Module 1 Assignment: Regression Diagnostics with R

William Scott Breckwoldt

College of Professional Studies: Analytics, Northeastern University

ALY 6015: Intermediate Analytics

Professor Eric Gero

January 16, 2022

Module 1 Assignment: Regression Diagnostics with R

Introduction

The Ames Housing Regression Diagnostics assignment includes the analysis and interpretations of a data set that contains information from the Ames Assessor's Office on individual residential properties sold in Ames, Iowa from 2006 to 2010.

The goal of this assignment is to implement the skills I have learned from Module 1 involved in fitting, interpreting, and evaluating regression models. I will also correct issues with overfitting, linearity, and multi collinearity. Finally, I will select my best model from multiple predictors using automated techniques. The question that needs to be answered for this assignment: what is the best regression model I can create for the Ames Housing Dataset?

In the following analysis, I focus on the steps involved in fitting regression models, interpretating the results of the models, and implementing diagnostic techniques to identify and correct issues with the models, while following the steps listed by Professor Gero. After the analysis, I will provide a conclusion, in which I will share my overall interpretations of the regression models, the diagnostic techniques, and the issues with the models.

Analysis

1. Load R Packages, Set Working Directory, Load Dataset

Figure 1

Loading Libraries, Setting WD, Loading Dataset

```
library(dplyr)
library(ggplot2)
library(corrplot)
library(RColorBrewer)
library(car)
library(pastecs)
library(utils)
library(MASS)
library(leaps)

setwd("C:/Users/Scott/Desktop/ALY6015")

ameshousing = read.csv("DataSets/AmesHousing.csv")

```

a) The R packages necessary for my code are described below.

- dplyr: "dplyr is a grammar of data manipulation, providing a consistent set of verbs that help you solve the most common data manipulation challenges […] dplyr is a part of the tidyverse, an ecosystem of packages designed with common APIs and a shared philosophy" (Francois, Henry, & Wickham).

- ggplot2: "ggplot2 is a system for declaratively creating graphics, based on The Grammar of Graphics. You provide the data, tell ggplot2 how to map variables to aesthetics, what graphical

primitives to use, and it takes care of the details" (Francois, Henry, & Wickham); from the tidyverse ecosystem.

- corrplot: "corrplot provides a visual exploratory tool on correlation matrix that supports automatic variable reordering to help detect hidden patterns among variables" (Simko & Wei, 2021)

- RColorBrewer: "creates nice looking color palettes especially for thematic maps" (Neuwirth, 2015).

- car: car is an acronym for "Companion to Applied Regression" (Fox, 2021) ; car "compliments [Applied Regression] techniques by providing numerous functions that perform tests, creates visualizations, and transform data"(Baiju, Lindsay, & St. John,2020).

- pastecs: "Regularisation, decomposition and analysis of space-time series" (Grosjean, 2018).

- utils: "Utility functions useful when programming and developing R packages" (Bengtsson, 2021).

- MASS: "Functions and datasets to support Venables and Ripley, 'Modern Applied Statistics with S'" (4th edition, 2002)" (Ripley, 2022).

- leaps: "Regression subset selection, including exhaustive search" (Lumley, 2020).

After loading my R packages, I set my working directory to my ALY6015 folder on my desktop and read the AmesHousing.csv file from the "DataSets" folder in my working directory.

2. Exploratory Data Analysis

```
> ncol(ameshousing)
[1] 82
> nrow(ameshousing)
[1] 2930
```

The Ames Housing Data Set contains 2,930 observations and 82 variables. According to the AmesHousingDataDocumentation.text file, of these 82 variables, 23 are nominal, 23 are ordinal, 14 are discrete, 20 are continuous, and there are 2 observation identifiers.

I used glimpse(ameshousing) and view(ameshousing) to get a broad idea of the data. Then I used summary(ameshousing).

> summary(ameshousing)

SalePrice

Min. : 12789

1st Qu.:129500

Median :160000

Mean :180796

3rd Qu.:213500

Max. :755000

This function contained an outcome with descriptive statistics from each numeric column. The function tells us that the minimum SalePrice is 12,789, the first quartile is 129,500, the median is 160,000, the mean is 180,796, the 3rd quartile is 213,500 and the maximum is 755,000. Although not displayed in SalePrice, summary(ameshousing) also allowed me to identify columns with missing (NA) values. For instance:

> summary(ameshousing)

Lot.Frontage

Min.: 21.00

1st Qu.: 58.00

Median: 68.00

Mean: 69.22

3rd Qu.: 80.00

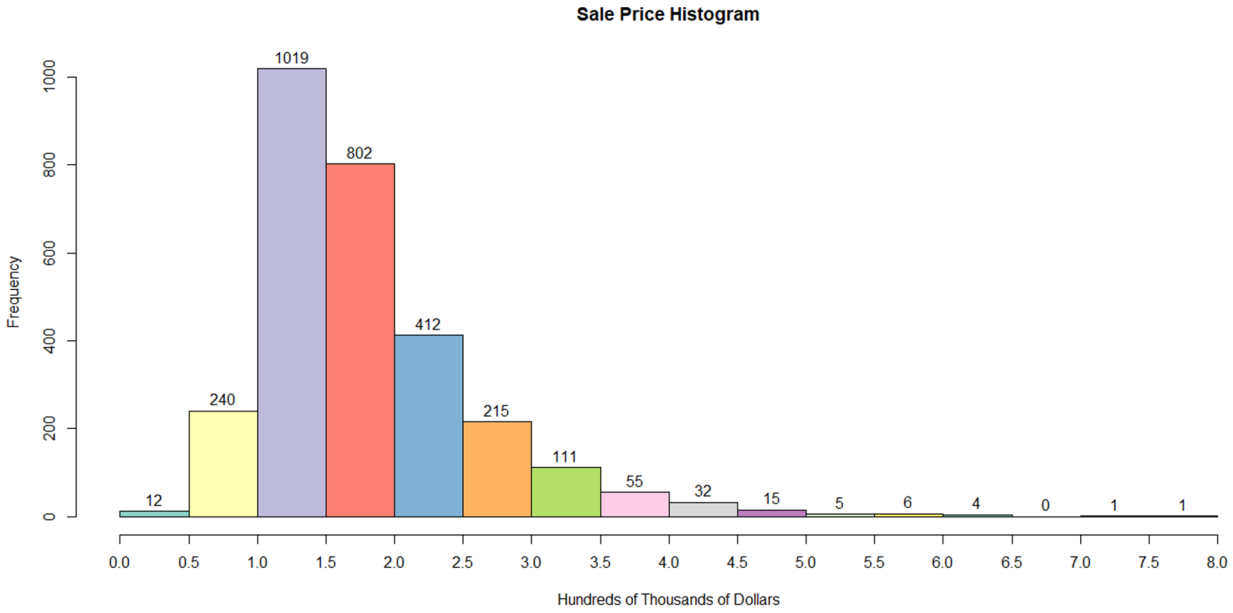
Max. :313.00

NA's :490

As I would be finding correlation among variables and testing fitting regression models for SalePrice, I created some SalePrice plots.

Figure 2

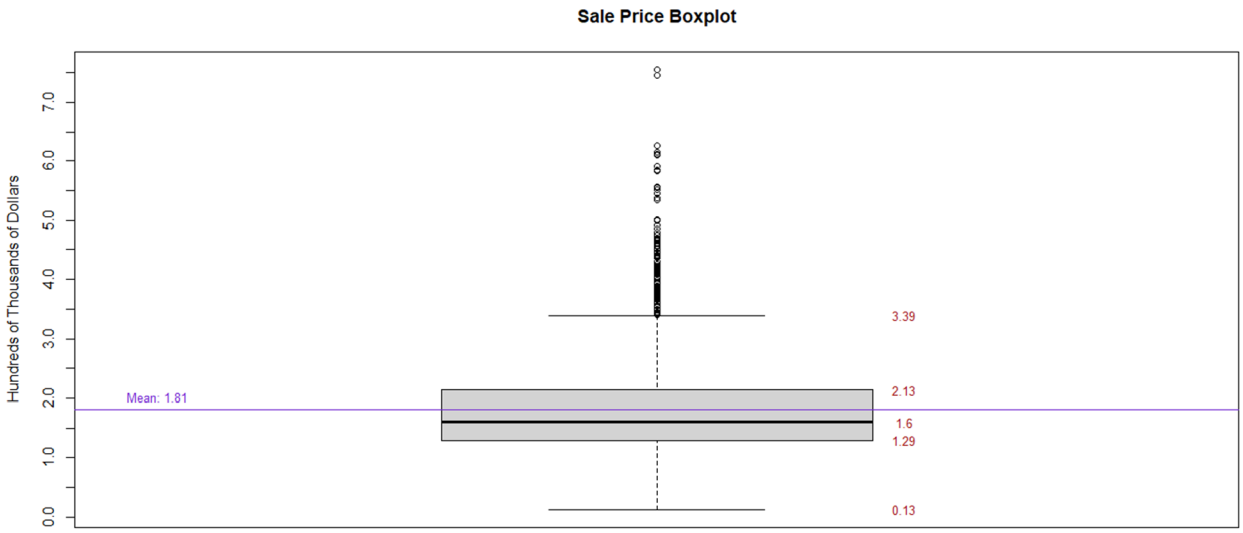
Sale Price Histogram



The Sale Price Histogram provides a visual of the distribution of SalePrice values among the observations. We can see that most houses (1019 + 802 + 412 = 2233) (n=2930) in Ames were sold between $100,000 and $250,000.

Figure 3

Sales Price Boxplot



The Sale Price Boxplot demonstrates the distribution of the data based on the six number summary shown in the summary() function.

I then created a scatter plot matrix with the numeric variables in the dataset.

Figure 4

Creating Numeric Variables Subset (Ameshousing1) & Creating Scatter Plot Matrix Code

```
ameshousing1=select_if(ameshousing, is.numeric)
View(ameshousing1)
ameshousing1 = subset(ameshousing1, select = -c(X..Order,PID))
View(ameshousing1)

scatterplotMatrix(ameshousing1, spread=F, smoother.args = list(lty = 2), main = "Scatter Plot Matrix")
```

Although the subset will be useful later, the scatter plot matrix was illegible. Thus, I created a subset (ameshousing2) with variables that I thought would most effect SalePrice.

