DSC 520 Week 7 Multiple Regression Assignment

Dan Wiltse

Jan 26th 2020

## Research Question

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

## 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, Bedrooms, and Bath Full Count as predictors.

#Linear Regression  
saleprice\_lr <- lm(sale\_price~sq\_ft\_lot, data = housing7)  
#Multiple Regression  
saleprice\_mr <- lm(sale\_price~sq\_ft\_lot+bedrooms+bath\_full\_count, data = housing7)  
  
saleprice\_lr

##   
## Call:  
## lm(formula = sale\_price ~ sq\_ft\_lot, data = housing7)  
##   
## Coefficients:  
## (Intercept) sq\_ft\_lot   
## 6.418e+05 8.510e-01

saleprice\_mr

##   
## Call:  
## lm(formula = sale\_price ~ sq\_ft\_lot + bedrooms + bath\_full\_count,   
## data = housing7)  
##   
## Coefficients:  
## (Intercept) sq\_ft\_lot bedrooms bath\_full\_count   
## 1.429e+05 7.227e-01 6.887e+04 1.458e+05

## 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?

For the first variable created (saleprice\_lr) that compared sales price to sq ft of lot, the R value is .119 and the R2 is .014. This means that the simple regression model including only square foot of the lot accounts for only 1.4% of the variation in sales price.

For the 2nd variable created that had multiple variables (saleprice\_mr), the R value is .336 and the R2 value is .113. This means that the multiple regression accounts for 11.3% of the variation in sales prices, so including additional predictors helped explain nearly 10% more of the variance in predicting sales price.

options(scipen = 999)  
summary(saleprice\_lr)

##   
## Call:  
## lm(formula = sale\_price ~ sq\_ft\_lot, data = housing7)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -2016064 -194842 -63293 91565 3735109   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 641821.40609 3799.91526 168.90 <0.0000000000000002 \*\*\*  
## sq\_ft\_lot 0.85099 0.06217 13.69 <0.0000000000000002 \*\*\*  
## ---  
## 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: < 0.00000000000000022

summary(saleprice\_mr)

##   
## Call:  
## lm(formula = sale\_price ~ sq\_ft\_lot + bedrooms + bath\_full\_count,   
## data = housing7)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -3566287 -153218 -51375 69545 3736381   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 142928.9011 14812.9592 9.649 <0.0000000000000002 \*\*\*  
## sq\_ft\_lot 0.7227 0.0591 12.229 <0.0000000000000002 \*\*\*  
## bedrooms 68865.1686 4027.4827 17.099 <0.0000000000000002 \*\*\*  
## bath\_full\_count 145784.6656 5421.9053 26.888 <0.0000000000000002 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 381000 on 12861 degrees of freedom  
## Multiple R-squared: 0.1127, Adjusted R-squared: 0.1125   
## F-statistic: 544.3 on 3 and 12861 DF, p-value: < 0.00000000000000022

## 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?

1. Sq\_ft\_lot has a beta of .101, which means that as sq ft of the lot increases by 1 SD , sale price increases by .101 standard deviations. This is only true if the effects of bedrooms and full bathrooms are held constant.
2. Bedrooms has a standardized beta of .1492, which means as the number of bedrooms increases by 1 SD, sale price will increase by .149 standard deviations. This is only true if the effects of sq ft of the lot and full bathrooms are held constant.
3. Bath full count has a standardized beta of .235, so this means as the number of full bathrooms increases by 1 SD, sale price will increase by .235 standard deviations. This is only true if the effects of bedrooms and sq ft of the lot are held constant.

lm.beta(saleprice\_mr)

## sq\_ft\_lot bedrooms bath\_full\_count   
## 0.1017484 0.1492025 0.2346207

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

See results below. The results of the confidence interval show how close the model comes to predicting the true value for each beta. The closer the intervals are together the better the predictor. Looking at the 3 variables, none of them cross zero (eg where some of the predictors have a negative relationship that cross over the confidence intervals), so this indicates these variables are representative of the true values for each b value.

confint(saleprice\_mr)

## 2.5 % 97.5 %  
## (Intercept) 113893.30200 171964.5001988  
## sq\_ft\_lot 0.60685 0.8385307  
## bedrooms 60970.70465 76759.6325970  
## bath\_full\_count 135156.92628 156412.4048972

