Reselection of Continuous Variables

Now that I’ve gotten a firm grasp of the iterative process of developing starting data values, I would like to simplify my approach so I can concentrate on learning. Before I had selected most of the variables to implement in my model, but there were a few issues.

1. It became overwhelming to try to understand the effects of those variables in the model.
2. I could not tell which variables were adding nothing to the model.
3. A lot of those variables were repeated information (ie, totalsqft and 1fltotalsqft)

So the update I will be making from my previous selection is EDA is I will choose what I believe to be the four most critical continuous variables that do not overlap each other.

1. LotArea – I did not include this in the previous analysis and I think that was a mistake. The price of the house most certainly includes the amount of yard included with it.
2. FirstFlrSF – This will represent all square footage for the house. Below I will present my drop conditions for this selection.
3. GarageArea – This is the final variable which represents another value adder.
4. TotalBsmtSF – I only include basements because there were so few houses available without basements.

It total, all four, according to my drop conditions, will represent all possible area that has value.

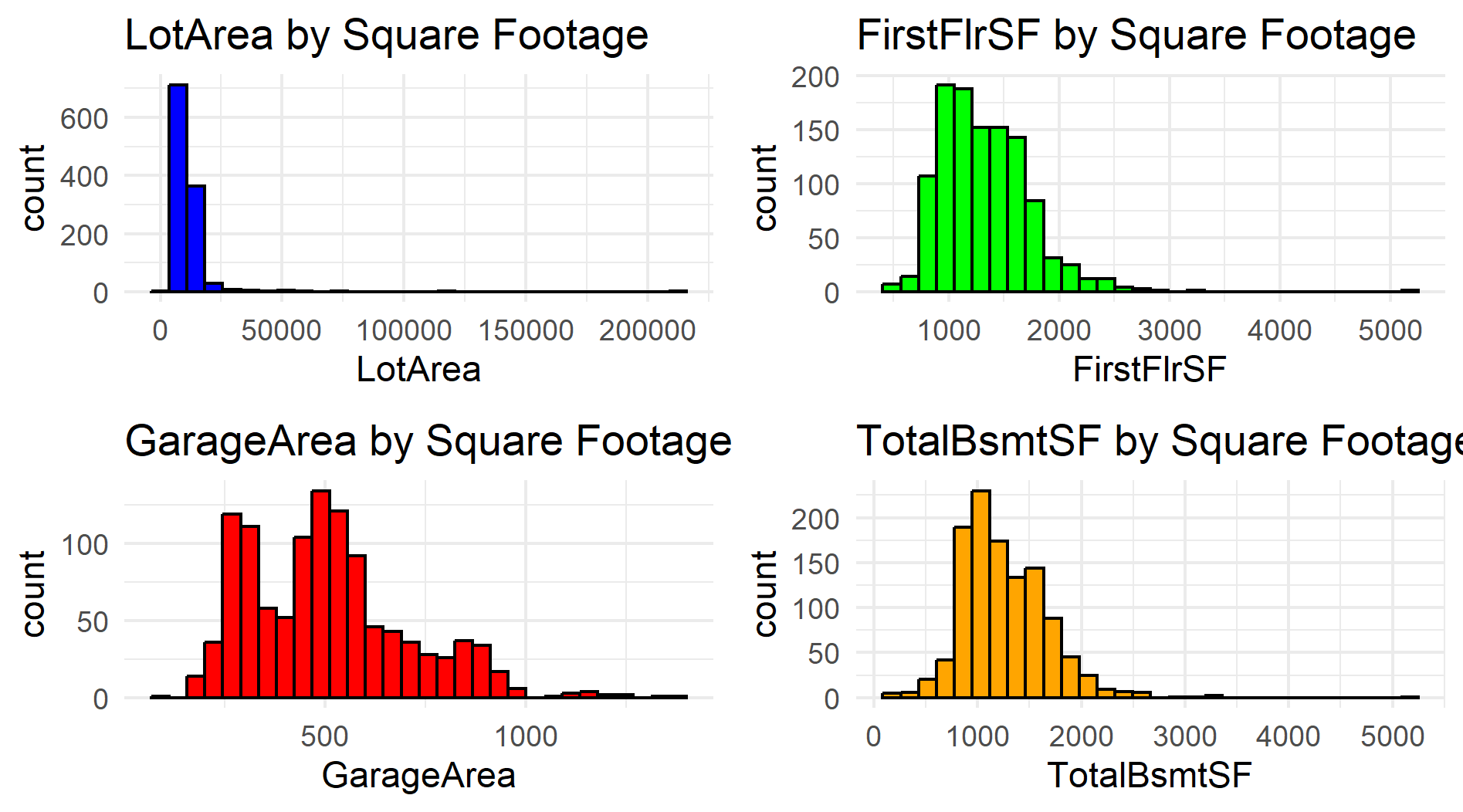
Drop Conditions