Figure 5

Creating Subset (Ameshousing2) & Scatter Plot Matrix Code

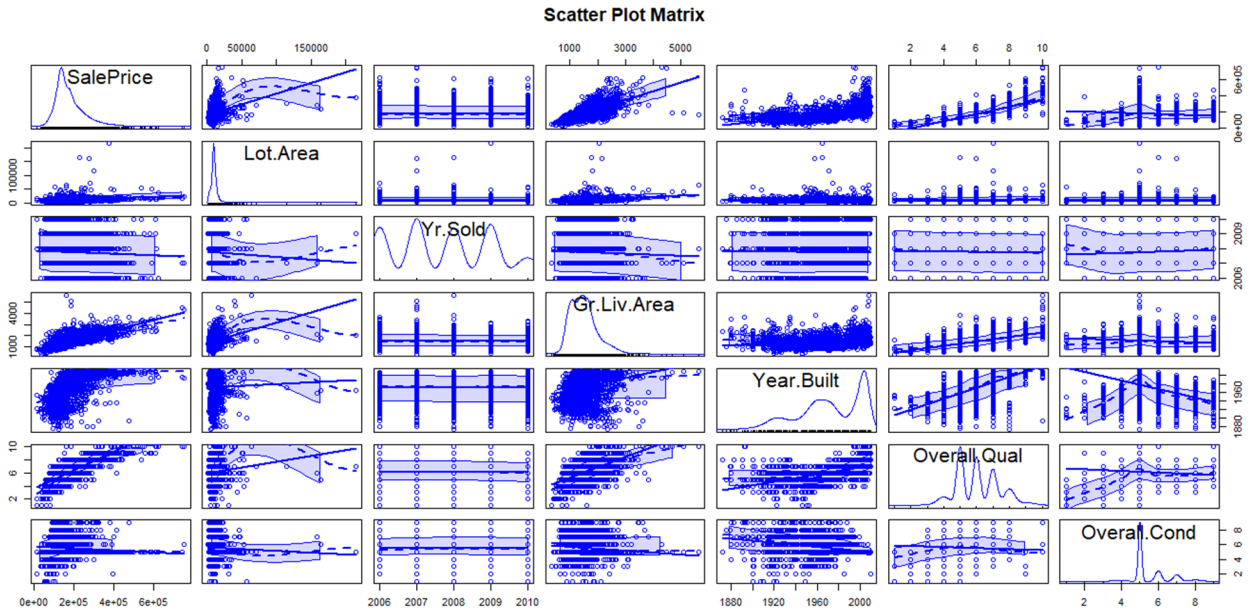
```
ameshousing2 <- ameshousing[c('SalePrice','Lot.Area','Yr.Sold','Gr.Liv.Area','Year.Built','Overall.Qual','Overall.Cond')
glimpse(ameshousing2)

scatterplotMatrix(ameshousing2, spread=F, smoother.args = list(lty = 2), main = "Scatter Plot Matrix")
```

Then created a scatter plot matrix using the subset.

Figure 6

Ameshousing2 Scatter Plot Matrix



It appears that there SalePrice and Gr.Liv.Area, Year.Built, and Overall.Qual have significant positive correlation which will be examined in more detail later on.

3. Prepare Data Set for Modeling (Impute Missing Values)

Two of our numeric values (Lot.Frontage and Mas.Vnr.Area) have a significant number of NA values (493 and 23) as shown by summary(ameshousing) and will be tested for correlation and could potentially be used in our fit formula, thus, I will replace those NA values.

Figure 7

Replacing NA Values with 0

```
####Lot.Frontage NA values to 0
ameshousing$Lot.Frontage[is.na(ameshousing$Lot.Frontage)] <- 0
ameshousing1$Lot.Frontage[is.na(ameshousing1$Lot.Frontage)] <- 0
####Mas.Vnr.Area NA values to 0
ameshousing$Mas.Vnr.Area[is.na(ameshousing$Mas.Vnr.Area)] <- 0
ameshousing1$Mas.Vnr.Area[is.na(ameshousing1$Lot.Frontage)] <- 0
```

I decided to set the missing value in these columns equal to zero. This is because NA values for other variables were rare, suggesting that the NA values for Lot.Frontage and Mas.Vnr.Area were due to immeasurable circumstance, suggesting these observations were impossible to measure or nonexistent.

4. Correlation Matrix

Using ameshousing1, the subset of ameshousing with numeric values, I created a correlation matrix.

Figure 8

Correlation Matrix A

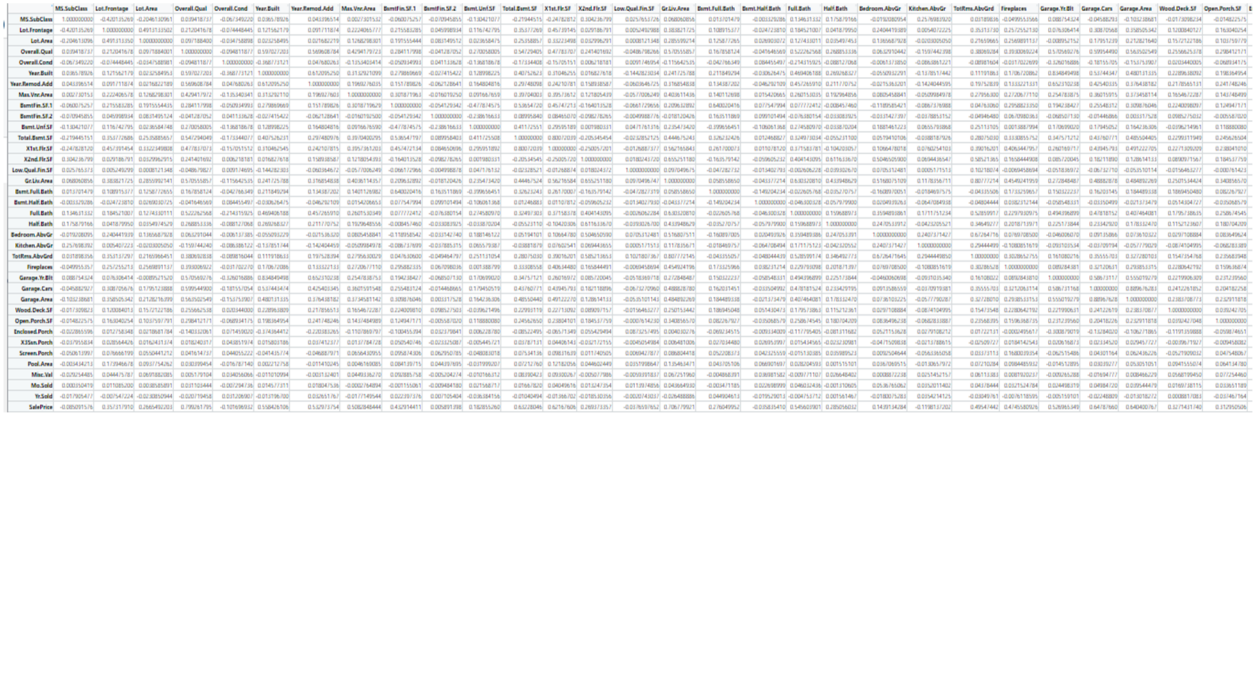
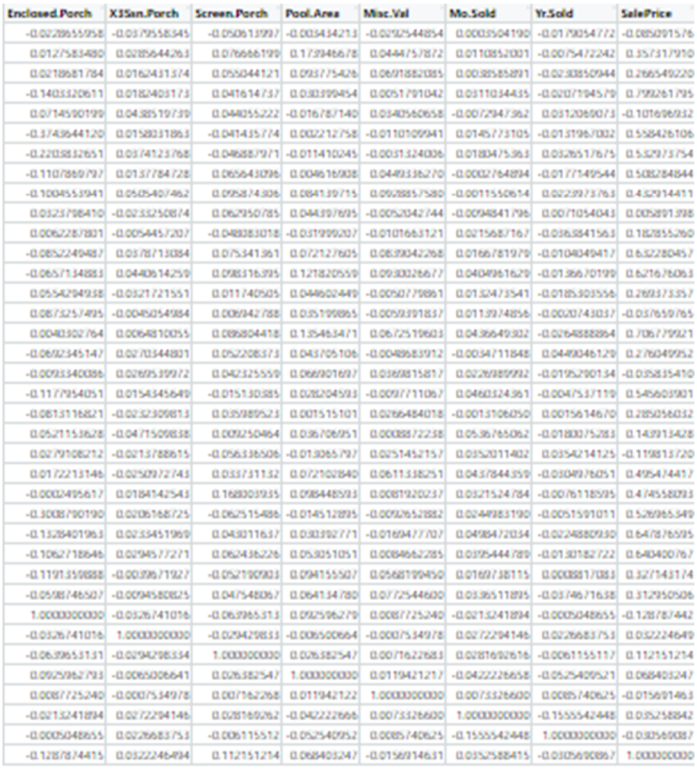