## 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.

The ANOVA shows an improvement significant at the .001 level.

anova(saleprice\_mr, saleprice\_lr)

## Analysis of Variance Table  
##   
## Model 1: sale\_price ~ sq\_ft\_lot + bedrooms + bath\_full\_count  
## Model 2: sale\_price ~ sq\_ft\_lot  
## Res.Df RSS Df Sum of Sq F Pr(>F)   
## 1 12861 1866579716716768   
## 2 12863 2073376756946868 -2 -206797040230100 712.43 < 0.00000000000000022 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

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

See #i

## 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.

See #i

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

#Create outlier residual variables  
housing7$residuals <- resid(saleprice\_mr)  
housing7$standardized.residuals <- rstandard(saleprice\_mr)  
#Create influential cases variables used in j:  
housing7$cooks.distance <- cooks.distance(saleprice\_mr)  
housing7$leverage <- hatvalues(saleprice\_mr)  
housing7$covariance.ratios <- covratio(saleprice\_mr)  
#Create variable that finds standardized residuals smaller than -2 and bigger than 2  
housing7$large\_residual <- housing7$standardized.residuals > 2 | housing7$standardized.residuals < -2  
#Find number of records with large residuals  
sum(housing7$large\_residual)

## [1] 329

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

See below for all variables(columns) that have large residuals.

housing7[housing7$large\_residual, c("sale\_price", "sq\_ft\_lot", "bedrooms", "bath\_full\_count", "standardized.residuals")]

## # A tibble: 329 x 5  
## sale\_price sq\_ft\_lot bedrooms bath\_full\_count standardized.residuals  
## <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 265000 112650 4 4 -2.15  
## 2 1900000 37017 4 3 2.67  
## 3 1390000 225640 0 1 2.47  
## 4 1588359 8752 2 2 2.65  
## 5 1450000 14043 3 2 2.10  
## 6 1450000 14043 2 1 2.66  
## 7 270000 89734 4 23 -9.81  
## 8 90000 574992 3 1 -2.16  
## 9 90000 574992 3 1 -2.16  
## 10 2500000 36362 4 2 4.63  
## # ... with 319 more rows

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

According to the book, we should further examine columns with a Cooks distance greater than 1, so row 7 below has a Cook’s distance of 2.38, so should definitely be reviewed.

We should also be on the lookout for any records with a leverage (hat value) 2-3 times larger than the average leverage. And for covariance ratios, will need to look at cases that deviate the expected range of these ratios in the data to see if these are influencing the overall results.

housing7[housing7$large\_residual, c("cooks.distance","leverage", "covariance.ratios")]

## # A tibble: 329 x 3  
## cooks.distance leverage covariance.ratios  
## <dbl> <dbl> <dbl>  
## 1 0.00132 0.00115 1.00   
## 2 0.000616 0.000345 0.998  
## 3 0.00367 0.00241 1.00   
## 4 0.000633 0.000360 0.998  
## 5 0.000134 0.000122 0.999  
## 6 0.000607 0.000343 0.998  
## 7 2.38 0.0900 1.07   
## 8 0.00901 0.00765 1.01   
## 9 0.00901 0.00765 1.01   
## 10 0.000591 0.000110 0.994  
## # ... with 319 more rows

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

The dw statistic is .702, so this does raise alarm bells, as we typically would like this value between 1 and 3, and the author stated the closer this is to 2, the better. The p value is 0, so it shows that this conclusion is significant.

library(boot)  
dwt(saleprice\_mr)

## lag Autocorrelation D-W Statistic p-value  
## 1 0.6487202 0.7025582 0  
## Alternative hypothesis: rho != 0

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

The largest VIF is less than 10 and the tolerance statistics are all well above 0.2, so these results conclude there is no collinearity within the data.

mean(vif(saleprice\_mr))

## [1] 1.070187

vif(saleprice\_mr)

## sq\_ft\_lot bedrooms bath\_full\_count   
## 1.003410 1.103585 1.103567

1/vif(saleprice\_mr)

