Exercise 8.2.3

Nicole Buros

May 5th, 2022

Exercise 8.2.3, Housing Data:

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 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.

```
## Load the ggplot2 package
library(readxl)

## Set the working directory to the root of your DSC 520 directory
setwd("C:/Users/njack/OneDrive/Documents/DSC 520/dsc520")
## Read week-7-housing.xlsx file and create data frame, summarize data and type
excel_sheets('data/week-7-housing.xlsx')
```

```
## [1] "Sheet2"
```

```
housing_df <- read_excel('data/week-7-housing.xlsx', sheet=1)
```

1. Explain any transformations or modifications you made to the data set

a. When I started playing around with the data in previous weeks, I changed the name of columns that had spaces in them and replaced the spaces with underscores.

2. 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.

```
SalePrice_lm <-  lm(formula = Sale_Price ~ square_feet_total_living, data = housing_df)
SalePriceV2_lm <-  lm(formula = Sale_Price ~ square_feet_total_living + year_built +
                        bedrooms + bath_full_count + zip5, data = housing_df)
```

a. I chose to add year built, bedrooms, bath count, and zip code as these are all key factors in determining a home's value and therefore sales price.

3. 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(SalePrice_lm)
```

```
##
## Call:
## lm(formula = Sale_Price ~ square_feet_total_living, data = housing_df)
##
## 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(SalePriceV2_lm)
```

```
##
## Call:
## lm(formula = Sale_Price ~ square_feet_total_living + year_built +
##     bedrooms + bath_full_count + zip5, data = housing_df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1718367  -120730   -42444    45575  3905221
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)               -1.349e+08  1.847e+08  -0.731  0.46507
## square_feet_total_living  1.741e+02  4.443e+00  39.180  < 2e-16 ***
## year_built                2.335e+03  2.119e+02  11.023  < 2e-16 ***
## bedrooms                 -1.342e+04  4.541e+03  -2.956  0.00312 **
## bath_full_count           1.712e+04  6.100e+03   2.806  0.00502 **
## zip5                      1.331e+03  1.883e+03   0.707  0.47985
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 357300 on 12859 degrees of freedom
## Multiple R-squared:  0.2194, Adjusted R-squared:  0.2191
## F-statistic:   723 on 5 and 12859 DF,  p-value: < 2.2e-16
```

a. The inclusion of other factors only added a marginal explanation for variation of sales price, and it does seem that square footage is still the biggest factor in the variation. Alone, it accounts for 21% of the variation, while together with other predictors they all account for 22%, meaning the other factors only accounted for an additional 1% of the variation in sales price.

4. 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?

```
library(lm.beta)
lm.beta(SalePriceV2_lm)
```

```
##
## Call:
## lm(formula = Sale_Price ~ square_feet_total_living + year_built +
##     bedrooms + bath_full_count + zip5, data = housing_df)
##
## Standardized Coefficients::
##              (Intercept) square_feet_total_living               year_built
##                       NA              0.426048962              0.099448571
##                 bedrooms          bath_full_count                     zip5
##             -0.029082601              0.027547824              0.005578569
```

a. They each indicate their affect on sales price. For example, as square footage increases by 1 standard deviation, sale price increases by .43 standard deviations.

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

```
confint(SalePriceV2_lm)
```

```
##                                  2.5 %        97.5 %
## (Intercept)              -4.968840e+08  2.270656e+08
## square_feet_total_living  1.653504e+02  1.827666e+02
## year_built                1.920055e+03  2.750645e+03
## bedrooms                 -2.232482e+04 -4.521631e+03
## bath_full_count           5.160265e+03  2.907415e+04
## zip5                     -2.361107e+03  5.022730e+03
```

a. Square feet and year built have relatively tight intervals, meaning it's very likely that our model is representative of the true population. However, zip does cross zero which indicates a bad model and a negative relationship with the outcome.

6. 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(SalePrice_lm, SalePriceV2_lm)
```

```
## Analysis of Variance Table
##
## Model 1: Sale_Price ~ square_feet_total_living
## Model 2: Sale_Price ~ square_feet_total_living + year_built + bedrooms +
##     bath_full_count + zip5
##   Res.Df        RSS Df  Sum of Sq      F    Pr(>F)
## 1  12863 1.6689e+15
## 2  12859 1.6420e+15  4 2.6895e+13 52.656 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

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