Figure 9

Correlation Matrix B



Enclosed.Porch	X3Ssn.Porch	Screen.Porch	Pool.Area	Misc.Val	Mo.Sold	Yr.Sold	SalePrice
-0.028655958	-0.037558345	-0.050613997	-0.003434213	-0.029254485	0.00035041	-0.017805477	-0.085091576
0.012758348	0.028564426	0.076666199	0.173946678	0.044475787	0.011085203	-0.007547224	0.357317910
0.021868178	0.016243137	0.055044121	0.093775426	0.069188238	0.003858591	-0.023085094	0.266549220
-0.140332061	0.018240317	0.041614737	0.033995454	0.005179104	0.031103443	-0.020719457	0.799261795
0.071459019	0.043851973	0.044055222	-0.016787140	0.034056065	-0.007294736	0.031206907	-0.101698332
-0.374364412	0.015803186	-0.041435774	0.002212758	-0.011010994	0.014577310	-0.013196703	0.558426106
-0.220383265	0.037412376	-0.046887971	-0.011410245	-0.003132403	0.018047536	0.032651767	0.532973754
-0.110786979	0.013778472	0.065643096	0.004616808	0.044933627	-0.000276489	-0.017714954	0.508284844
-0.100455394	0.050540746	0.095874306	0.084139715	0.092885758	-0.001155061	0.022397376	0.432914411
0.032378841	-0.023325087	0.062950785	0.044397695	-0.005204274	-0.009484179	0.007105404	0.005891398
0.006228788	-0.005445720	-0.048383018	-0.031999207	-0.010166312	0.021568716	-0.036384156	0.186855260
-0.085224948	0.037813084	0.075341361	0.072127605	0.083304226	0.016678197	-0.010404941	0.632280457
-0.065713488	0.044061425	0.098316395	0.121820559	0.093002667	0.040496162	-0.013667019	0.621676063
0.055429493	-0.032172155	0.011740505	0.044602449	-0.005077586	0.013247354	-0.018530355	0.269373357
0.087325749	-0.004505498	0.006942788	0.035199865	-0.005939183	0.011397485	-0.002074303	-0.037659765
0.004830276	0.006481005	0.086804418	0.135463471	0.067251960	0.043664930	-0.026488886	0.706775921
-0.069234514	0.027034480	0.052208373	0.043705106	-0.004868391	-0.003471184	0.044904612	0.276049952
-0.009334008	0.026953997	0.042325559	0.066901697	0.036881581	0.022698999	-0.019529015	-0.035835410
-0.117795405	0.015434564	-0.015130385	0.035204593	-0.009771106	0.046032436	-0.004753711	0.545603981
-0.081311682	-0.023230981	0.035989523	0.001515101	0.026648401	-0.001370605	0.001561467	0.285056032
0.052115362	-0.047150983	0.009250464	0.036706951	0.000887223	0.053678506	-0.018807528	0.140913428
0.027910821	-0.021378861	-0.056136506	-0.013065797	0.025145215	0.035201140	0.035421412	-0.119813720
0.017221314	-0.025097274	0.033731132	0.072102840	0.061133825	0.043784435	-0.030497605	0.495474417
-0.000249561	0.018414254	0.168303935	0.058448593	0.008192023	0.032152478	-0.007611859	0.474558093
-0.300879019	0.020616872	-0.062551486	-0.014512895	-0.009265288	0.024498319	-0.005159101	0.526965349
-0.132840196	0.023345196	0.043011637	0.033927771	-0.016947770	0.049847203	-0.022488809	0.647876595
-0.106271864	0.029457727	0.062436226	0.053051051	0.008466228	0.039544478	-0.013018272	0.640400767
-0.119135988	-0.003961927	-0.052190803	0.094155507	0.056819945	0.016973811	0.000881708	0.327143174
-0.058874650	-0.009458082	0.047548067	0.064134780	0.077254460	0.033651189	-0.037467163	0.312950506
1.000000000	-0.032674101	-0.063965313	0.092596279	0.008772524	-0.021324189	-0.000504865	-0.128787442
-0.032674101	1.000000000	-0.029429833	-0.006500664	-0.000753497	0.027229414	0.022668375	0.032224649
-0.063965313	-0.029429833	1.000000000	0.026382547	0.007162268	0.028169261	-0.006115511	0.112151214
0.092596279	-0.006500664	0.026382547	1.000000000	0.011942121	-0.042222266	-0.052540952	0.068403247
0.008772524	-0.000753497	0.007162268	0.011942122	1.000000000	0.007332660	0.008574062	-0.015691463
-0.021324189	0.027229414	0.028169262	-0.042222666	0.007332660	1.000000000	-0.155555424	0.035258842
-0.000504865	0.022668375	-0.006115512	-0.052540952	0.008574062	-0.155555424	1.000000000	-0.030569087
-0.128787441	0.032224649	0.112151214	0.068403247	-0.015691463	0.035258841	-0.030569086	1.000000000

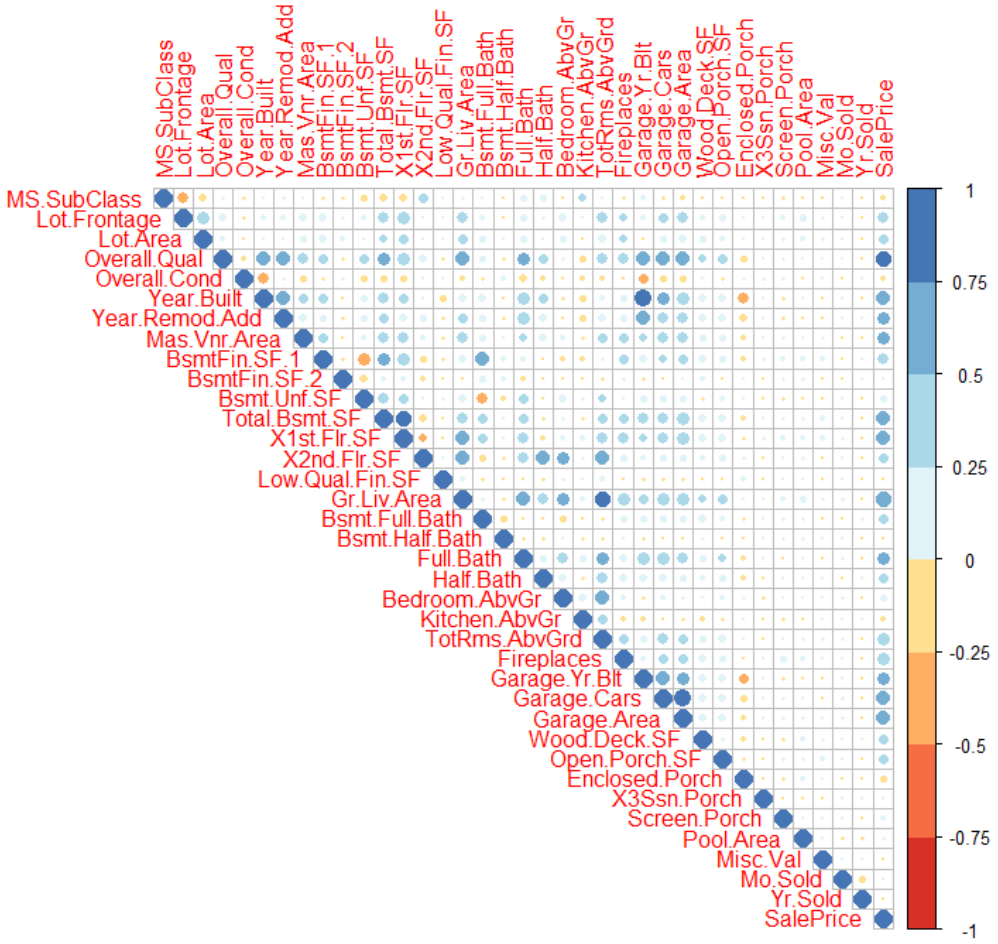
The correlation matrix is quite large, so I split it into two figures. The column/row of most importance is SalePrice. I looked for variables with high correlations with SalePrice. I would use this as a reference, later to choose my initial variables for my fit function.

5. Correlation Matrix Plot

> corrplot(cors, type = 'upper',col=brewer.pal(n=8,name="RdYlBu"))

Figure 10

Correlation Matrix Plot



Again, I am paying attention to high correlations between variables and SalePrice. Overall.Qual seems to have the highest correlation, which is confirmed when looking at the Correlation Matrix. Some other variables with significantly high correlation to SalePrice include Year.Built, Year.Remod.Add, Mas.Vnr.Area, Total.Bsmt.SF, X1st.Flr.Sf, Gr.Liv.Area, Garage.Cars, and Garage.Area.

6. SalePrice Scatter Plots

a. Scatter Plot for Continuous Variable with the Highest Correlation

The continuous variable with the highest correlation with SalePrice is Gr.Liv.Area with 0.706779921.

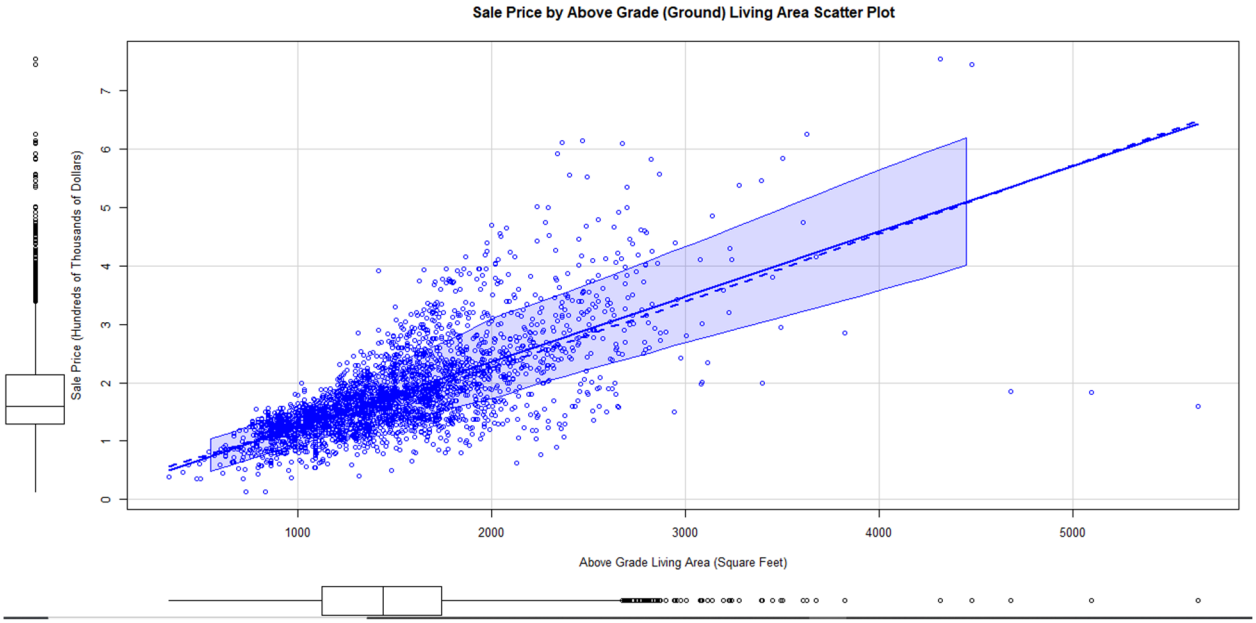
Figure 11

Sale Price by Above Grade (Ground) Living Area Scatter Plot Code

```
104  scatterplot(SalePrice/100000 ~ Gr.Liv.Area,
105              data = ameshousing1,
106              main = 'Sale Price by Above Grade (Ground) Living Area Scatter Plot',
107              ylab='Sale Price (Hundreds of Thousands of Dollars)',
108              xlab='Above Grade Living Area (Square Feet)',
109              yaxp=c(0,8,8))
```

Figure 12

Sale Price by Above Grade (Ground) Living Area Scatter Plot



The positive correlation between the variables is displayed with the trend line, lowess line, that resemble the assumed regression line, and the 95% envelope. Plenty of points fall outside the 95% envelope and are far from the trend line which is not a surprise because there are thousands of observations in the dataset.

b. Scatter Plot for Variable with the Lowest Correlation

The variable with the lowest correlation is BsmtFin.SF.2 with 0.005891398.