1. LotArea > 0
2. HouseStyle = 1Story
3. BldType = 1Fam
4. TotalBsmtSF > 0
5. GarageArea > 0

These drop conditions ensure that all houses selected will be single story family residences, a ranch style, with basements and garages.

I will also be selecting TotRmsAbvGrd, FullBath, HalfBath, and Neighbourhood for their categorical and quantitative values

Histograms



*Figure 1*

The four continuous variables I selected shown in Figure 1, include outliers which could show a right skew, meaning people prefer smaller areas. Total basement square footage mirrors the first floor showing that the foundations of these houses are built using the basement. Lot area shows most houses have reasonably sized lots, just enough for a front and back yard, while several have several acres of land, but do not constitute the majority of the houses.

Garages have a bimodal distribution, most likely single and double wide garages.

Summary

A screenshot of a computer screen

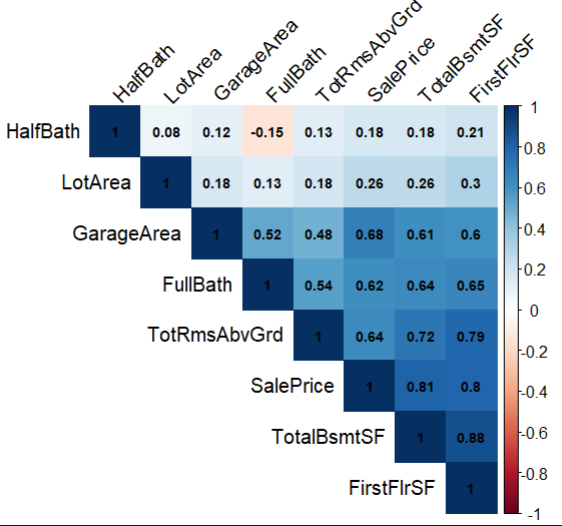
Description automatically generated

*Figure 2*

What stands out in the Ames region, is that there are clearly wealthy people who own lots of land twenty times larger than the mean. These are still ranch style homes, so it is likely the larger areas are farms.

Most lots are around 11,308 sqft, first floor areas of 1325 sqft, and basements just a bit smaller.