```
housing_df$residuals <- resid(SalePriceV2_lm)
housing_df$stand.residuals <- rstandard(SalePriceV2_lm)
housing_df$stude.residuals <- rstudent(SalePriceV2_lm)
housing_df$cooks.distance <- cooks.distance(SalePriceV2_lm)
housing_df$dfbeta <- dfbeta(SalePriceV2_lm)
housing_df$dffit <- dffits(SalePriceV2_lm)
housing_df$leverage <- hatvalues(SalePriceV2_lm)
housing_df$covar <- covratio(SalePriceV2_lm)
head(housing_df)
```

```
## # A tibble: 6 x 32
##   Sale_Date           Sale_Price sale_reason sale_instrument sale_warning
##   <dttm>                   <dbl>       <dbl>           <dbl> <chr>
## 1 2006-01-03 00:00:00     698000           1               3 <NA>
## 2 2006-01-03 00:00:00     649990           1               3 <NA>
## 3 2006-01-03 00:00:00     572500           1               3 <NA>
## 4 2006-01-03 00:00:00     420000           1               3 <NA>
## 5 2006-01-03 00:00:00     369900           1               3 15
## 6 2006-01-03 00:00:00     184667           1              15 18 51
## # ... with 27 more variables: sitetype <chr>, addr_full <chr>, zip5 <dbl>,
## #   ctyname <chr>, postalctyn <chr>, lon <dbl>, lat <dbl>,
## #   building_grade <dbl>, square_feet_total_living <dbl>, bedrooms <dbl>,
## #   bath_full_count <dbl>, bath_half_count <dbl>, bath_3qtr_count <dbl>,
## #   year_built <dbl>, year_renovated <dbl>, current_zoning <chr>,
## #   sq_ft_lot <dbl>, prop_type <chr>, present_use <dbl>, residuals <dbl>,
## #   stand.residuals <dbl>, stude.residuals <dbl>, cooks.distance <dbl>, ...
```

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

```
housing_df$large.standresid <- housing_df$stand.residuals > 2 | housing_df$stand.residuals < -2
```

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

```
sum(housing_df$large.standresid)
```

```
## [1] 328
```

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

```
housing_df[housing_df$large.standresid, c("Sale_Price", "square_feet_total_living",
                                          "year_built", "bedrooms", "bath_full_count",
                                          "zip5", "stand.residuals")]
```

```
## # A tibble: 328 x 7
##    Sale_Price square_feet_total_living year_built bedrooms bath_full_count  zip5
##         <dbl>                    <dbl>      <dbl>    <dbl>           <dbl> <dbl>
```

```
## 1    184667                    4160    2005    4         2 98053
## 2    265000                    4920    2007    4         4 98053
## 3   1390000                     660    1955    0         1 98053
## 4    390000                    5800    2008    5         4 98052
## 5   1588359                    3360    2005    2         2 98053
## 6   1450000                     900    1918    2         1 98052
## 7    163000                    4710    2014    4         2 98053
## 8    270000                    5060    2016    4        23 98053
## 9    200000                    6880    2008    5         1 98053
## 10   300000                    4490    2008    4         2 98052
## # ... with 318 more rows, and 1 more variable: stand.residuals <dbl>
```

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

```
housing_df[housing_df$large.standresid, c("cooks.distance", "leverage", "covar")]
```

