living 185.72 182.80  
## SE 3.21 5.27  
##   
## total\_bathrooms 29793  
## SE 6969  
##   
## bedrooms -27340  
## SE 4506  
##   
## sq\_ft\_lot 0.0944  
## SE 0.0580  
##

* Above you can see the standardized beta coefficients for each model’s variables. As I found by [searching](https://www.dataanalytics.org.uk/beta-coefficients-from-linear-models/#beta):  
  ***Beta coefficients are regression coefficients (analogous to the slope in a simple regression/correlation) that are standardized against one another. This standardization means that they are “on the same scale”, or have the same units, which allows you to compare the magnitude of their effects directly.***  
  These results indicate that of the variables I’ve chosen that square\_feet\_total\_living has the greatest impact on sale\_price.

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

confint.lm(RegModel.1, level = 0.95)

## 2.5 % 97.5 %  
## (Intercept) 171965.2516 206247.8664  
## square\_feet\_total\_living 179.4286 192.0067

confint.lm(RegModel.2, level = 0.95)

## 2.5 % 97.5 %  
## (Intercept) 1.873207e+05 2.442088e+05  
## square\_feet\_total\_living 1.724756e+02 1.931291e+02  
## total\_bathrooms 1.613289e+04 4.345214e+04  
## bedrooms -3.617257e+04 -1.850824e+04  
## sq\_ft\_lot -1.927802e-02 2.081036e-01

* To be honest, I’m not sure how to interpret these values. Looking at our first model, if I understand this weeks reading I would say that with sq\_ft\_lot as a predictor we are 95% confident that the interval (.729, .972) captured the true mean of our model. This explanation becomes a little less clear as I begin looking at the next two models, I suspect this is due to using multiple predictors in these models.

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(RegModel.1, RegModel.2)

## Analysis of Variance Table  
##   
## Model 1: sale\_price ~ square\_feet\_total\_living  
## Model 2: sale\_price ~ square\_feet\_total\_living + total\_bathrooms + bedrooms +   
## sq\_ft\_lot  
## Res.Df RSS Df Sum of Sq F Pr(>F)   
## 1 12863 1.6689e+15   
## 2 12860 1.6622e+15 3 6.6248e+12 17.084 4.538e-11 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

* Based on our reading, I interpret these tables to indicate that both RegModel.2 significantly improved the fit over RegModel.1. As I discussed in my Discussion post last week what we’re looking at here are the F-ratio and the p value to arrive at this conclusion.

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

RegModelOrg <-   
 lm(sale\_price~square\_feet\_total\_living+total\_bathrooms+bedrooms+building\_grade+lat+lon+present\_use+sale\_instrument+sale\_reason+sq\_ft\_lot+year\_built+year\_renovated+zip5,  
 data=week\_7\_housing\_orginial)  
summary(RegModelOrg)

##   
## Call:  
## lm(formula = sale\_price ~ square\_feet\_total\_living + total\_bathrooms +   
## bedrooms + building\_grade + lat + lon + present\_use + sale\_instrument +   
## sale\_reason + sq\_ft\_lot + year\_built + year\_renovated + zip5,   
## data = week\_7\_housing\_orginial)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2289747 -120423 -44170 41409 3695867   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -4.034e+08 1.994e+08 -2.023 0.0431 \*   
## square\_feet\_total\_living 1.462e+02 6.481e+00 22.562 < 2e-16 \*\*\*  
## total\_bathrooms -5.018e+03 7.211e+03 -0.696 0.4865   
## bedrooms -1.109e+04 4.900e+03 -2.264 0.0236 \*   
## building\_grade 2.851e+04 4.487e+03 6.354 2.17e-10 \*\*\*  
## lat -3.033e+04 1.392e+05 -0.218 0.8276   
## lon -3.316e+05 7.561e+04 -4.385 1.17e-05 \*\*\*  
## present\_use -7.608e+02 1.048e+02 -7.258 4.15e-13 \*\*\*  
## sale\_instrument 1.506e+02 1.038e+03 0.145 0.8846   
## sale\_reason -1.163e+04 1.281e+03 -9.080 < 2e-16 \*\*\*  
## sq\_ft\_lot 3.966e-01 6.120e-02 6.481 9.47e-11 \*\*\*  
## year\_built 3.265e+03 2.625e+02 12.436 < 2e-16 \*\*\*  
## year\_renovated 8.348e+01 1.429e+01 5.844 5.23e-09 \*\*\*  
## zip5 3.651e+03 1.995e+03 1.830 0.0672 .   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 353700 on 12851 degrees of freedom  
## Multiple R-squared: 0.2356, Adjusted R-squared: 0.2348   
## F-statistic: 304.6 on 13 and 12851 DF, p-value: < 2.2e-16

