ASSIGNMENT 7 - Exercise 12: Housing Data

Ragunath Gunasekaran

2020-10-20

a. Explain why you chose to remove data points from your 'clean' dataset.

Data points like sale_reason, sale warning, Building_grade, sale_instrument and Etc do not provide proper detail to the housing data Analytics or provide any insights to correlation.

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.

```
squarefeet_lm <- lm(`Sale Price` ~ sq_ft_lot, housing_df)
```

```
SalePrice_lm <-  lm(`Sale Price` ~ sq_ft_lot + year_built + square_feet_total_living + bedrooms, housing
```

Show in New WindowClear OutputExpand/Collapse Output Call: lm(formula = `Sale Price` ~ sq_ft_lot + year_built + square_feet_total_living + bedrooms, data = housing_df)

Coefficients: (Intercept) sq_ft_lot year_built square_feet_total_living
-5.333e+06 2.729e-01 2.801e+03 1.717e+02
bedrooms
-9.085e+03

---

1. Bedrooms - Sale Price increases when square feet of the lot increases
2. Bedrooms - Sale Price increases when more number of bedrooms in the house
3. Year Built - Sale Price may decrease when the year built decrese (older)of the property
4. Total Square Foot - Sale Price may increase when more total square feet in the living area

---

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?

```
r summary(squarefeet_lm)
## ## Call: ## lm(formula = `Sale Price` ~ sq_ft_lot, data = housing_df) ## ##
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
```

R-squared: 0.01435, Adjusted R-squared: 0.01428
A higher R-squared value indicates a higher amount of variability being explained by our model. The R-squared value is a measure of variability in the outcome obtained by the predictors.1.435% of the variation in Sale Price. (Source - Ref 6)
The Adjusted R-Squared provides just 0.01% variance, The cross-validity is extremely good.

```
summary(SalePrice_lm)
```

```
##
## Call:
## lm(formula = `Sale Price` ~ sq_ft_lot + year_built + square_feet_total_living +
##     bedrooms, data = housing_df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2147040  -119970   -41454    45830  3766517
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)              -5.333e+06  4.046e+05 -13.181  < 2e-16 ***
## sq_ft_lot                 2.729e-01  5.917e-02   4.612 4.02e-06 ***
## year_built                2.801e+03  2.031e+02  13.789  < 2e-16 ***
## square_feet_total_living  1.717e+02  4.425e+00  38.809  < 2e-16 ***
## bedrooms                 -9.085e+03  4.557e+03  -1.994   0.0462 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 357100 on 12860 degrees of freedom
## Multiple R-squared:  0.2202, Adjusted R-squared:   0.22
## F-statistic: 907.9 on 4 and 12860 DF,  p-value: < 2.2e-16
```

Multiple R-squared: 0.2202, Adjusted R-squared: 0.22 The value for R-squared increases to 22.2% of the variance in the Sale Price.

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?

```
r library(QuantPsyc)
## Warning: package 'QuantPsyc' was built under R version 4.0.3
## Loading required package: boot
## Warning: package 'boot' was built under R version 4.0.2
## Loading required package: MASS
## Warning: package 'MASS' was built under R version 4.0.2
## ## Attaching package: 'QuantPsyc'
## The following object is masked from 'package:base': ## ##     norm
r lm.beta(SalePrice_lm)
##                sq_ft_lot               year_built square_feet_total_living ##
0.03842553               0.11928935               0.42036390 ##
bedrooms ##              -0.01968450 standardized regression coefficients to objects created by lm
```

sq_ft_lot - Sale Price increases by 0.03842553 SD (+ correlation) year_built - Sale Price increases by 0.11928935 SD (+ correlation) square_feet_total_living - Sale Price increases by 0.42036390 SD (+ correlation) bedrooms - Sale Price decreases by 0.01968450 SD (- correlation)

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

```
confint(SalePrice_lm)
```

```
##                                 2.5 %        97.5 %
## (Intercept)              -6.125803e+06 -4.539774e+06
## sq_ft_lot                 1.569387e-01  3.889127e-01
## year_built                2.403070e+03  3.199472e+03
## square_feet_total_living  1.630620e+02  1.804098e+02
## bedrooms                 -1.801844e+04 -1.525239e+02
```

confint is a generic function. The default method assumes normality, and needs suitable coef and vcov methods to be available (Source - Ref 5) 2.5 % 97.5 % (Intercept) -6.125803e+06 -4.539774e+06 sq_ft_lot 1.569387e-01 3.889127e-01 year_built 2.403070e+03 3.199472e+03 square_feet_total_living 1.630620e+02 1.804098e+02 bedrooms -1.801844e+04 -1.525239e+02

Square Feet of total living area and bedrooms have confidence intervals are close, Representative of the true population values.