```
## # A tibble: 328 x 3
##    cooks.distance leverage covar
##             <dbl>    <dbl> <dbl>
##  1       0.000274 0.000342 0.999
##  2       0.00119  0.00119  0.999
##  3       0.00300  0.00185  0.998
##  4       0.00139  0.00134  0.999
##  5       0.000476 0.000678 0.999
##  6       0.00393  0.00194  0.997
##  7       0.000697 0.000628 0.998
##  8       0.312    0.120    1.13
##  9       0.00583  0.00300  0.998
## 10       0.000355 0.000509 0.999
## # ... with 318 more rows
```

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

```
library(car)
```

```
## Loading required package: carData
```

```
dwt(SalePriceV2_lm)
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1       0.7209806     0.5580296       0
##  Alternative hypothesis: rho != 0
```

a. Since the value is less than 1, and the p-value is 0, it is not met.

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

```r
vif(SalePriceV2_lm)
```

```
## square_feet_total_living                year_built                  bedrooms
##                 1.948022                  1.340997                  1.594809
##          bath_full_count                      zip5
##                 1.587703                  1.026922
```

```r
1/vif(SalePriceV2_lm)
```

```
## square_feet_total_living                year_built                  bedrooms
##                0.5133412                 0.7457141                 0.6270344
##          bath_full_count                      zip5
##                0.6298406                 0.9737843
```

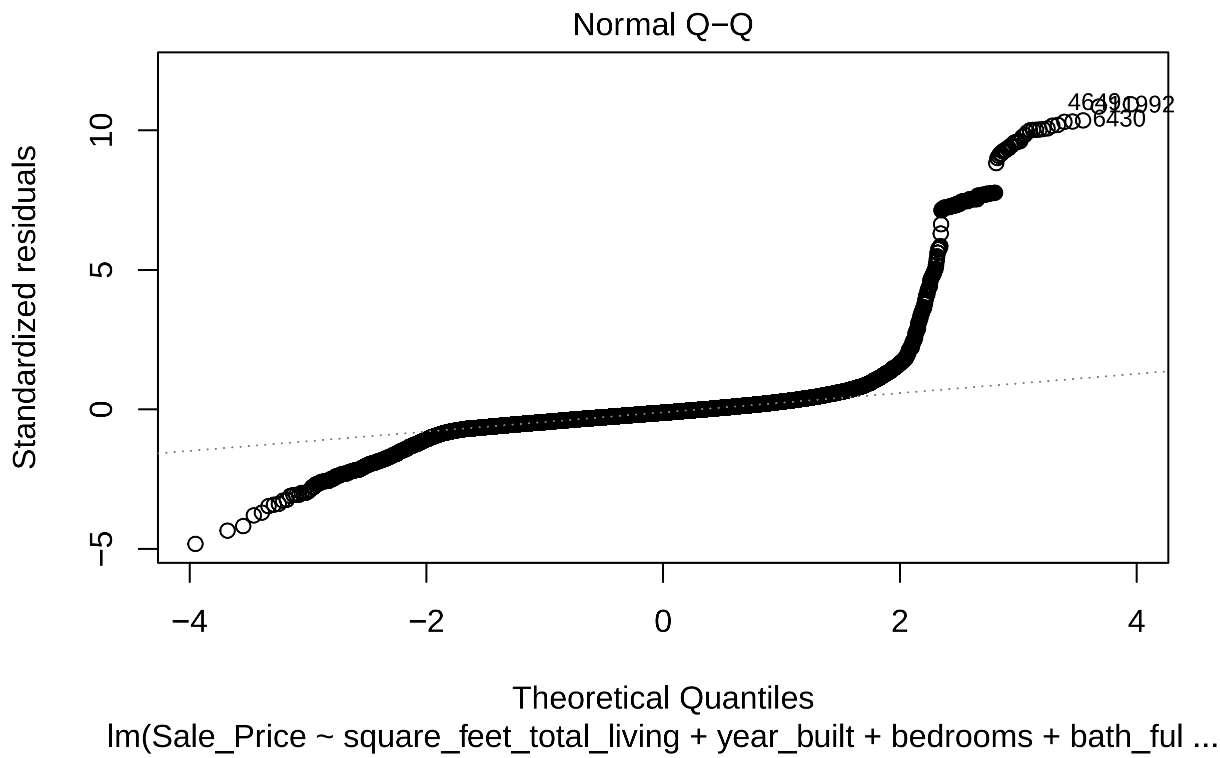
```r
mean(vif(SalePriceV2_lm))
```