Correlations

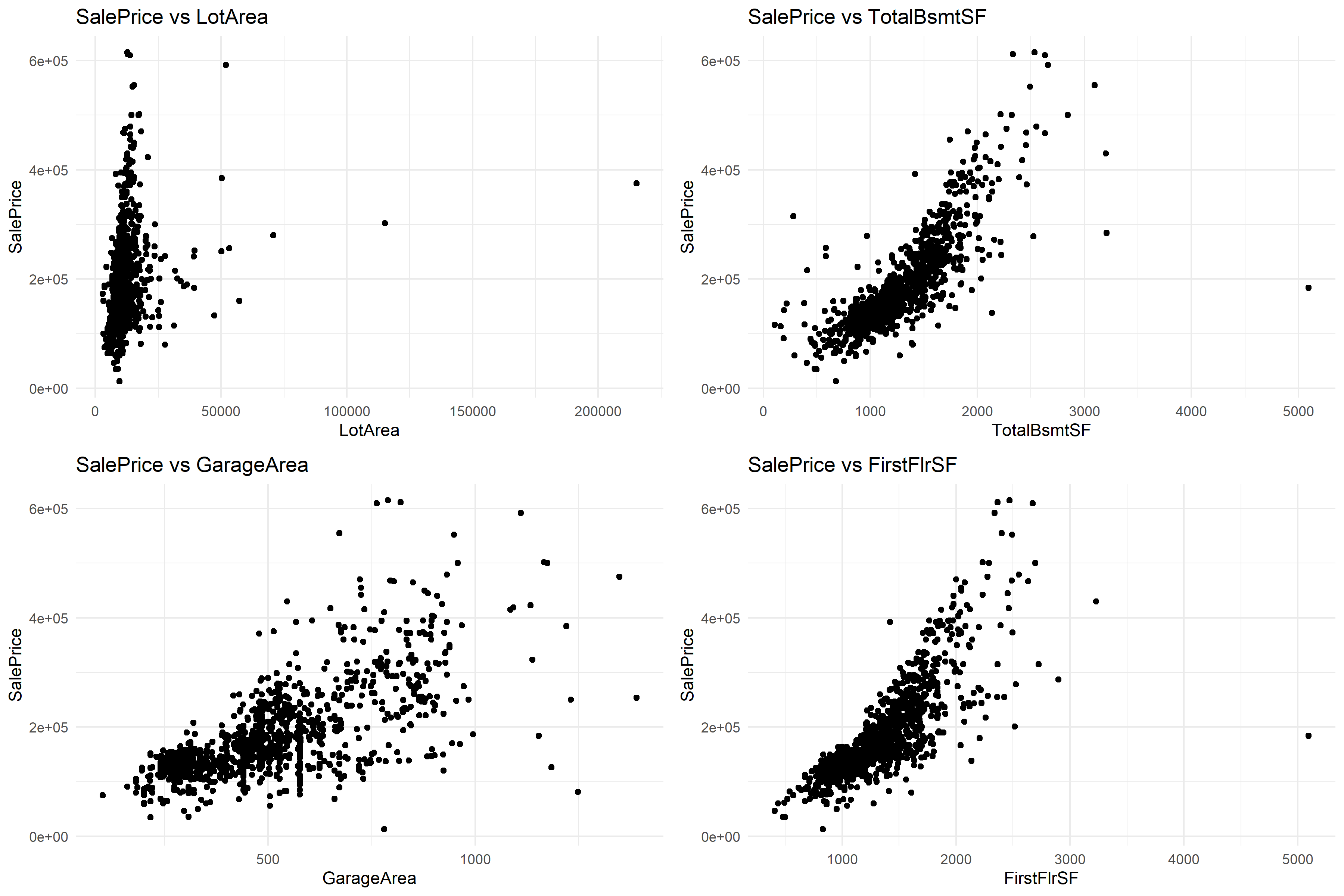


*Figure 3*

Like earlier, Total Basement square footage and First Floor square footage are highly correlated. All variables exceed 0.6 concerning Sale Price, but Lot Area and Half Bath are only slightly correlated. Size seems to matter the most and the number of Full Baths.

Size based variables correlate with other ones which gives me confidence our model will be able to predict Sale Price nicely.

Scatterplots



*Figure 4*

Figure 4 shows all continuous variables have positive relationship. Although there are strong correlations between each one there is noticeable variability especially with Lot Area, a few of the outliers could throw off the model as well.