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(squarefeet_lm, SalePrice_lm)
```

```
## Analysis of Variance Table
##
## Model 1: `Sale Price` ~ sq_ft_lot
## Model 2: `Sale Price` ~ sq_ft_lot + year_built + square_feet_total_living +
##     bedrooms
##   Res.Df        RSS Df  Sum of Sq      F    Pr(>F)
## 1  12863 2.0734e+15
## 2  12860 1.6404e+15  3 4.3302e+14 1131.6 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Considering the Values of 1131.6 < 2.2e-16, the multiple regression model is better fit than the Linear regression model.

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.

```
SalePrice_DF <- as.data.frame(resid(SalePrice_lm))
SalePrice_DF$residuals <- resid(SalePrice_lm)
SalePrice_DF$standardized.residuals <- rstandard(SalePrice_lm)
SalePrice_DF$studentized.residuals <- rstudent(SalePrice_lm)
SalePrice_DF$cooks.distance <- cooks.distance(SalePrice_lm)
SalePrice_DF$dfbeta <- dfbeta(SalePrice_lm)
SalePrice_DF$dffit <- dffits(SalePrice_lm)
SalePrice_DF$leverage <- hatvalues(SalePrice_lm)
SalePrice_DF$covariance.ratios <- covratio(SalePrice_lm)
head(SalePrice_DF, 10)
```

```
##    resid(SalePrice_lm)  residuals standardized.residuals studentized.residuals
## 1            -28204.07  -28204.07            -0.07897577           -0.07897272
## 2            -96348.73  -96348.73            -0.26979364           -0.26978391
## 3           -102508.02 -102508.02            -0.28703607           -0.28702583
## 4            -13688.58  -13688.58            -0.03833275           -0.03833127
## 5            -65925.32  -65925.32            -0.18460565           -0.18459872
## 6           -779159.11 -779159.11            -2.18198477           -2.18230394
## 7            138578.25  138578.25             0.38809812            0.38808530
## 8             27981.29   27981.29             0.07835585            0.07835282
## 9           -237855.55 -237855.55            -0.66615541           -0.66614100
## 10           -16069.75  -16069.75            -0.04500098           -0.04499923
##    cooks.distance dfbeta.(Intercept) dfbeta.sq_ft_lot dfbeta.year_built
## 1    1.745138e-07       1.703704e+02     5.355628e-06     -8.416208e-02
## 2    2.281235e-06       7.239735e+02     1.875237e-05     -3.587384e-01
## 3    2.044510e-06      -4.044122e+02     5.081846e-05      2.083279e-01
## 4    8.128986e-08      -1.815548e+02     6.721463e-06      8.937857e-02
## 5    1.289477e-06      -3.020882e+02     9.202416e-06      1.427482e-01
## 6    3.240312e-04      -3.977274e+02     8.066552e-04      2.552474e-01
## 7    1.315291e-05      -1.492054e+02     2.273868e-04      4.130629e-02
## 8    2.945096e-07       2.176835e+02    -1.407910e-05     -1.112233e-01
## 9    4.521074e-05      -3.857019e+03     1.685813e-04      1.955383e+00
## 10   1.143199e-07      -1.099903e+02    -2.605697e-05      5.618261e-02
##    dfbeta.square_feet_total_living dfbeta.bedrooms         dffit     leverage
## 1                     7.884791e-04   -1.997648e+00 -0.0009340778 0.0001398786
## 2                     2.439747e-03   -6.643317e+00 -0.0033771814 0.0001566782
## 3                    -1.207844e-03   -4.834394e+00 -0.0031971584 0.0001240602
## 4                     1.727672e-04    5.089786e-01 -0.0006375090 0.0002765325
## 5                     4.890778e-03   -4.580985e-02 -0.0025390746 0.0001891522
## 6                    -1.348186e-01    4.394666e+01 -0.0402570485 0.0003401775
## 7                     2.729358e-03    1.887867e+01  0.0081092657 0.0004364345
## 8                     3.909937e-03   -9.934490e-01  0.0012134390 0.0002397852
## 9                    -5.439222e-02    2.179593e+01 -0.0150347573 0.0005091425
## 10                    2.521125e-04   -9.464089e-01 -0.0007560126 0.0002821794
##    covariance.ratios
## 1          1.0005264
## 2          1.0005173
## 3          1.0004810
## 4          1.0006650
## 5          1.0005649
## 6          0.9988782
## 7          1.0007671
## 8          1.0006264
## 9          1.0007258
## 10         1.0006705
```