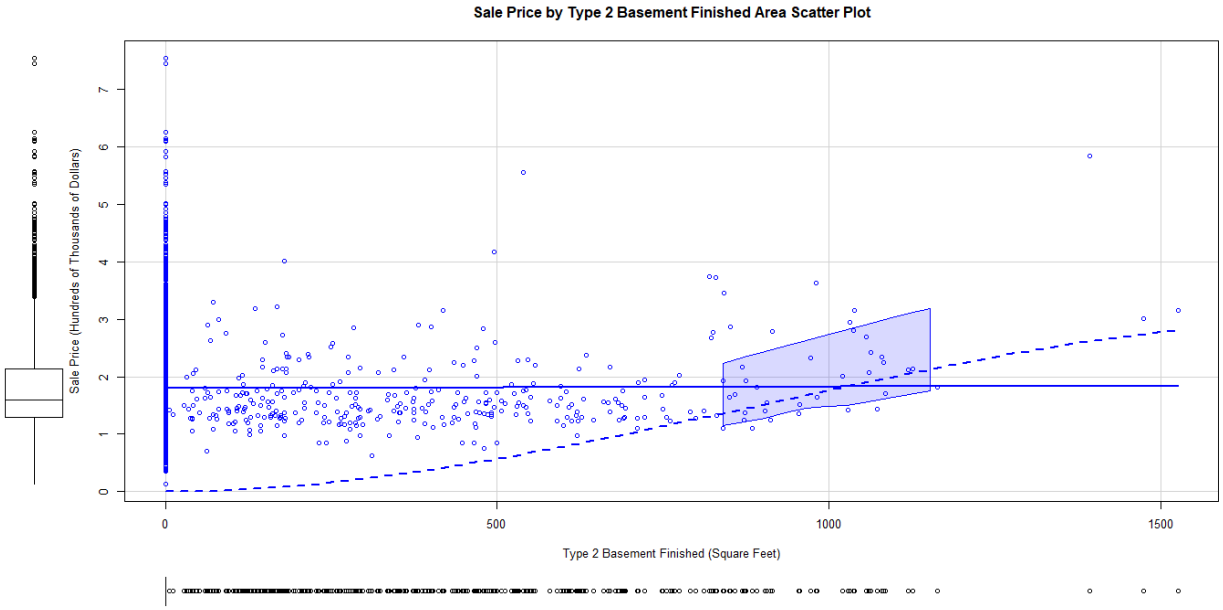
Figure 13

Scatter Plot of Sale Price and Basement Type 2 Finished Rating Code

```
113  scatterplot(SalePrice/100000 ~ BsmtFin.SF.2,
114              data=ameshousing1,
115              main='Sale Price by Year Sold Scatter Plot',
116              ylab='Sale Price (Hundreds of Thousands of Dollars)',
117              xlab='Type 2 Basement Finished Sq. Ft.',
118              yaxp=c(0,8,8))
```

Figure 14

Scatter Plot of Sale Price and Basement Type 2 Finished Rating



There appears to be a random distribution of observations and the trend line is linear. This visual suggests that there is no correlation between BsmtFin.SF.2 and SalePrice because of the lowess line's linearity.

c. Scatter Plot for Variable with the Correlation Closest to 0.5

The variable with a correlation closest to 0.5 is TotRms.AbvGrd with 0.495474417.

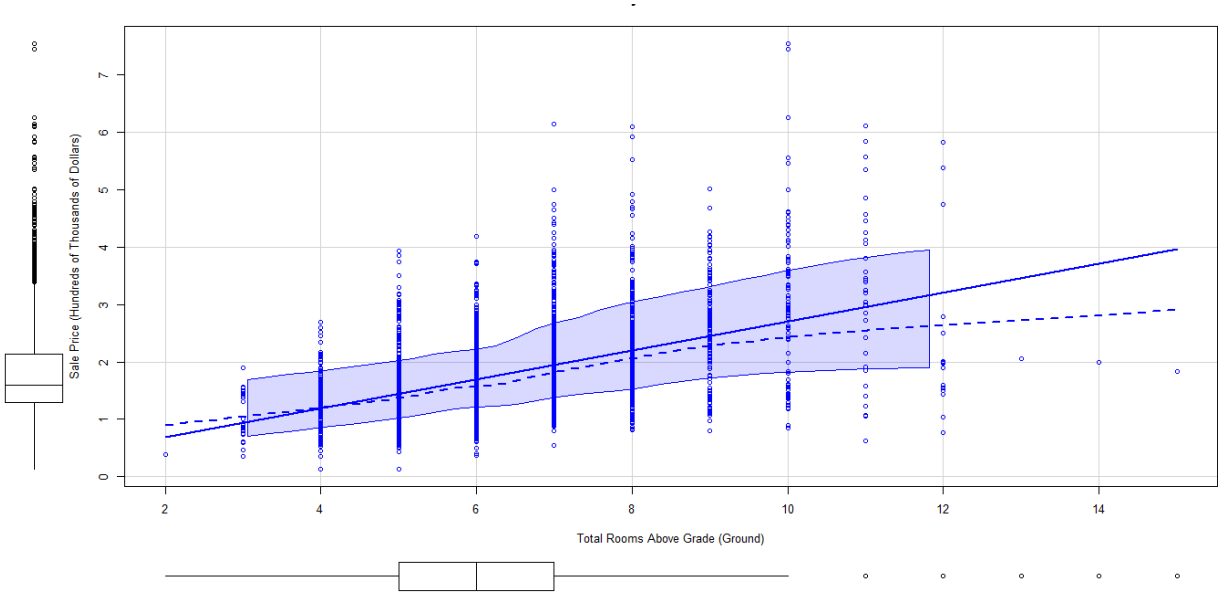
Figure 15

Scatter Plot of Sale Price by Total Rooms Above Grade (Ground) Code

```
122 scatterplot(SalePrice/100000 ~TotRms.AbvGrd,
123               data=ameshousing1,
124               main='Sale Price by Year Built Scatter Plot',
125               ylab='Sale Price (Hundreds of Thousands of Dollars)',
126               xlab='Total Rooms Above Grade (Ground)',
127               yaxp=c(0,8,8))
```

Figure 16

Scatter Plot of Sale Price by Total Rooms Above Grade (Ground) Code



The lowess trend line is diagonal, however, many observations lie outside of the 95% envelope. From looking at the plot, there appears to be moderate correlation between SalePrice and TotRms.AbvGrd.

7. Fit a Regression Model

I used the 4 continuous variables with the highest correlation values associated as seen with corr() using SalePrice to fit my regression model.

fit <-lm(formula = SalePrice ~ Gr.Liv.Area + Garage.Area + Total.Bsmt.SF + X1st.Flr.SF, data=ameshousing1)

8. Regression Model in Equation Form

Figure 17

summary(fit) Outcome

```
Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)   -29050.259   2947.298  -9.857   <2e-16 ***
Gr.Liv.Area      69.289      2.095  33.070   <2e-16 ***
Garage.Area     105.298      4.750  22.167   <2e-16 ***
Total.Bsmt.SF    55.957      3.227  17.342   <2e-16 ***
X1st.Flr.SF      -2.302      3.869  -0.595    0.552
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 45260 on 2923 degrees of freedom
  (2 observations deleted due to missingness)
Multiple R-squared:  0.6795,    Adjusted R-squared:  0.6791
F-statistic:  1549 on 4 and 2923 DF,  p-value: < 2.2e-16
```

This tells me that my regression model in equation for is Y= -29,050.259 + 69.289*X1 + 105.298*X2 + 55.957*X3 – 2.302*X4, where X1 = Gr.Liv.Area, X2 = Garage.Area, X3 = Total.Bsmt.SF, and X4 = X1st.Flr.SF.

The outcome of summary(fit) also tells me that X1st.Flr.SF has a Pr(>|t|) of 0.552, which is unusually high. This leads me to run stepAIC(fit) to make sure I have the most accurate regression model.

Figure 18

stepaic(fit, direction = "both") Outcome

```
> stepAIC(fit,direction = "both")
Start:  AIC=62782.2
SalePrice ~ Gr.Liv.Area + Garage.Area + Total.Bsmt.SF + X1st.Flr.SF

               Df  Sum of Sq        RSS   AIC
- X1st.Flr.SF    1 7.2514e+08 5.9881e+12 62781
<none>                       5.9873e+12 62782
- Total.Bsmt.SF  1 6.1607e+11 6.6034e+12 63067
- Garage.Area    1 1.0065e+12 6.9939e+12 63235
- Gr.Liv.Area    1 2.2402e+12 8.2275e+12 63711

Step:  AIC=62780.56
SalePrice ~ Gr.Liv.Area + Garage.Area + Total.Bsmt.SF

               Df  Sum of Sq        RSS   AIC
<none>                       5.9881e+12 62781
+ X1st.Flr.SF    1 7.2514e+08 5.9873e+12 62782
- Garage.Area    1 1.0080e+12 6.9961e+12 63234
- Total.Bsmt.SF  1 1.1973e+12 7.1853e+12 63312
- Gr.Liv.Area    1 2.5098e+12 8.4979e+12 63803

Call:
lm(formula = SalePrice ~ Gr.Liv.Area + Garage.Area + Total.Bsmt.SF,
    data = ameshousing1)

Coefficients:
  (Intercept)    Gr.Liv.Area    Garage.Area  Total.Bsmt.SF
    -29536.07          68.86         105.09          54.59
```

The AIC is lower, suggesting the model is more accurate, when:

fit <-lm(formula = SalePrice ~ Gr.Liv.Area + Garage.Area + Total.Bsmt.SF, data=ameshousing1).