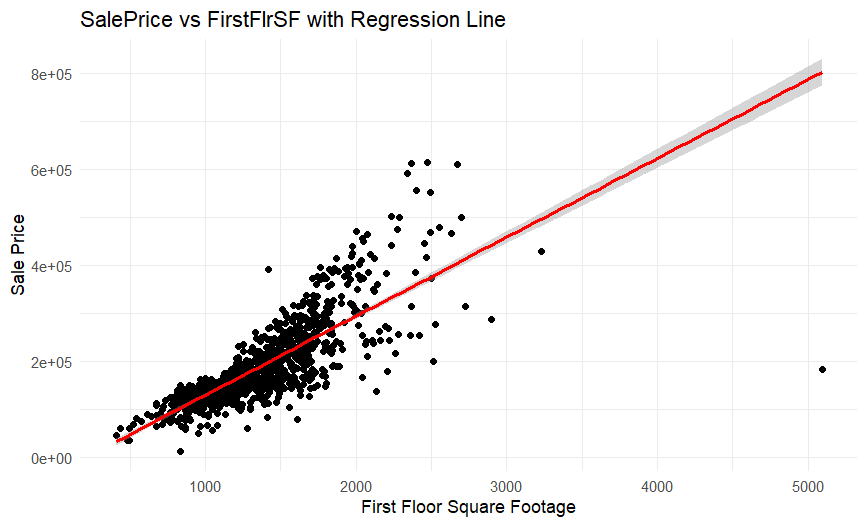
New Variables

I created one new variable type named Square Foot Per Dollar. The calculation is SalePrice / (LotArea + TotalBsmtSF + GarageArea + FirstFlrSF)) so that each house has all square footage included in the calculation. My hope is it will give the most accurate indication by smoothing out how much each square foot is worth.

Assignment Tasks

1.

I’ve selected FirstFlrSF as the best continuous explanatory variable to predict Y (SalePrice). I think it’s a straightforward and well known variable, and although when I ran the correlations, basement square footage beat it out slightly at 0.81 to 0.8, if I were to ever expand this model to include more, lots of basements can vary in size, but most people buy a house for its living area. Also, the quality of the basement could affect it as many basements are not livable.

1. 

A computer screen shot of a number

Description automatically generated

1. The model in equation form is SalePrice = -34294 + 164.242 \* FirstFlrSF

The intercept (-34294.9) is the estimated Sale Price when FirstFlrSF is zero. Obviously when there is no square footage, the house will be less than worthless. The slope (164.242) is that each square footage that increases in the house, the amount of the price will increase by $164.24

1. 63.88% of the variation in Sale Price can be explained by the variation in the First Floor Square Footage of houses. This means the prices of houses can be predicted by knowing just the First Floor Square footage. There is still a large percentage that is unaccounted for, however this is huge.
2. A screenshot of a computer

   Description automatically generated

Null Hypothesis H0: B1 = 0 : There is no relationship between FirstFlrSF and SalePrice

Alternative Hypothesis HA: B1 != 0: There is a relationship between FirstFlrSF and SalePrice

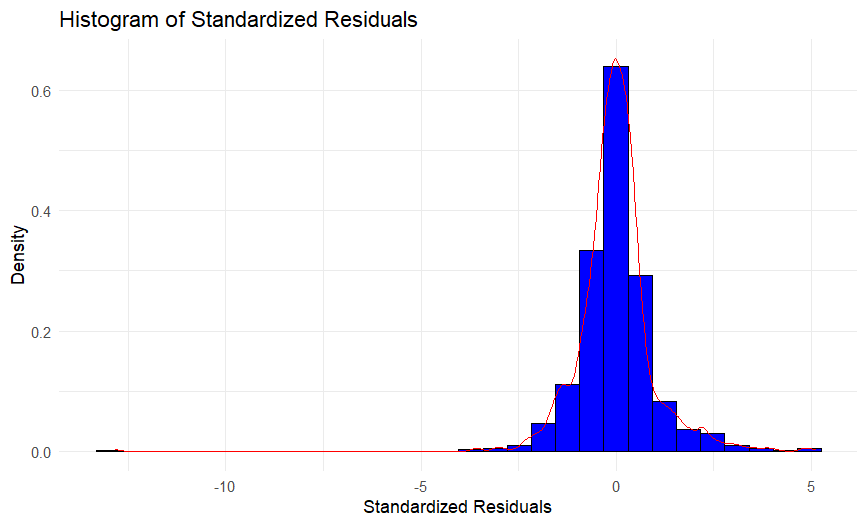
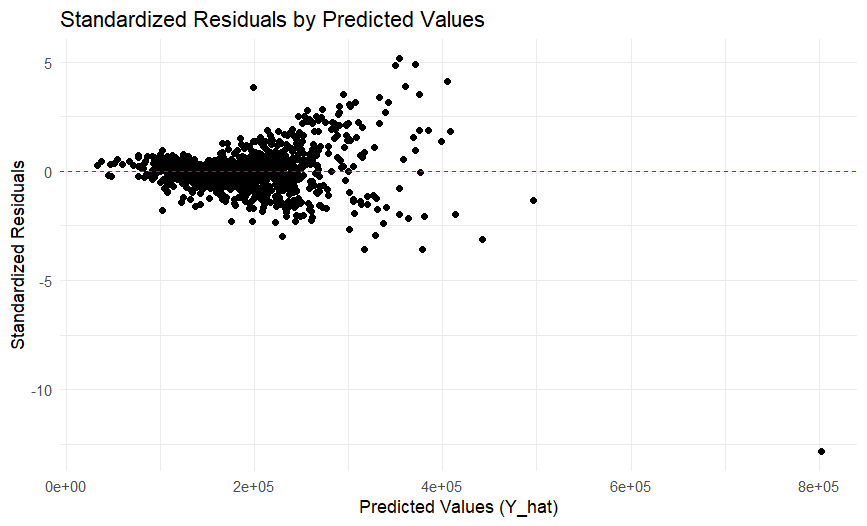
The P-Value is well below the accepted value of 0.05 as 2.2e-16, therefore we must reject the Null hypothesis.