---

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.

```
SalePrice_DF$large.residuals <- SalePrice_DF$standardized.residuals > 2 |
SalePrice_DF$standardized.residuals < 2
```

---

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

```
sum(SalePrice_DF$large.residuals)
```

```
## [1] 12865
```

---

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

```
head(SalePrice_DF$large.residuals, 50)
```

```
##  [1] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
## [16] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
## [31] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE
## [46] TRUE TRUE TRUE TRUE TRUE
```

---

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

```
head(SalePrice_DF[SalePrice_DF$large.residuals, c("cooks.distance", "leverage", "covariance.ratios")],
```

```
##    cooks.distance     leverage covariance.ratios
## 1    1.745138e-07 0.0001398786         1.0005264
## 2    2.281235e-06 0.0001566782         1.0005173
## 3    2.044510e-06 0.0001240602         1.0004810
## 4    8.128986e-08 0.0002765325         1.0006650
## 5    1.289477e-06 0.0001891522         1.0005649
## 6    3.240312e-04 0.0003401775         0.9988782
## 7    1.315291e-05 0.0004364345         1.0007671
## 8    2.945096e-07 0.0002397852         1.0006264
## 9    4.521074e-05 0.0005091425         1.0007258
## 10   1.143199e-07 0.0002821794         1.0006705
```

Zero cases are problematic

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

```
library(carData)
```

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

```r
library(car)
```

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

```
##
## Attaching package: 'car'
```

```
## The following object is masked from 'package:boot':
##
##     logit
```

```r
dwt(SalePrice_lm)
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1       0.7209323     0.5581258       0
##  Alternative hypothesis: rho != 0
```

The D-W Statistic Value is 0.5581258. The value is less than 1. So the assumption of independence is not met.

---

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

```r
## assumption of no multicollinearity

vif(SalePrice_lm)
```

```
##                sq_ft_lot               year_built square_feet_total_living
##                 1.144597                 1.234185                 1.934814
##                 bedrooms
##                 1.607780
```

```r
## calculate tolerance = 1 / VIF

1/vif(SalePrice_lm)
```

```
##                sq_ft_lot               year_built square_feet_total_living
##                0.8736699                0.8102512                0.5168455
##                 bedrooms
##                0.6219757
```

```r
## Average VIF

mean(vif(SalePrice_lm))
```

```
## [1] 1.480344
```