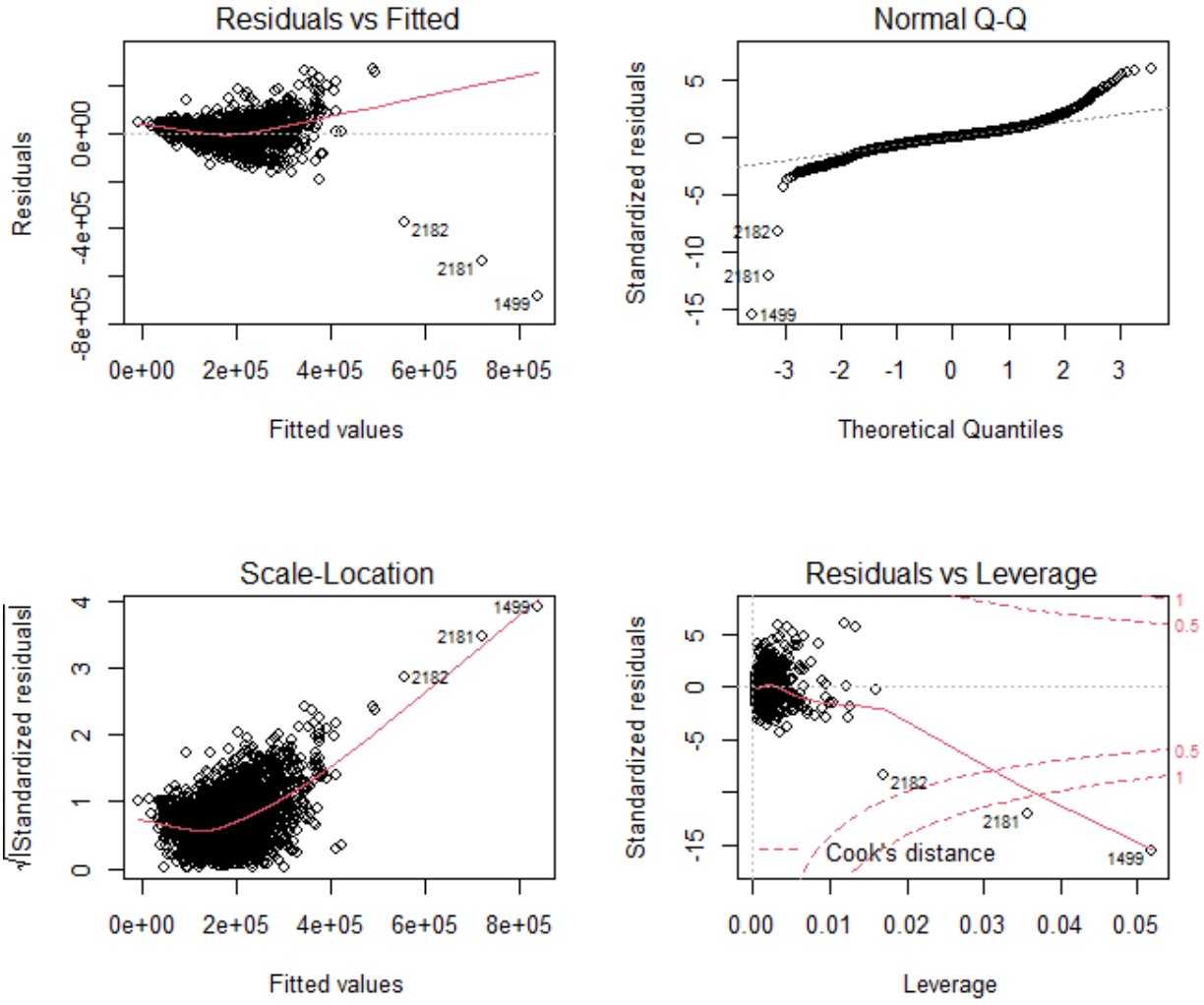
Thus, I use this formula for the regression model. I run summary(fit) again, as well as AIC(fit) and BIC(fit) to ensure they are both lower for this formula, which is the case.

Y= -29,536.070 + 68.860*X1 + 105.091*X2 + 54.586*X3 is our regression model in equation form. X1 = Gr.Liv.Area, X2 = Garage.Area, and X3 = Total.Bsmt.SF.

9. Plot the Regression Model

Figure 19

plot(fit) Outcome



There is random dispersion in the Residual vs Fitted plot as I hoped. However, the assumption of linearity appears to not be true; our fitted value and residuals increase faster beginning at a fitted value of 2e+5 causing the lowess line to get further from the suggested regression line.

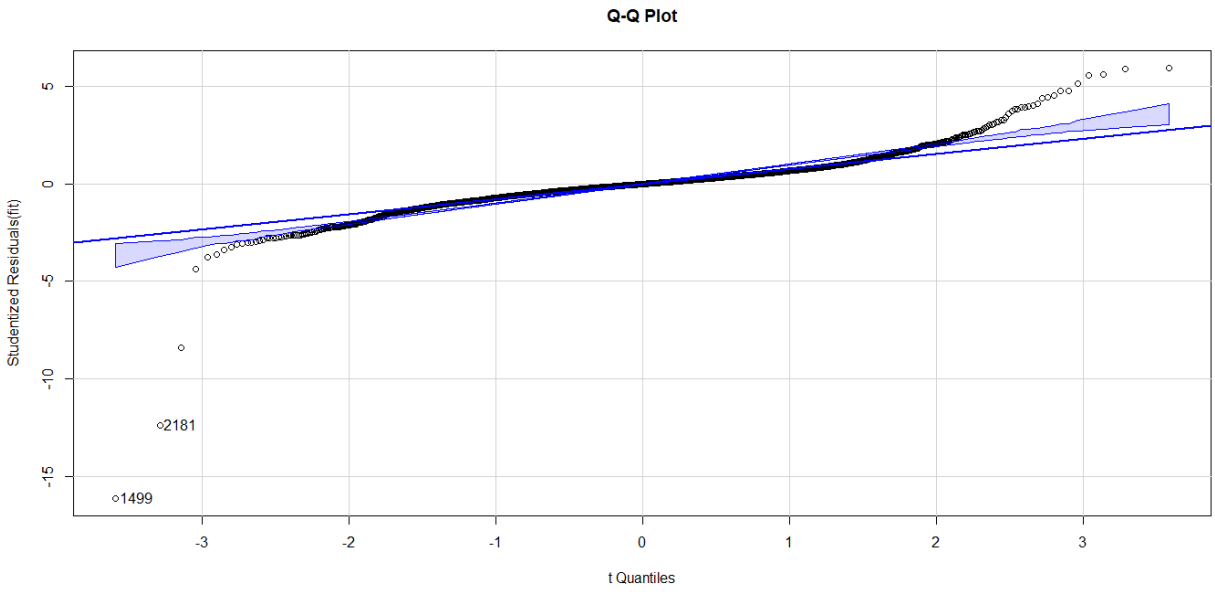
We will look more closely at the other Q-Q plot shortly. One note, before we do so is that 1499,2181, and 2182 all appear to be outliers considering their unusual dispersion, which is constant in all four plots.

The scale location plot shows somewhat constant variance or homosdasticity. I want to see a random, not distinctive, pattern. Although somewhat random distribution, the slope of the lowess line increases exponentially, starting when the fitted value is roughly 3e+05. This could be attributed to our outliers, which seem to appear after 4e+05.

In the Residuals vs Leverage plot, we can identify unusual observations. Again, we have observations 1499, 2181, and 2182 as standouts.

Figure 20

Fit Q-Q Plot



The Q-Q Plot evaluates normality. Ideally, we want these plots to fall on a diagonal line. Although the points somewhat resemble a diagonal line, it appears most points are outside our 95% confidence envelope especially, towards the beginning and end of the plot (<-2 quartile and >2 quartile). This causes some concern; however, most points are focused on the center and there are thousands of observations in the dataset. Regardless, there are many outliers that are far from the diagonal and 95% envelope so I will keep this in mind when we are examining outliers.

I would like to point out that none of these plots are great. As an analytics student, not a housing professional, I cannot accurately attribute these flaws in the plots. A subject matter expert may provide more insight.

10. Check Model for Multicollinearity

I check for multicollinearity using the variable inflation factors:

```
> vif(fit)
Gr.Liv.Area   Garage.Area Total.Bsmt.SF
   1.413159      1.483145      1.414582
```

Multicollinearity is relatively low for our predictors. In general, anything less than 5 is not of concern and anything 10 or greater would be cause for concern. If there were high multicollinearity, I would remove the variable from the fit function.

11. Check Model for Outliers

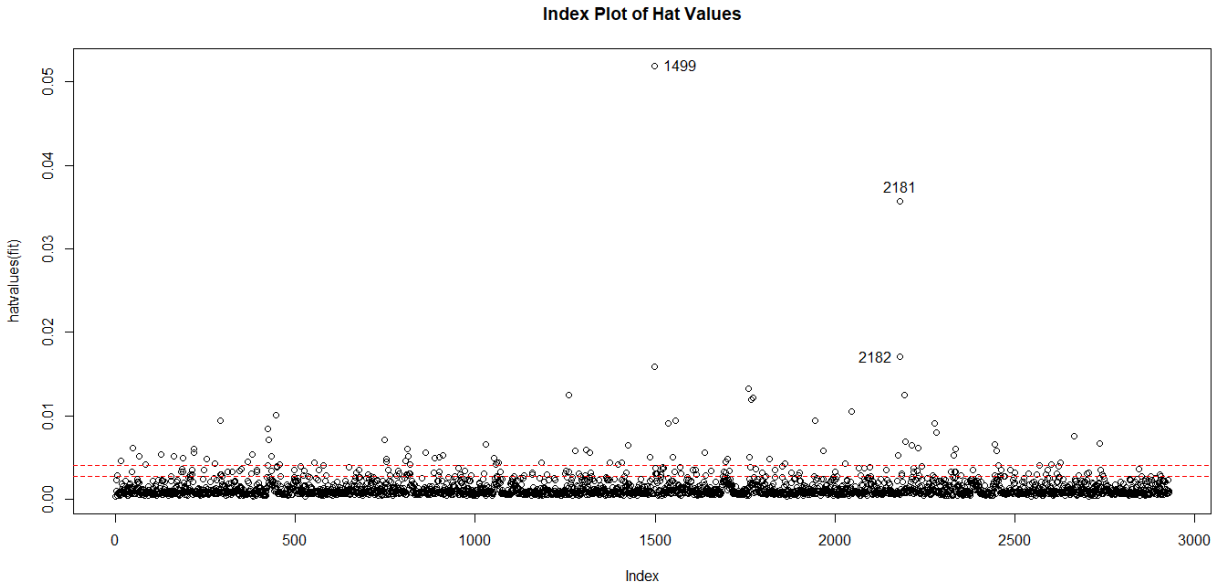
```
> outlierTest(model=fit)
      rstudent unadjusted p-value Bonferroni p
1499 -16.137902        3.4506e-56   1.0103e-52
2181 -12.391668        2.0879e-34   6.1134e-31
2182  -8.386564        7.6814e-17   2.2491e-13
1768   5.959263        2.8368e-09   8.3062e-06
45     5.900465        4.0410e-09   1.1832e-05
1064   5.635849        1.9079e-08   5.5865e-05
1761   5.592287        2.4479e-08   7.1674e-05
433    5.157161        2.6745e-07   7.8310e-04
434    4.806008        1.6175e-06   4.7360e-03
2446   4.775965        1.8766e-06   5.4946e-03
```

Seeing this many outliers is a surprise because only 1499, 2181, 2182 were identified as potential outliers in our plots.

Furthermore, I plot for high-leverage observations, when creating hat.plot(fit).