outlierTest(RegModelOrg)

## rstudent unadjusted p-value Bonferroni p  
## 11992 10.52023 8.8719e-26 1.1414e-21  
## 6430 10.46022 1.6653e-25 2.1424e-21  
## 6438 10.42325 2.4505e-25 3.1526e-21  
## 6437 10.41869 2.5698e-25 3.3060e-21  
## 6431 10.31333 7.6660e-25 9.8623e-21  
## 6436 10.28181 1.0609e-24 1.3648e-20  
## 6441 10.24501 1.5485e-24 1.9922e-20  
## 6432 10.20397 2.3570e-24 3.0323e-20  
## 6433 10.17790 3.0754e-24 3.9565e-20  
## 6442 10.16690 3.4402e-24 4.4258e-20

outlierTest(RegModel.1)

## rstudent unadjusted p-value Bonferroni p  
## 11992 10.62908 2.8052e-26 3.6089e-22  
## 4649 10.49256 1.1864e-25 1.5263e-21  
## 6438 10.42218 2.4775e-25 3.1873e-21  
## 6430 10.41696 2.6161e-25 3.3656e-21  
## 6437 10.36474 4.5024e-25 5.7923e-21  
## 6431 10.26036 1.3225e-24 1.7014e-20  
## 6436 10.22905 1.8232e-24 2.3455e-20  
## 6441 10.15082 4.0508e-24 5.2113e-20  
## 6432 10.12475 5.2784e-24 6.7907e-20  
## 6442 10.07261 8.9440e-24 1.1506e-19

outlierTest(RegModel.2)

## rstudent unadjusted p-value Bonferroni p  
## 11992 10.60351 3.6804e-26 4.7348e-22  
## 6430 10.47895 1.3685e-25 1.7606e-21  
## 6437 10.42752 2.3434e-25 3.0148e-21  
## 4649 10.42102 2.5076e-25 3.2260e-21  
## 6438 10.40716 2.8973e-25 3.7274e-21  
## 6431 10.32415 6.8549e-25 8.8188e-21  
## 6436 10.29126 9.6244e-25 1.2382e-20  
## 6433 10.17133 3.2878e-24 4.2297e-20  
## 6434 10.17131 3.2886e-24 4.2308e-20  
## 6432 10.16850 3.3841e-24 4.3537e-20

Out\_L\_week\_7\_housing\_orginial <- week\_7\_housing\_orginial

str(Out\_L\_week\_7\_housing\_orginial)