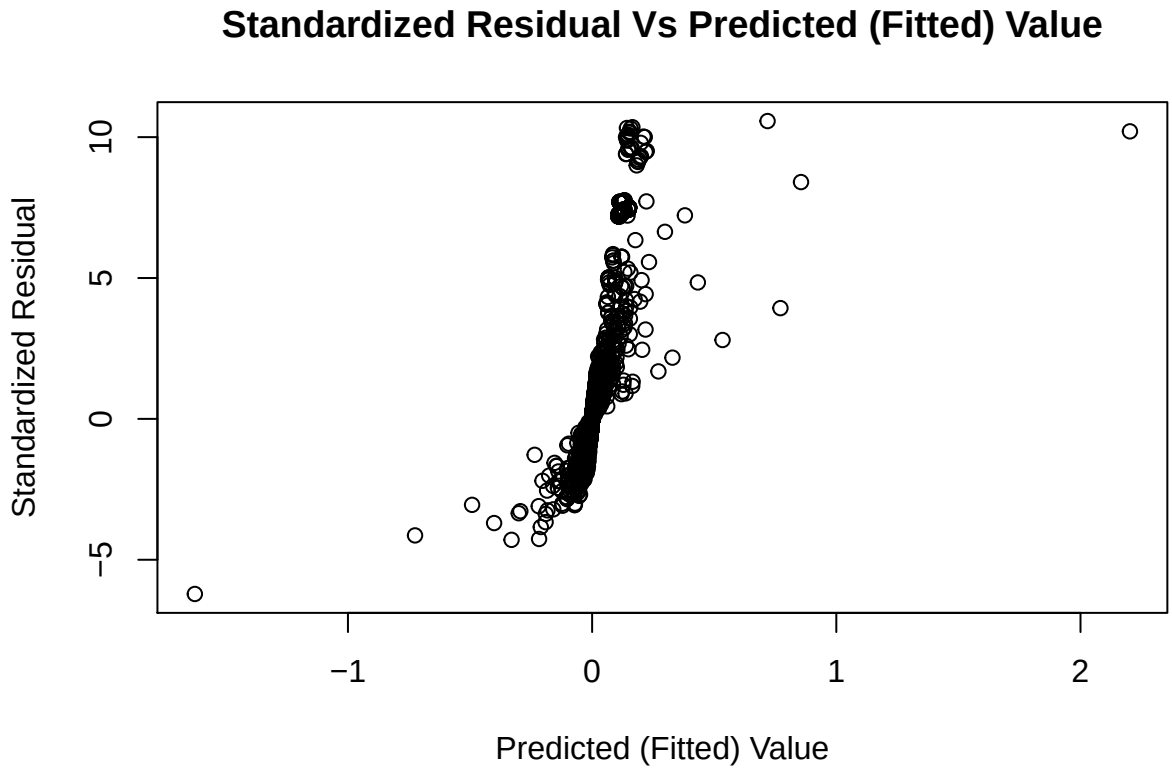
The VIF is less than 10 . no cause of concern. Tolerance is not below 0.2, no potential problem.

So the assumption of no multicollinearity

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.

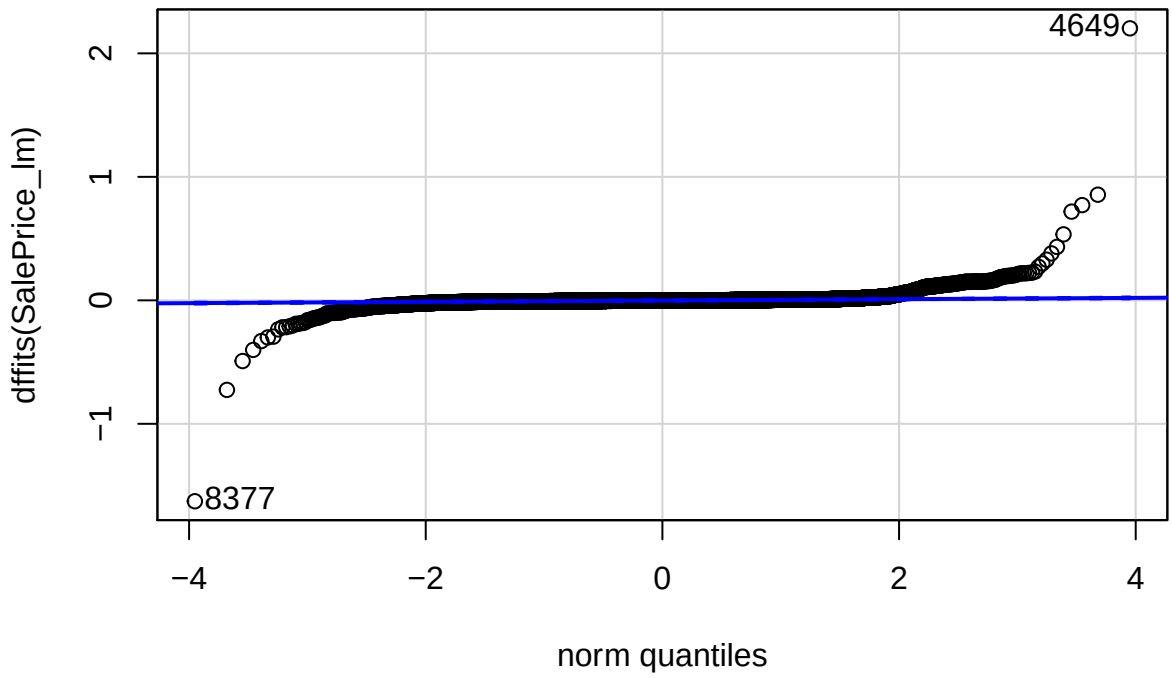
```
## Standardized Residual (y)
## Predicted (Fitted) Value (x)

plot(dffits(SalePrice_lm), rstandard(SalePrice_lm), xlab = "Predicted (Fitted) Value", ylab = "Standard
```




```
## qqplot()
## Predicted (Fitted) Value (x-axis)

qqPlot(dffits(SalePrice_lm))
```




```
## [1] 4649 8377
```