Overall Omnibus Model

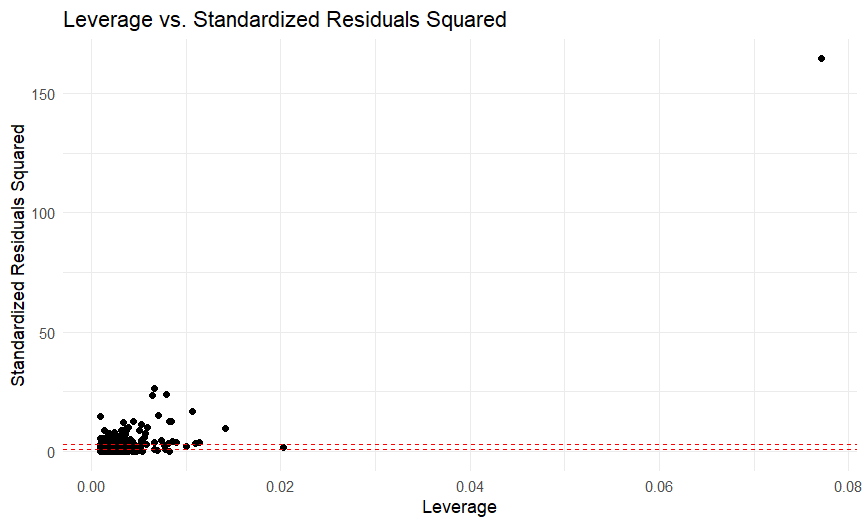
Null Hypothesis H0: B1 = 0 The model provides no explanatory power.

Alternative Hypothesis: HA: B1 != 0 The model does provide explanatory power.

A larger F value provides a better fit and explanatory power of B1. Again, our value of 2.2e-16 is well below the 0.05 needed, and we must reject the null hypothesis and say that it does have explanatory power.

1. 
2. The histogram shows a nice curve for normality, and the scatterplot is nicely evenly distributed. Although there are a few outliers, it does not seem to affect much.

I do not believe I will need to fix this by re weighing the variable for heteroscedasticity.

G.Nothing is too concerning looking at the leverage vs standardized R^2 plot. I see this one over represented house popping up over and over again, it probably is not going to affect too much, however it could cause problems, and dropping it in the future may improve the model

2.

A. A screenshot of a computer program

Description automatically generated

Prediction equation: SalePrice = -103.464 + 0.107\*FirstFlrSF + 3.948\*OverallQual

The intercept (-118,000) represents the estimated Sale Price when both FirstFlrSF and OverallQual are at 0.

FirstFlrSF (80.29) says that for each additional square foot in the first floor square footage, while overall quality remains the same, will increase by $80.29.