Figure 21

Fit Hat Plot

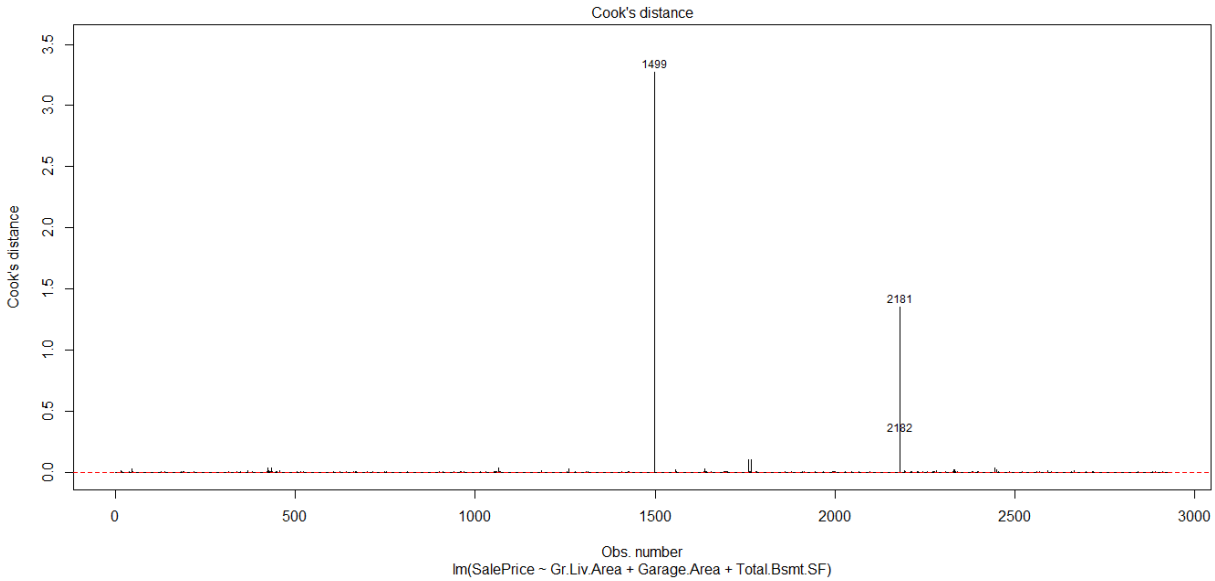


The hat plot identifies 1499, 2181, and 2182 as the highest leverage observations, but there also seems to be plenty more (at least 20) high-leverage observations.

To test for influential observations, I create a Cook's Distance plot.

Figure 22

Fit Cook's Distance Plot



The plot tells us that 1499, 2181, and 2182 are the most influential observations.

12. Correct Issues with the Model

It is apparent that observations 1499, 2181, and 2182 are outliers so I will create a subset without these observations.

> ameshousing3 = ameshousing1[-c(1499, 2181, 2182),]

Then, I will run a new fit formula using this subset.

>fit1 <-lm(formula = SalePrice ~ Gr.Liv.Area + Garage.Area + Total.Bsmt.SF, data=ameshousing3)

> summary(fit1)

This gives us a new regression model equation: Y= -46,101.585 + 75.089*X1 + 96.149*X2 + 66.049*X3. Again, X1 = Gr.Liv.Area, X2 = Garage.Area, and X3 = Total.Bsmt.SF.

I will test for the new AIC and BIC:

>AIC(fit1)

70,492.39

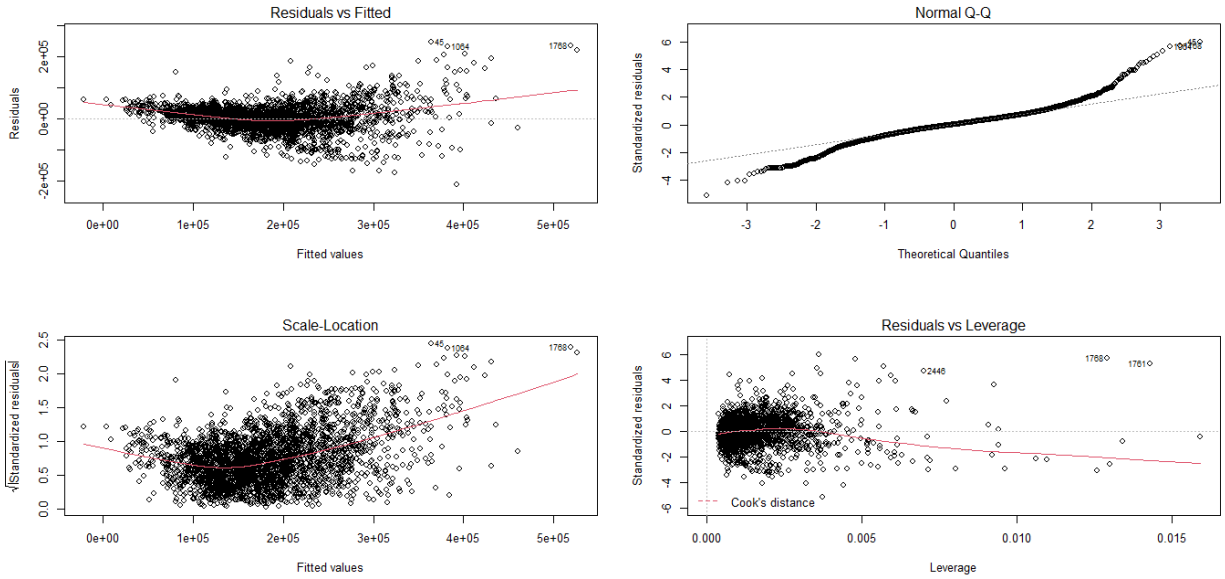
>BIC(fit1)

70,522.3

These are both lower than the AIC and BIC from the old formula, which were AIC = 71091.86 and BIC = 71,121.77. This suggests a more accurate model. Let us look at the new plots.

Figure 23

Plot(fit1) Outcome



The plots have improved significantly, however, we still have some outliers identified in each plot.

Next, I will test again for multicollinearity.

> vif(fit1)

Gr.Liv.Area Garage.Area Total.Bsmt.SF

1.367761 1.477458 1.370315

All the variable inflation factors have decreased, once again suggesting a more accurate model.

I will check again for outliers, hoping to find some of the unusual observations from the plots.

>outlierTest(model=fit1)

rstudent unadjusted p-value Bonferroni p

45 6.031656 1.8271e-09 5.3441e-06

1768 5.765281 9.0047e-09 2.6339e-05

1064 5.661006 1.6510e-08 4.8292e-05

1761 5.348393 9.5594e-08 2.7961e-04

2593 -5.168926 2.5131e-07 7.3509e-04

433 5.073275 4.1550e-07 1.2153e-03

434 4.974328 6.9265e-07 2.0260e-03

2446 4.716709 2.5095e-06 7.3402e-03

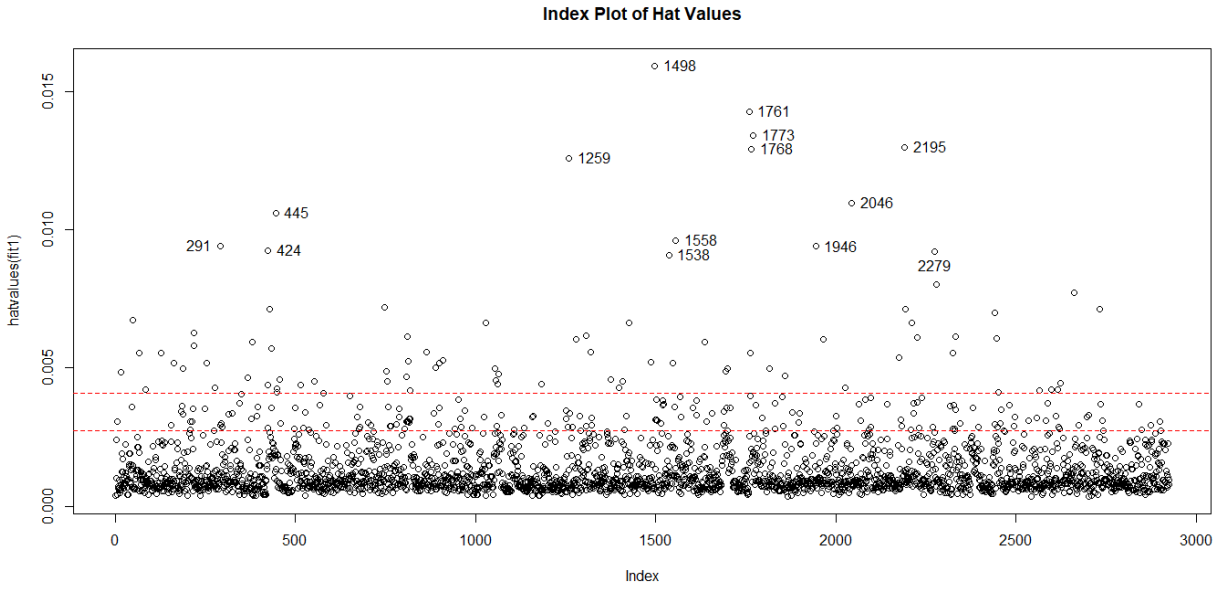
2333 4.542936 5.7726e-06 1.6885e-02

2335 4.507582 6.8150e-06 1.9934e-02

We still have some outliers with high rstudent values, although, fortunately they are not as high as the rstudent values of observations 499, 2181, and 2182. I will run a hat.plot.

Figure 24

Fit1 Hat Plot

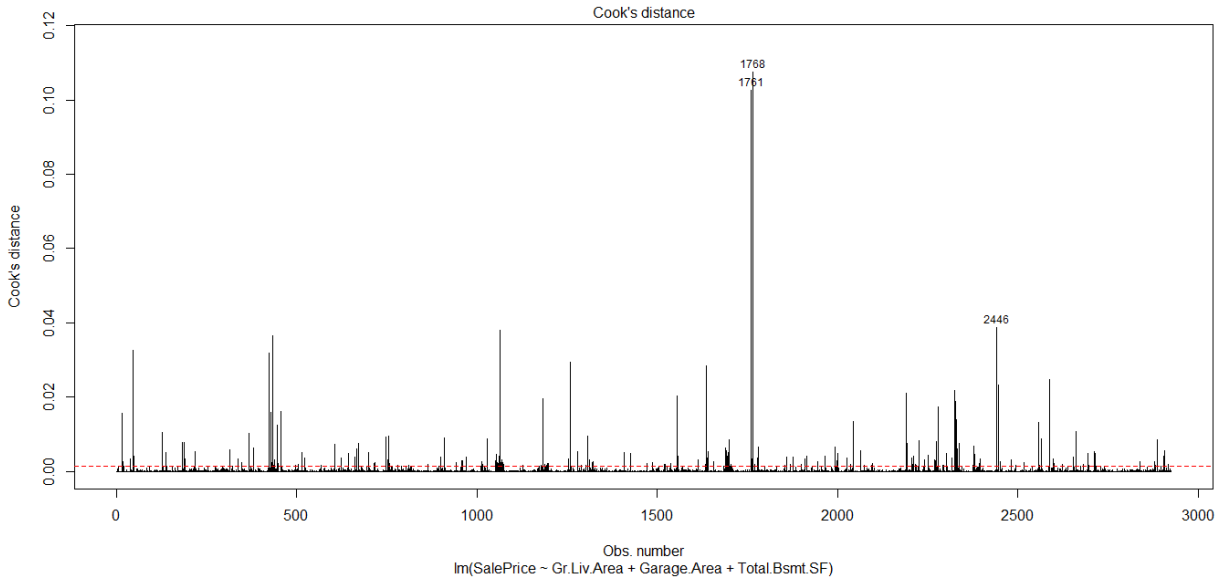


The hat plot has identified high leverage observations. I selected observations that seemed far from the acceptable dotted line, which include 1498, 1761, 1773, 2195, 1768, 1259, 2046, 445, 1558, 291, 1946, 2279, 291, 424, and 1538. Some of these observations were mentioned in the outlierTest() outcome, others were not.