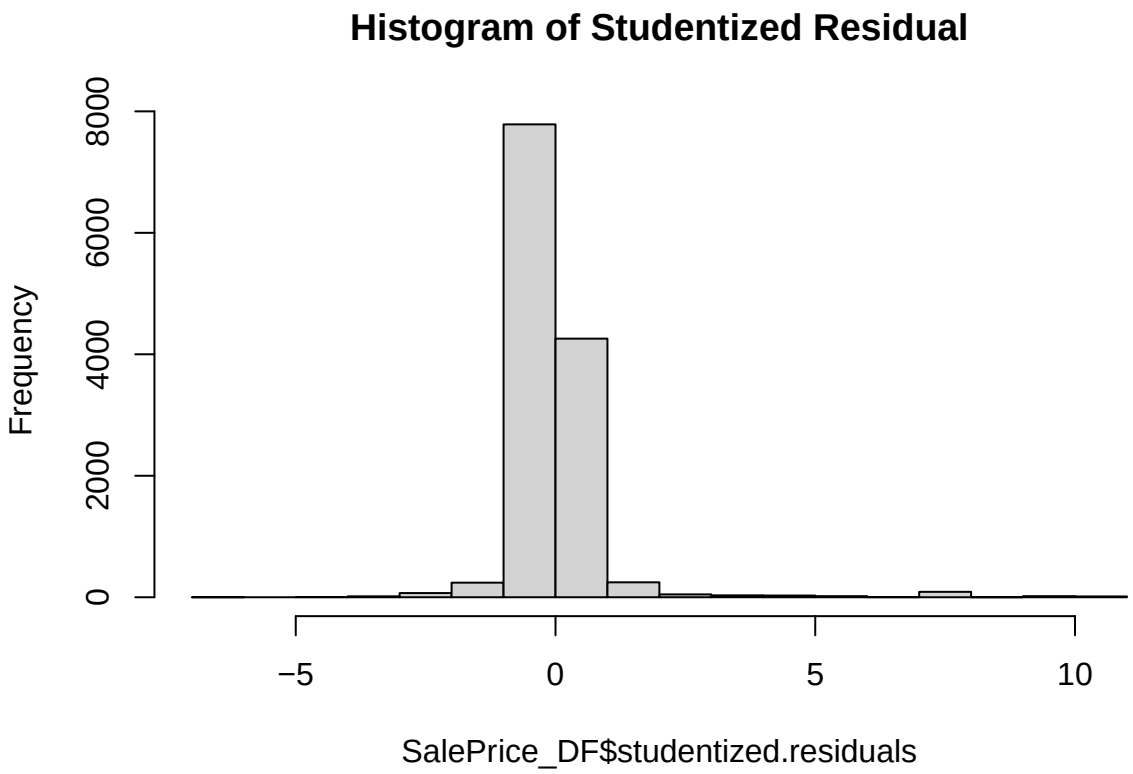
Q-Q Plot produced by the plot() function shows few data points deviate from the normality.

The two values:

1. Data Point - 4649, 8377

```
## hist()
## Studentized Residual (x-axis)

hist(SalePrice_DF$studentized.residuals, main = "Histogram of Studentized Residual")
```



Normal distribution - Right skewed.

---

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?

Yes. Regression model is unbiased whose Average VIF (1.480344) > 1.

References

1. Lander, J. P. 2014. R for Everyone: Advanced Analytics and Graphics. Addison-Wesley Data and Analytics Series. Addison-Wesley. https://books.google.com/books?id=3eBVAgAAQBAJ.
2. R Core Team. 2020. R: A Language and Environment for Statistical Computing. Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.
3. Xie, Yihui. 2016. Bookdown: Authoring Books and Technical Documents with R Markdown. Boca Raton, Florida: Chapman; Hall/CRC. https://github.com/rstudio/bookdown.
4. https://bookdown.org/yihui/rmarkdown-cookbook
5. https://stat.ethz.ch/R-manual/R-devel/library/stats/html/confint.html
6. https://www.analyticsvidhya.com/blog/2020/07/difference-between-r-squared-and-adjusted-r-squared/