OverallQual (32,350) says for every one-unit increase in overall quality, assuming FirstFlrSF remains constant, the price will increase by 32,350.

The largest change is that I must assume that the other coefficients will not change to get an accurate prediction for any given coefficient. Another large change is that FirstFlrSF coefficient has halved compared to model 2.

B. The R^2 was 0.79 which means with the addition of OverallQual the variability can be explained by the combination of FirstFlrSF and OverallQual, which is higher than 63.88% in Model 2. This means that this model works better as a predicator compared to just FirstFlrSF.

C. A screenshot of a computer

Description automatically generated

D.

Null Hypothesis: H0: Bi = 0

Alternative Hypothesis: HA: Bi != 0

The null hypothesis states that any given variable has no affect on the response variable.

The alternative hypothesis states that any given variable does have an effect on the response variable.

FirstFlrSF: The p-value is less than 0.05 with a value of 2.2e-16 so we must reject the null hypothesis.

OverallQual: The p-value is less than 0.05 with a value of 2.2e-16 so we must reject the null hypothesis.

Overall Omnibus Model

Null Hypothesis H0: Bi = 0 The model provides no explanatory power.

Alternative Hypothesis: HA: Bi != 0 The model does provide explanatory power.

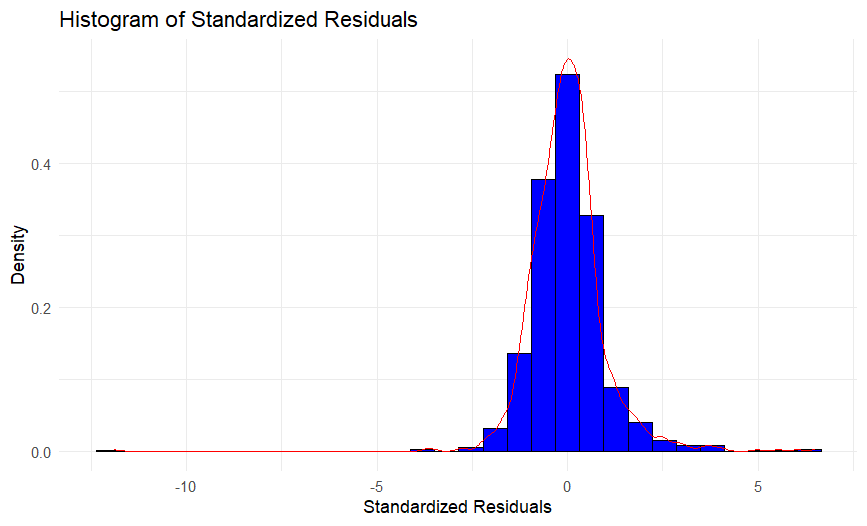
F-statistic

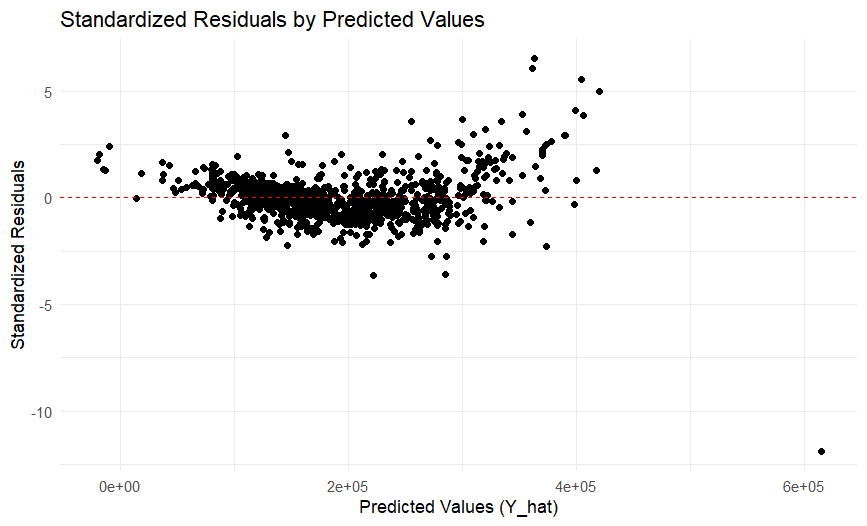
FirstFlrSF = 3420.79

OverallQual = 808.01

Being that all values are well below their p-values, we reject all null hypotheses and accept the alternative hypotheses.