I will run a Cook's D plot again to find influential observations.

Figure 25

Fit1 Cook's Distance Plot



As we can see, 1768 , 1761, and 2446 are the most influential observations. However, it appears there are many other observations, which I am guessing were mentioned in the outcome of outlierTest(fit1) and the hat plot, that also seem to be very influential.

I correct for the unusual observations identified in the outlierTest(fit1) outcome, the hat plot, and Cook's D plot to see if I can get an even more accurate model. This seems appropriate as there were almost 3,000 observations in our original dataset and there are bound to be plenty of outliers.

>ameshousing4 = ameshousing3[-c(957, 1266, 2767, 1761, 45, 1064, 2593, 433, 434, 1641, 2333, 1768, 2446, 1498, 1761, 1773, 2195, 1768, 1259, 2046, 445, 1558, 291, 1946, 2279, 291, 424, 1538),]

I am going to add some variables to the fit function and run stepAIC() once again to see if I can get a more accurate model with more variables.

>fit3 <- lm(formula = SalePrice ~ Gr.Liv.Area + Garage.Area + Total.Bsmt.SF + Mas.Vnr.Area + Lot.Frontage + Lot.Area , data=ameshousing4)

Figure 26

stepaic(fit3) Outcome

```
> stepAIC(fit3, direction = "both")
Start:  AIC=50652.19
SalePrice ~ Gr.Liv.Area + Garage.Area + Total.Bsmt.SF + Mas.Vnr.Area +
    Lot.Frontage + Lot.Area

                Df  Sum of Sq        RSS   AIC
<none>                        3.6760e+12 50652
- Lot.Frontage   1 1.5647e+10 3.6917e+12 50660
- Lot.Area       1 5.7433e+10 3.7334e+12 50687
- Mas.Vnr.Area   1 2.7551e+11 3.9515e+12 50823
- Garage.Area    1 6.1615e+11 4.2922e+12 51021
- Total.Bsmt.SF  1 9.8702e+11 4.6630e+12 51220
- Gr.Liv.Area    1 1.6473e+12 5.3233e+12 51537

Call:
lm(formula = SalePrice ~ Gr.Liv.Area + Garage.Area + Total.Bsmt.SF +
    Mas.Vnr.Area + Lot.Frontage + Lot.Area, data = ameshousing4)

Coefficients:
  (Intercept)    Gr.Liv.Area    Garage.Area  Total.Bsmt.SF   Mas.Vnr.Area   Lot.Frontage       Lot.Area
   -35484.997         67.490         91.898         59.355         69.930       -144.453          1.322
```

Surprisingly, after when using ameshousing4 and adding variable back into the fit formula, our model appears to be more accurate according to stepAIC. Our most accurate regression model in equation form is Y = -35484.997 + 67.490*X1 + 91.898*X2 + 59.355*X3 + 69.930*X4 – 144.453*X5 + 1.322*X6. X1 = Gr.Liv.Area, X2 = Garage.Area, X3 = Total.Bsmt.Sf, X4 = Mas.Vnr.Area, X5 = Lot.Frontage, and X6 = Lot.Area.

I run AIC and BIC to confirm the improvement in accuracy.

> AIC(fit3)

57448.07

> BIC(fit3)

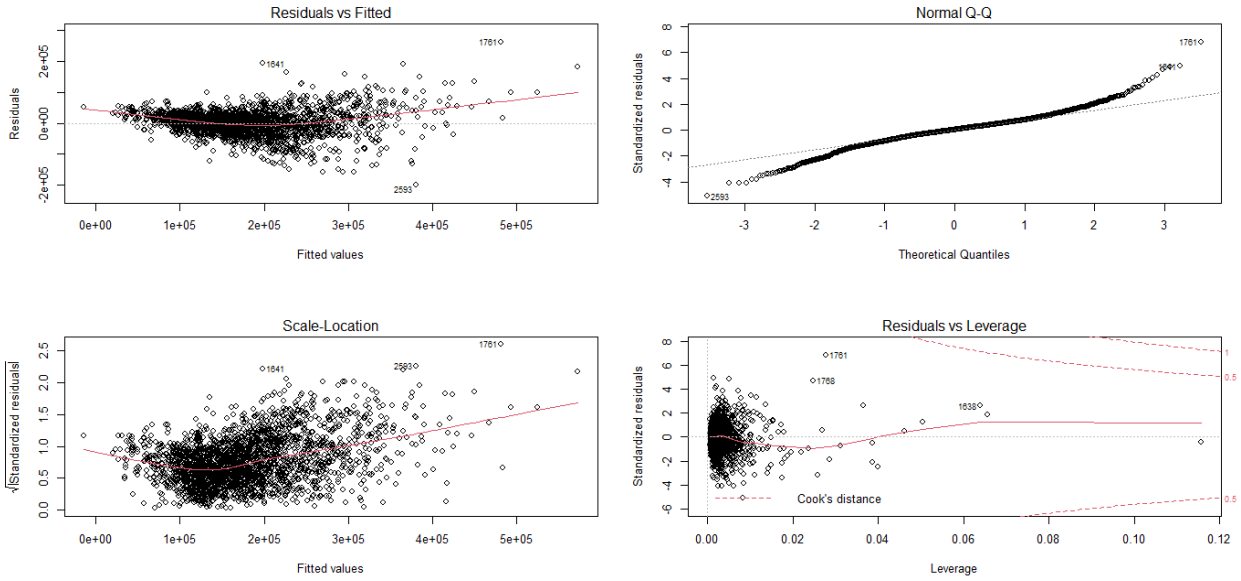
57494.32

These are in fact the lowest values for AIC and BIC.

I will run plot(fit3) to see the improvement in plots.

Figure 27

plot(fit3) Outcome



The plots do seem to be more accurate, however, there are still unusual observations identified in each plot.

I test for multicollinearity.

>vif(fit3)

Gr.Liv.Area	Garage.Area	Total.Bsmt.SF	Mas.Vnr.Area	Lot.Frontage	Lot.Area
1.553738	1.577423	1.506675	1.307273	1.564191	1.561766

All variable inflation factors are low. Time to identify our outliers.

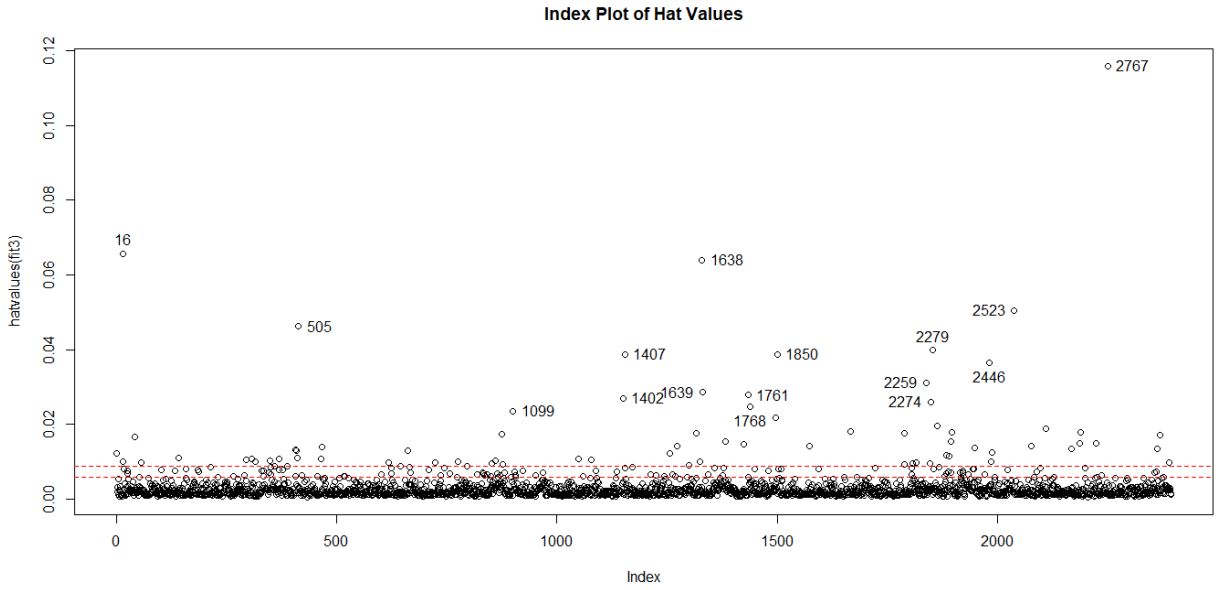
> outlierTest(model=fit3)

	rstudent	unadjusted p-value	Bonferroni p
1761	6.886904	7.2659e-12	1.7394e-08
2593	-5.141593	2.9455e-07	7.0516e-04
1641	4.955002	7.7401e-07	1.8530e-03
2333	4.902173	1.0115e-06	2.4216e-03
1768	4.727490	2.4055e-06	5.7587e-03

There are still some somewhat high rstudent scores, however, there are only 5 outliers identified, which is the best outcome I have had so far. I run the hat plot for fit3.