## tibble [12,865 x 24] (S3: tbl\_df/tbl/data.frame)  
## $ Sale Date : POSIXct[1:12865], format: "2006-01-03" "2006-01-03" ...  
## $ Sale Price : num [1:12865] 698000 649990 572500 420000 369900 ...  
## $ sale\_reason : num [1:12865] 1 1 1 1 1 1 1 1 1 1 ...  
## $ sale\_instrument : num [1:12865] 3 3 3 3 3 15 3 3 3 3 ...  
## $ sale\_warning : chr [1:12865] NA NA NA NA ...  
## $ sitetype : chr [1:12865] "R1" "R1" "R1" "R1" ...  
## $ addr\_full : chr [1:12865] "17021 NE 113TH CT" "11927 178TH PL NE" "13315 174TH AVE NE" "3303 178TH AVE NE" ...  
## $ zip5 : num [1:12865] 98052 98052 98052 98052 98052 ...  
## $ ctyname : chr [1:12865] "REDMOND" "REDMOND" NA "REDMOND" ...  
## $ postalctyn : chr [1:12865] "REDMOND" "REDMOND" "REDMOND" "REDMOND" ...  
## $ lon : num [1:12865] -122 -122 -122 -122 -122 ...  
## $ lat : num [1:12865] 47.7 47.7 47.7 47.6 47.7 ...  
## $ building\_grade : num [1:12865] 9 9 8 8 7 7 10 10 9 8 ...  
## $ square\_feet\_total\_living: num [1:12865] 2810 2880 2770 1620 1440 4160 3960 3720 4160 2760 ...  
## $ bedrooms : num [1:12865] 4 4 4 3 3 4 5 4 4 4 ...  
## $ bath\_full\_count : num [1:12865] 2 2 1 1 1 2 3 2 2 1 ...  
## $ bath\_half\_count : num [1:12865] 1 0 1 0 0 1 0 1 1 0 ...  
## $ bath\_3qtr\_count : num [1:12865] 0 1 1 1 1 1 1 0 1 1 ...  
## $ year\_built : num [1:12865] 2003 2006 1987 1968 1980 ...  
## $ year\_renovated : num [1:12865] 0 0 0 0 0 0 0 0 0 0 ...  
## $ current\_zoning : chr [1:12865] "R4" "R4" "R6" "R4" ...  
## $ sq\_ft\_lot : num [1:12865] 6635 5570 8444 9600 7526 ...  
## $ prop\_type : chr [1:12865] "R" "R" "R" "R" ...  
## $ present\_use : num [1:12865] 2 2 2 2 2 2 2 2 2 2 ...

Out\_L\_week\_7\_housing\_updated <- week\_7\_housing\_Updated

str(Out\_L\_week\_7\_housing\_updated)

## tibble [12,865 x 5] (S3: tbl\_df/tbl/data.frame)  
## $ Sale Price : num [1:12865] 698000 649990 572500 420000 369900 ...  
## $ square\_feet\_total\_living: num [1:12865] 2810 2880 2770 1620 1440 4160 3960 3720 4160 2760 ...  
## $ bedrooms : num [1:12865] 4 4 4 3 3 4 5 4 4 4 ...  
## $ sq\_ft\_lot : num [1:12865] 6635 5570 8444 9600 7526 ...  
## $ total\_bathrooms : num [1:12865] 2.5 2.75 2.25 1.75 1.75 3.25 3.75 2.5 3.25 1.75 ...

RegModel.3 <- lm(sale\_price~square\_feet\_total\_living, data = Out\_L\_week\_7\_housing\_updated)  
summary(RegModel.3)

##   
## Call:  
## lm(formula = sale\_price ~ square\_feet\_total\_living, data = Out\_L\_week\_7\_housing\_updated)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1800136 -120257 -41547 44028 3811745   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.891e+05 8.745e+03 21.62 <2e-16 \*\*\*  
## square\_feet\_total\_living 1.857e+02 3.208e+00 57.88 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 360200 on 12863 degrees of freedom  
## Multiple R-squared: 0.2066, Adjusted R-squared: 0.2066   
## F-statistic: 3351 on 1 and 12863 DF, p-value: < 2.2e-16

RegModel.4 <- lm(sale\_price~square\_feet\_total\_living+total\_bathrooms+bedrooms+sq\_ft\_lot,   
 data = Out\_L\_week\_7\_housing\_updated)  
summary(RegModel.4)