```
## [1] 1.49969
```

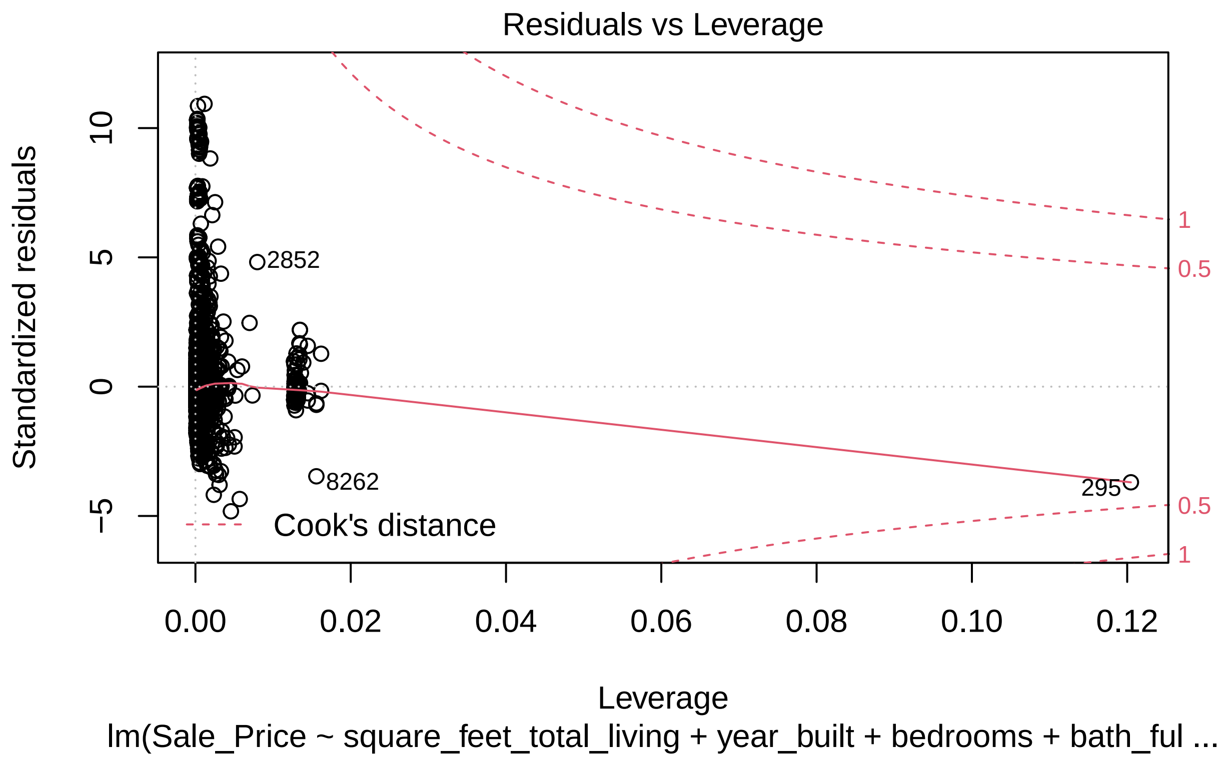
a. For this model, all VIF values are well below 10, all tolerance statistics well above 0.2, and the average VIF is still very close to 1 which all indicate there is no collinearity.

14. 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.

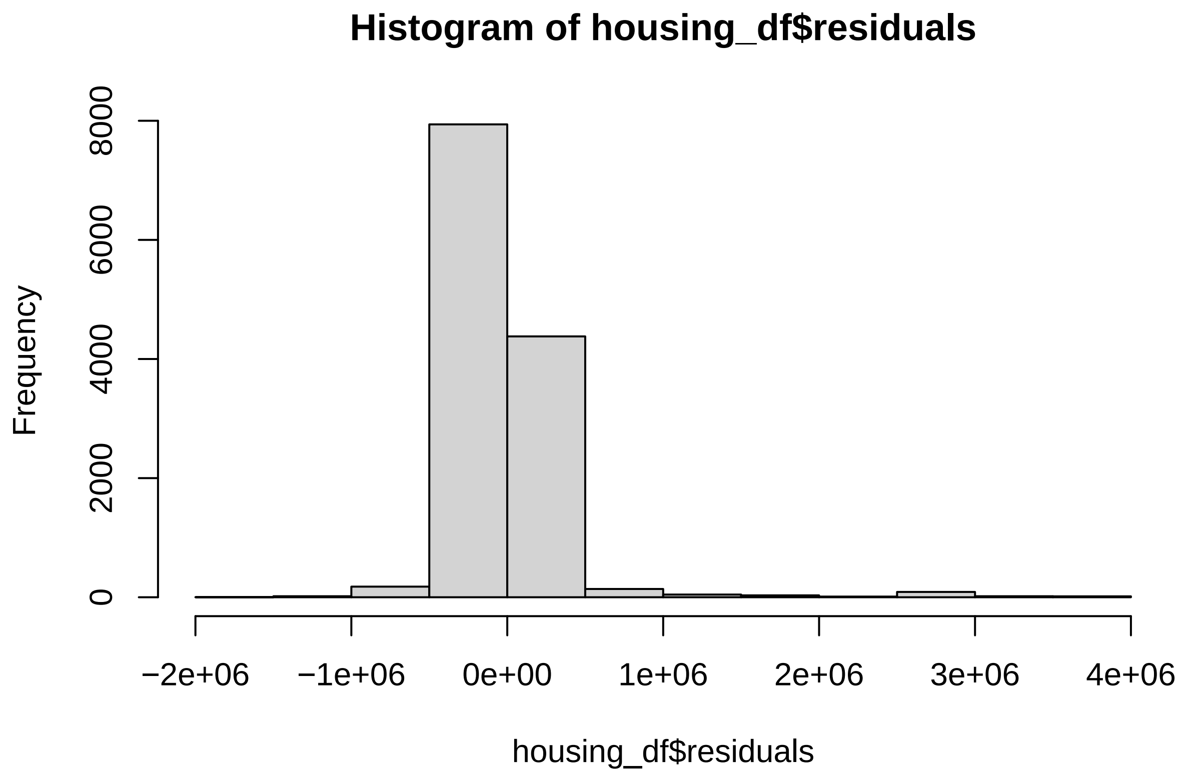
```r
plot(SalePriceV2_lm)
```


Normal Q−Q
Standardized residuals
Theoretical Quantiles
lm(Sale_Price ~ square_feet_total_living + year_built + bedrooms + bath_ful ...
−5
0
5
10
−4
−2
0
2
4
4649
10992
6430


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
2.5
3.0
500000
1000000
1500000
2000000
2500000
Fitted values
lm(Sale_Price ~ square_feet_total_living + year_built + bedrooms + bath_ful ...




```
hist(housing_df$residuals)
```



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

a. Overall, I think the model relatively unbiased and representative of the entire population.

References

- Discovering Statistics Using R (Field, Miles, and Field 2012)

Field, A., J. Miles, and Z. Field. 2012. *Discovering Statistics Using r*. SAGE Publications. https://books.google.com/books?id=wd2K2zC3swIC.