## sq\_ft\_lot bedrooms bath\_full\_count   
## 0.9966015 0.9061373 0.9061528

## 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.

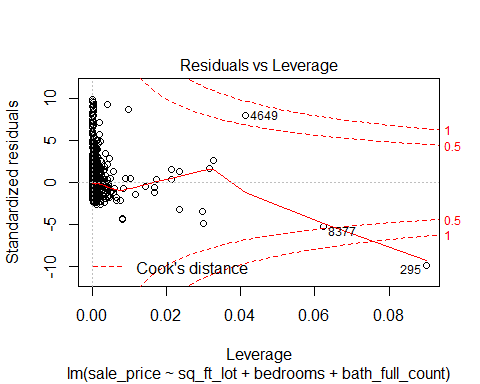
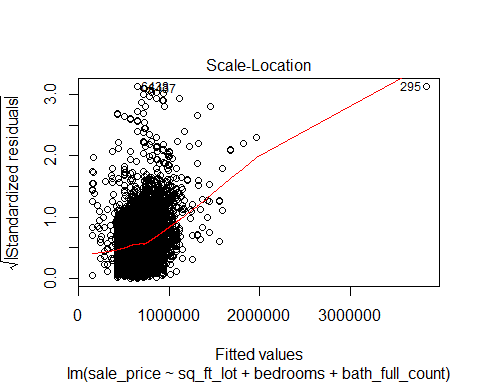
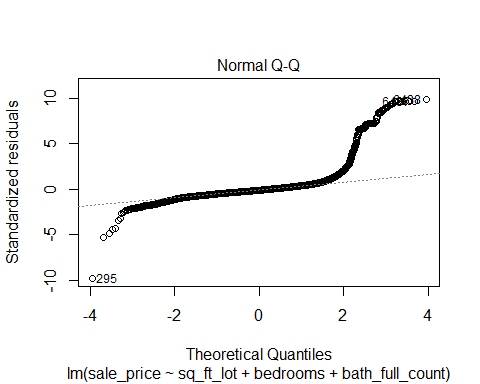
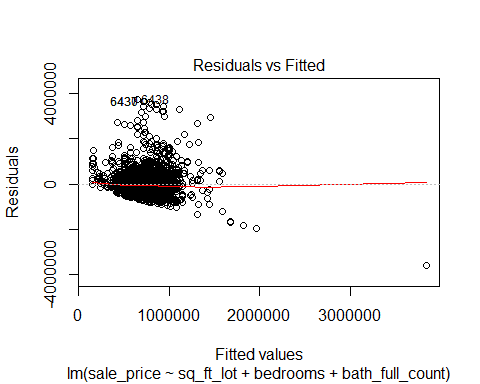
The plot function creates 4 plots. The first plots the residuals versus fitted values. This shows a random pattern of plots, which indicates that assumtions of linearity, randomness and homoscedacity have been met.

The second plot is the Q-Q plot, which shows deivations from normality. If the data was perfectally distributed, the data would all appear on the dotted line, but since you can see the line deviate on the right side of the plot, which indicates a deviation from normality.

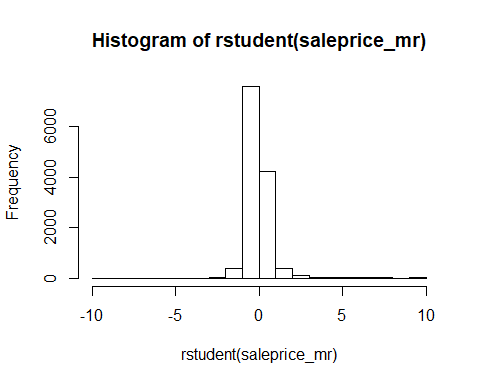
The third plot and forth plots also show the scale location and residuals vs leverage.

Using the hist() function with the residuals, the distribution is roughly normal and not skewed in any particular direction.

plot(saleprice\_mr)



hist(rstudent(saleprice\_mr))



## 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?

Besides the questionable value of the Durbin Watson Test that assesses the assumption of independence, I believe this regression model is unbiased, as the above tests show there are no significant issues with the model or data impacting the results.

So we can likely safely assume as an unbiased regression model, this regresion model would generalize towards any house in the area to predict the sale price.