##   
## Call:  
## lm(formula = sale\_price ~ square\_feet\_total\_living + total\_bathrooms +   
## bedrooms + sq\_ft\_lot, data = Out\_L\_week\_7\_housing\_updated)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1975012 -117703 -40353 44570 3787149   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 2.158e+05 1.451e+04 14.869 < 2e-16 \*\*\*  
## square\_feet\_total\_living 1.828e+02 5.268e+00 34.698 < 2e-16 \*\*\*  
## total\_bathrooms 2.979e+04 6.969e+03 4.275 1.92e-05 \*\*\*  
## bedrooms -2.734e+04 4.506e+03 -6.068 1.33e-09 \*\*\*  
## sq\_ft\_lot 9.441e-02 5.800e-02 1.628 0.104   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 359500 on 12860 degrees of freedom  
## Multiple R-squared: 0.2098, Adjusted R-squared: 0.2096   
## F-statistic: 853.6 on 4 and 12860 DF, p-value: < 2.2e-16

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

Out\_L\_week\_7\_housing\_updated$standardized.residuals <- rstandard(RegModel.4)  
Out\_L\_week\_7\_housing\_updated$studentized.residuals <- rstudent(RegModel.4)  
Out\_L\_week\_7\_housing\_updated$cooks.distance <- cooks.distance(RegModel.4)  
Out\_L\_week\_7\_housing\_updated$dfbeta <- dfbeta(RegModel.4)  
Out\_L\_week\_7\_housing\_updated$leverage <- hatvalues(RegModel.4)  
Out\_L\_week\_7\_housing\_updated$covariance.ratios <- covratio(RegModel.4)  
  
str(Out\_L\_week\_7\_housing\_updated)

## tibble [12,865 x 11] (S3: tbl\_df/tbl/data.frame)  
## $ Sale Price : num [1:12865] 698000 649990 572500 420000 369900 ...  
## $ square\_feet\_total\_living: num [1:12865] 2810 2880 2770 1620 1440 4160 3960 3720 4160 2760 ...  
## $ bedrooms : num [1:12865] 4 4 4 3 3 4 5 4 4 4 ...  
## $ sq\_ft\_lot : num [1:12865] 6635 5570 8444 9600 7526 ...  
## $ total\_bathrooms : num [1:12865] 2.5 2.75 2.25 1.75 1.75 3.25 3.75 2.5 3.25 1.75 ...  
## $ standardized.residuals : Named num [1:12865] 0.00783 -0.18175 -0.30069 -0.17504 -0.22233 ...  
## ..- attr(\*, "names")= chr [1:12865] "1" "2" "3" "4" ...  
## $ studentized.residuals : Named num [1:12865] 0.00783 -0.18174 -0.30068 -0.17504 -0.22232 ...  
## ..- attr(\*, "names")= chr [1:12865] "1" "2" "3" "4" ...  
## $ cooks.distance : Named num [1:12865] 1.52e-09 7.56e-07 3.12e-06 1.03e-06 1.82e-06 ...  
## ..- attr(\*, "names")= chr [1:12865] "1" "2" "3" "4" ...  
## $ dfbeta : num [1:12865, 1:5] -0.115 7.646 -4.183 -22.068 -26.146 ...  
## ..- attr(\*, "dimnames")=List of 2  
## .. ..$ : chr [1:12865] "1" "2" "3" "4" ...  
## .. ..$ : chr [1:5] "(Intercept)" "square\_feet\_total\_living" "total\_bathrooms" "bedrooms" ...  
## $ leverage : Named num [1:12865] 0.000124 0.000114 0.000172 0.000167 0.000184 ...  
## ..- attr(\*, "names")= chr [1:12865] "1" "2" "3" "4" ...  
## $ covariance.ratios : Named num [1:12865] 1 1 1 1 1 ...  
## ..- attr(\*, "names")= chr [1:12865] "1" "2" "3" "4" ...