E.



Much like Model 1, we see that the residuals have a nice normal distribution. We

We have a bit of a problem now. The spread of residuals seems to be increasing with the predicted values. This suggests that the variance of the residuals is not constant.

F . The residuals do not appear to be randomly scattered which means the relationship may not be linear. The points that are far away from the red line could be points with high leverage which will affect the slope.

There is also that pesky outlier appearing again.

1. I should retain both variables. Separately they were both better predictors, but together they have an even higher R^2 score. Although this score could be higher just because there are more variables, the score was much higher so I will keep both.

3.

I decided to add Neighborhood for the final variable as that will add the most variety to the different types available.

A.

A screenshot of a computer

Description automatically generated

Model = SalePrice = -67211.48 + 71.05(FirstFlrSF) + 22203.31(OverallQual) + SUM(Neighborhood Coefficients)

The largest different here is since it uses categorical variables, it must add each Neighborhood sub-variable to the model, hence why we must sum them.

B.

The R^2 value increased from 0.79 to 0.8447 which, although not as large as from Model 1 to 2, still increased enough to make a difference. It absolutely helps the model’s explanatory ability.

C. A screenshot of a computer

Description automatically generated

D.

H0: B\_FirstFlrSF = 0

HA: B\_FirstFlrSF != 0

H0: B\_OverallQual = 0

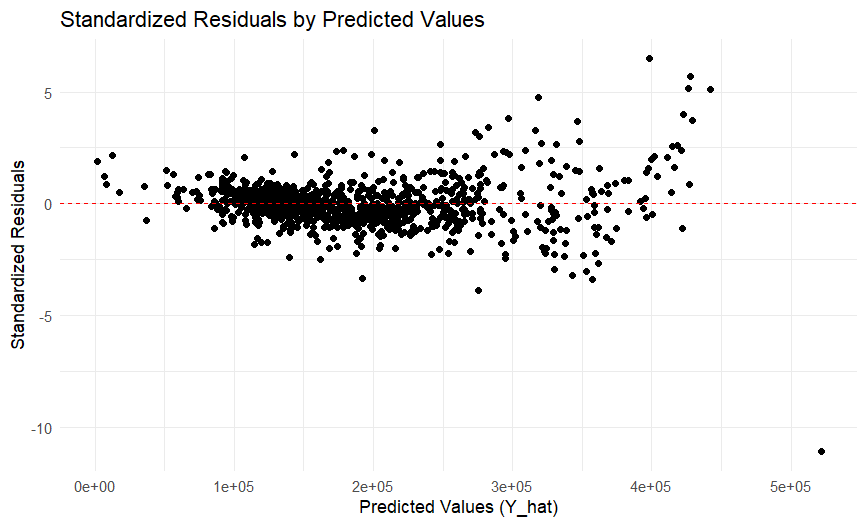
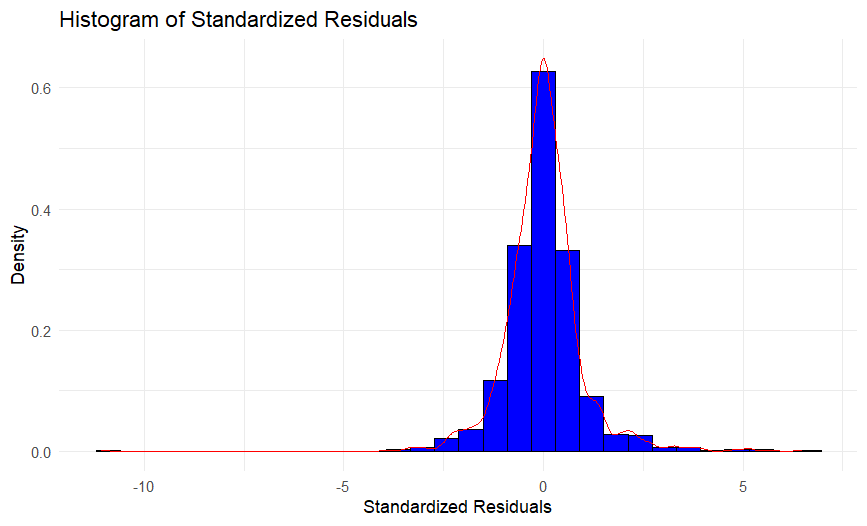
HA: B\_OverallQual != 0

H0: B\_Neighborhood[i] = 0

HA: B\_Neighborhood[i] != 0

All the other variables are like before, but B\_Neighborhood will be different as the null hypothesis is for any given variable inside.

For all three, the p-values are well below the required 0.05 at 2.2e-16 so we must reject the null hypotheses for all three.

E.

Even with the added values, we still have a fairly normal distribution, however the scatterplot does indicate that it may not be totally linear although it seems to have evened out a bit compared to Model 2.

F. My concerns are the same with Model 2.

G. The F-statistic for Neighborhood is only 19.582 compared with FirstFlrSF at 4549.826 and OverallQual at 1074.693. Although its P-value is extremely significant, it does not appear to have affected the model. I’m unsure whether I should keep it in since most of that information is probably already present in the quality of houses. I would assume that higher quality homes are also in nicer neighborhoods.

4. A screenshot of a computer program

Description automatically generated

Comparing both R^2 and adjusted R^2 we see a small improved fit from 0.8447 to 0.8508. The largest difference is the RSE dropped from 33230 to 0.1606. I’m at a loss why it would have changed so much because of the log() function. The F-statistic increased from 273.5 to 286.6 while keeping several Neighborhood significant p-values.

Technically, the model fits much better than the previous, however I’m afraid I may have done something wrong with the log() function.

For the moment, I will say that the new model is superior to the other, but I would have to verify at the end that is the case.

5. A screenshot of a computer

Description automatically generated

Comparing to the log() model 4 with influential points, the new one with less significant points lowers the RSE from 0.1606 to 0.1078. What’s more, the R^2 and residual R^2 increased to 0.913 which means that up until this point, this model has performed best out of the all the previous models. I would assume that this method should be applied every time to help smooth out problems.

6.

I can only assume that this process is automated in a library. I believe researching before, libraries like PyCaret automatically iterate through all possible combinations, applying log(), and finds the best fit. So if I were to iterate through that I would use..

1. Create a list of all the variables
2. Iterate through the list and append the results of the model to another list to compare later.
3. Run through each of those combinations with log(), create a new list of the log results.
4. Any sort of extra model, run every combination of variables.
5. Compare that to removing influential data points.
6. Finally, sort by results and use the best model.

Again, I’m more than certain this is a much more efficient way to iterate through these as that could be very compute intense depending on the size of the data and the dimensions of it.

A computer screen shot of a code

Description automatically generated

I eventually found the leaps library that does just that. Here, I have it setup to find the best model by doing an exhaustive search for the 4 best predictors. However, even with my processor, it was too much for it to handle, meaning that my process would probably not be the best solution.

I was unable to get it to work so I did not have enough time unfortunately to finish the reports, but I would have run the reports similarly to the rest of the assignment.

1. Overall I learned a great deal from this assignment. At first, I found the process to be very repetitive, but now that I’ve gone through the process multiple times, I understand everything much better than when I began.

It’s obvious that taking the original data and applying algorithms can make it a better predictor by smoothing out data, removing influential points that are going to cause the model to veer too far in a direction we don’t want it going. Applying these processes improves the model, or at least has proven that in this example.

I don’t think that these analytical activities cause problems or add difficulties, I think these are great exercises to understand your data better, and to really hammer home which variables have the most impact.

I think there is a reason why statistical hypothesis testing is still around. It absolutely helps with regression and is a cheap and effective way to build quick models on well known and cleaned datasets.

I think the next step is to do what I suggested above and find the absolute best combination of variables to create the most effective models.