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

Out\_L\_week\_7\_housing\_updated$large.residual <- Out\_L\_week\_7\_housing\_updated$standardized.residuals > 2 | Out\_L\_week\_7\_housing\_updated$studentized.residuals < -2  
  
str(Out\_L\_week\_7\_housing\_updated)

## tibble [12,865 x 12] (S3: tbl\_df/tbl/data.frame)  
## $ Sale Price : num [1:12865] 698000 649990 572500 420000 369900 ...  
## $ square\_feet\_total\_living: num [1:12865] 2810 2880 2770 1620 1440 4160 3960 3720 4160 2760 ...  
## $ bedrooms : num [1:12865] 4 4 4 3 3 4 5 4 4 4 ...  
## $ sq\_ft\_lot : num [1:12865] 6635 5570 8444 9600 7526 ...  
## $ total\_bathrooms : num [1:12865] 2.5 2.75 2.25 1.75 1.75 3.25 3.75 2.5 3.25 1.75 ...  
## $ standardized.residuals : Named num [1:12865] 0.00783 -0.18175 -0.30069 -0.17504 -0.22233 ...  
## ..- attr(\*, "names")= chr [1:12865] "1" "2" "3" "4" ...  
## $ studentized.residuals : Named num [1:12865] 0.00783 -0.18174 -0.30068 -0.17504 -0.22232 ...  
## ..- attr(\*, "names")= chr [1:12865] "1" "2" "3" "4" ...  
## $ cooks.distance : Named num [1:12865] 1.52e-09 7.56e-07 3.12e-06 1.03e-06 1.82e-06 ...  
## ..- attr(\*, "names")= chr [1:12865] "1" "2" "3" "4" ...  
## $ dfbeta : num [1:12865, 1:5] -0.115 7.646 -4.183 -22.068 -26.146 ...  
## ..- attr(\*, "dimnames")=List of 2  
## .. ..$ : chr [1:12865] "1" "2" "3" "4" ...  
## .. ..$ : chr [1:5] "(Intercept)" "square\_feet\_total\_living" "total\_bathrooms" "bedrooms" ...  
## $ leverage : Named num [1:12865] 0.000124 0.000114 0.000172 0.000167 0.000184 ...  
## ..- attr(\*, "names")= chr [1:12865] "1" "2" "3" "4" ...  
## $ covariance.ratios : Named num [1:12865] 1 1 1 1 1 ...  
## ..- attr(\*, "names")= chr [1:12865] "1" "2" "3" "4" ...  
## $ large.residual : Named logi [1:12865] FALSE FALSE FALSE FALSE FALSE TRUE ...  
## ..- attr(\*, "names")= chr [1:12865] "1" "2" "3" "4" ...

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

sum(Out\_L\_week\_7\_housing\_updated$large.residual)

## [1] 322

head(Out\_L\_week\_7\_housing\_updated)

## # A tibble: 6 x 12  
## `Sale Price` square\_feet\_tot~ bedrooms sq\_ft\_lot total\_bathrooms  
## <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 698000 2810 4 6635 2.5   
## 2 649990 2880 4 5570 2.75  
## 3 572500 2770 4 8444 2.25  
## 4 420000 1620 3 9600 1.75  
## 5 369900 1440 3 7526 1.75  
## 6 184667 4160 4 7280 3.25  
## # ... with 11 more variables: standardized.residuals <dbl>,  
## # studentized.residuals <dbl>, cooks.distance <dbl>,  
## # dfbeta[,"(Intercept)"] <dbl>, [,"square\_feet\_total\_living"] <dbl>,  
## # [,"total\_bathrooms"] <dbl>, [,"bedrooms"] <dbl>, [,"sq\_ft\_lot"] <dbl>,  
## # leverage <dbl>, covariance.ratios <dbl>, large.residual <lgl>

Out\_L\_week\_7\_housing\_updated[Out\_L\_week\_7\_housing\_updated$large.residual , c("Sale Price", "square\_feet\_total\_living","total\_bathrooms" , "bedrooms", "sq\_ft\_lot")]

## # A tibble: 322 x 5  
## `Sale Price` square\_feet\_total\_living total\_bathrooms bedrooms sq\_ft\_lot  
## <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 184667 4160 3.25 4 7280  
## 2 265000 4920 4.5 4 112650  
## 3 1390000 660 1 0 225640  
## 4 390000 5800 4.5 5 63162  
## 5 1588359 3360 2.5 2 8752  
## 6 1450000 900 1 2 14043  
## 7 163000 4710 4 4 18498  
## 8 270000 5060 23.5 4 89734  
## 9 200000 6880 4.5 5 288367  
## 10 300000 4490 3.25 4 55303  
## # ... with 312 more rows

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

Out\_L\_week\_7\_housing\_updated[Out\_L\_week\_7\_housing\_updated$large.residual , c("leverage" , "cooks.distance","covariance.ratios") ]

## # A tibble: 322 x 3  
## leverage cooks.distance covariance.ratios  
## <dbl> <dbl> <dbl>  
## 1 0.000342 0.000322 0.999  
## 2 0.000979 0.00119 0.999  
## 3 0.00242 0.00379 1.00   
## 4 0.000966 0.00118 0.999  
## 5 0.000739 0.000623 0.999  
## 6 0.000465 0.000861 0.997  
## 7 0.000606 0.000803 0.998  
## 8 0.148 0.683 1.17   
## 9 0.00269 0.00705 0.998  
## 10 0.000427 0.000352 0.999  
## # ... with 312 more rows

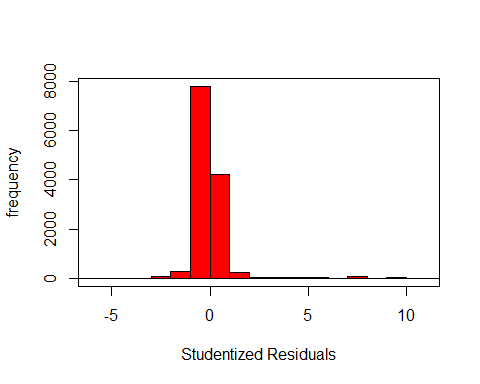
Perform the necessary calculations to assess the assumption of independence and state if the condition is met or not.

dwt(RegModel.4)

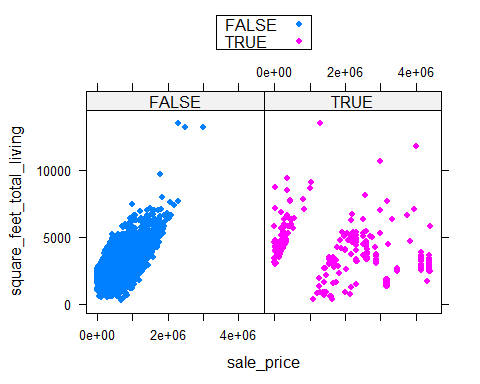
## lag Autocorrelation D-W Statistic p-value  
## 1 0.7309992 0.5379977 0  
## Alternative hypothesis: rho != 0

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.

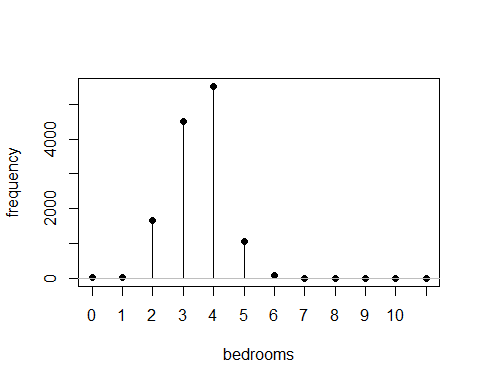
with(Out\_L\_week\_7\_housing\_updated, Hist(standardized.residuals, scale="frequency", breaks="Sturges", col="red",   
 xlab="Studentized Residuals"))



library(lattice, pos = 24)  
xyplot(square\_feet\_total\_living ~ sale\_price | large.residual, groups = large.residual, type = "p", pch = 16,   
 auto.key = list(border = TRUE), par.settings = simpleTheme(pch = 16), scales = list(x = list(relation = 'same'),   
 y = list(relation = 'same')), data = Out\_L\_week\_7\_housing\_updated)



with(Out\_L\_week\_7\_housing\_updated, discretePlot(bedrooms, scale = "frequency"))



scatterplotMatrix(~sale\_price+square\_feet\_total\_living | large.residual,   
 regLine = FALSE, smooth = FALSE, diagonal = list(method = "density"), by.groups = TRUE,   
 data = Out\_L\_week\_7\_housing\_updated)