Figure 28

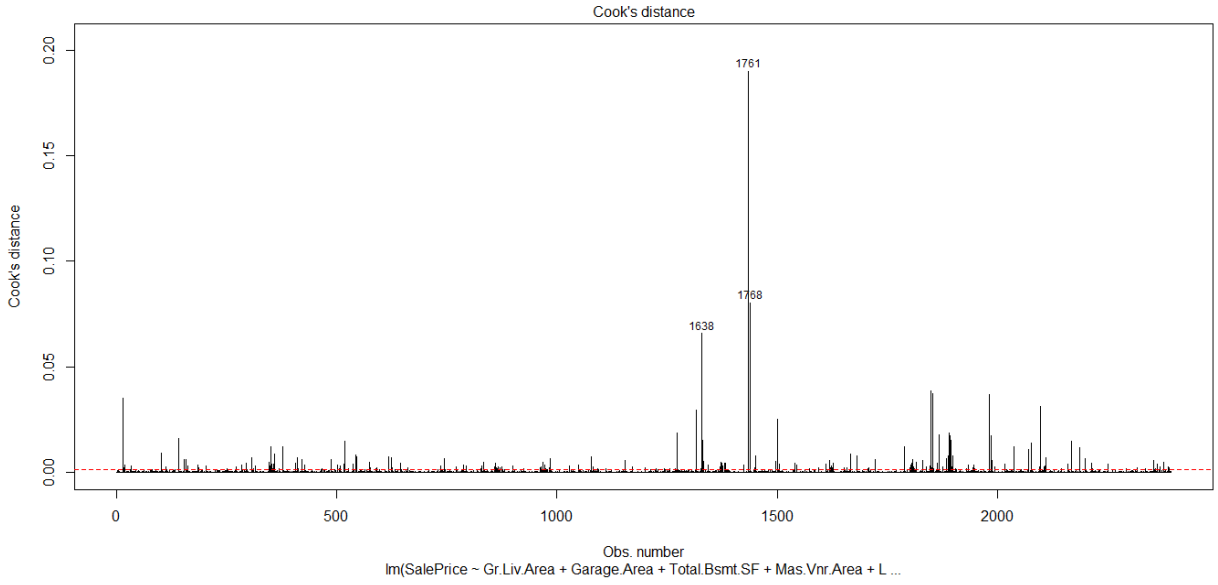
Fit3 Hat Plot



I select the high-leverage points of some concern. Next, I create the Cook's D plot.

Figure 29

Fit3 Cook's Distance Plot



The plot identifies 1761, 1768, and 1638 as influential observations.

The assignment requires 2 regression models, I have now created 3 so I will end this step here. It is worth noting that I could continue to eliminate unusual observations and outliers, which would be justified because of the size of the set and potentially add more variables to the fit formula, after checking the updated correlation matrix and using stepAIC(). However, the assignment does not require this.

13. All Subsets Regression Method to Identify "Best" Model

```
>leaps <- regsubsets(SalePrice ~ Gr.Liv.Area + Garage.Area + Total.Bsmt.SF + Mas.Vnr.Area +
Lot.Frontage + Lot.Area , data=ameshousing4, nbest = 6)

>summary(leaps)

         Gr.Liv.Area Garage.Area Total.Bsmt.SF Mas.Vnr.Area Lot.Frontage Lot.Area
1  ( 1 ) "*"         " "         " "           " "          " "          " "
1  ( 2 ) " "         " "         "*"           " "          " "          " "
1  ( 3 ) " "         "*"         " "           " "          " "          " "
1  ( 4 ) " "         " "         " "           "*"          " "          " "
1  ( 5 ) " "         " "         " "           " "          " "          "*"
1  ( 6 ) " "         " "         " "           " "          "*"          " "
2  ( 1 ) "*"         " "         "*"           " "          " "          " "
2  ( 2 ) "*"         "*"         " "           " "          " "          " "
2  ( 3 ) "*"         " "         " "           "*"          " "          " "
2  ( 4 ) " "         "*"         "*"           " "          " "          " "
2  ( 5 ) "*"         " "         " "           " "          " "          "*"
2  ( 6 ) "*"         " "         " "           " "          "*"          " "
3  ( 1 ) "*"         "*"         "*"           " "          " "          " "
3  ( 2 ) "*"         " "         "*"           "*"          " "          " "
3  ( 3 ) "*"         " "         "*"           " "          " "          "*"
3  ( 4 ) "*"         "*"         " "           "*"          " "          " "
3  ( 5 ) "*"         " "         "*"           " "          "*"          " "
3  ( 6 ) "*"         "*"         " "           " "          " "          "*"
4  ( 1 ) "*"         "*"         "*"           "*"          " "          " "
4  ( 2 ) "*"         "*"         "*"           " "          " "          "*"
```

4 (3) "*" "*" "*" " " "*" " "

4 (4) "*" " " "*" "*" " " "*"

4 (5) "*" " " "*" "*" "*" " "

4 (6) "*" "*" " " "*" " " "*"

5 (1) "*" "*" "*" "*" " " "*"

5 (2) "*" "*" "*" "*" "*" " "

5 (3) "*" "*" "*" " " "*" "*"

5 (4) "*" " " "*" "*" "*" "*"

5 (5) "*" "*" " " "*" "*" "*"

5 (6) " " "*" "*" "*" "*" "*"

6 (1) "*" "*" "*" "*" "*" "*"

The preferred model, the last one from step 12, is the preferred model. The equation form is as follows: Y = -35484.997 + 67.490*X1 + 91.898*X2 + 59.355*X3 + 69.930*X4 – 144.453*X5 + 1.322*X6. X1 = Gr.Liv.Area, X2 = Garage.Area, X3 = Total.Bsmt.Sf, X4 = Mas.Vnr.Area, X5 = Lot.Frontage, and X6 = Lot.Area.

14. Compare the Preferred Model from Step 13 and Model from Step 12

They are the same model.

Conclusion

The question I presented at the beginning of the report was "what is the best regression model I can create for the Ames Housing Dataset?" Following the steps, provided by Professor Gero, I arrived upon a regression model I am confident in. I also completed my goals of fitting, interpreting, and evaluating regression models using standard functions and diagnostic techniques, correcting issues with overfitting, linearity, multicollinearity, and outliers, and selecting the best model from multiple predictors using automated techniques.

However, I know that this report and my R code can be improved with more time and thought. This is mainly because of the size of this dataset, the lack of knowledge I have of the housing industry, and the limitations of this assignment based on requirements and time. I believe that I have already gone above and beyond. However, with more time, I could arrive on the most accurate regression model for the Ames housing market according to houses sold between 2006 and 2010.

This would take plenty of repetition of steps 12 and 13 by correcting issues and finding the best model. It would also be useful to revisit the correlation matrix and correlation matrix plot to see how correlation changes among variables as I eliminate unusual observations and outliers.

Regardless, again, I am confident that my model is accurate, and this was a great assignment for Regression Diagnostics with R.

References

Baiju, A., Lindsay, H., & St. John, M. (2020 October 5) *Midterm: CAR Package Overview*. RPubs. https://rpubs.com/mjs3pf/carpackage

Bengtsson, H. (2021 September 26) *Package 'R.utils'.* Cran.r-project.org. **https://cran.r-project.org/web/packages/R.utils/R.utils.pdf**

Fox, J. (2021 November 6) *Package 'car'.* Cran.r-project.org. **https://cran.r-project.org/web/packages/car/car.pdf**

Francois, R., Henry, L, Muller, K., & Wickham, H. (NA) *dplyr*. Tidyverse. **https://dplyr.tidyverse.org/**

Francois, R., Henry, L, Muller, K., & Wickham, H. (NA) *ggplot2*. Tidyverse. **https://ggplot2.tidyverse.org/**

Grosjean, P. (2018 March 15) *Package 'pastecs'.* Cran.r-project.org. **https://cran.r-project.org/web/packages/pastecs/pastecs.pdf**

Lumley, T. (2020 January 16) *Package 'leaps: Regression Subset Selection'.* Cran.r-project.org. **https://cran.r-project.org/web/packages/leaps/index.html**

Neuwirth, E. (2015 February 19) *Package 'RColorBrewer'.* Cran.r-project.org.**https://cran.r-project.org/web/packages/RColorBrewer/RColorBrewer.pdf**

Ripley, B. (2022 January 13) *Package 'MASS: Support Functions and Datasets for Venables and Ripley's MASS'.* Cran.r-project.org. https://cran.r-project.org/web/packages/MASS/index.html

Simko, V. & Wei, T. (2021 November 18) *An Introduction to corrplot Package.* Cran.r-projec.org. **https://cran.r-project.org/web/packages/corrplot/vignettes/corrplot-intro.html**

Appendix A: Additional Data Visualizations

Figure A1

>crPlots(model=fit)

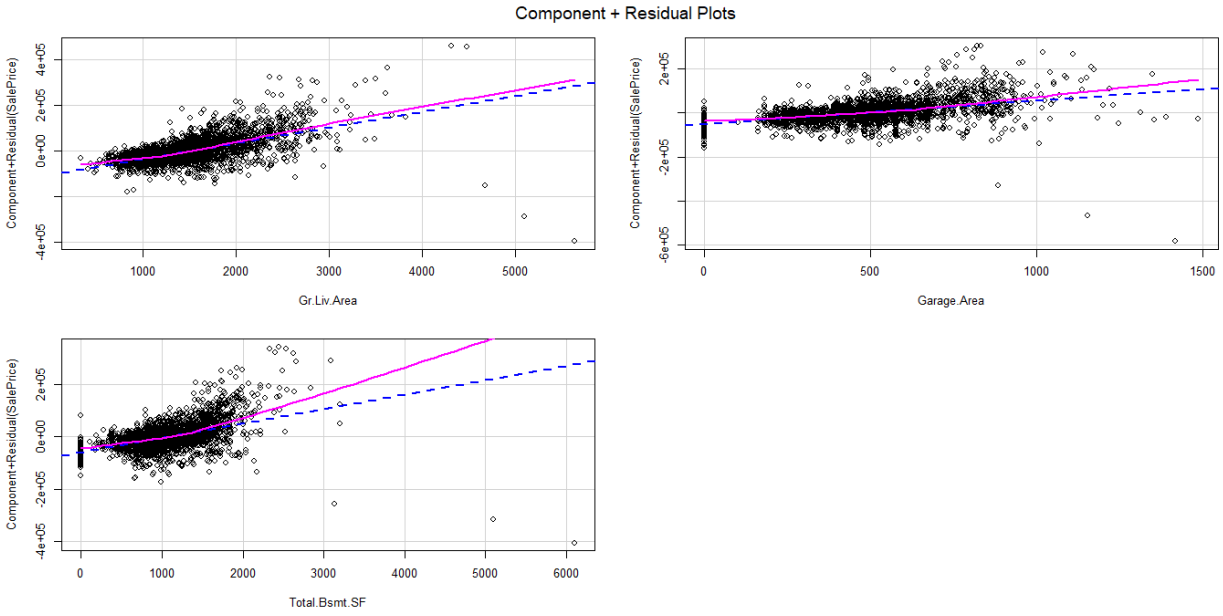


Figure A2

>spreadLevelPlot(fit)

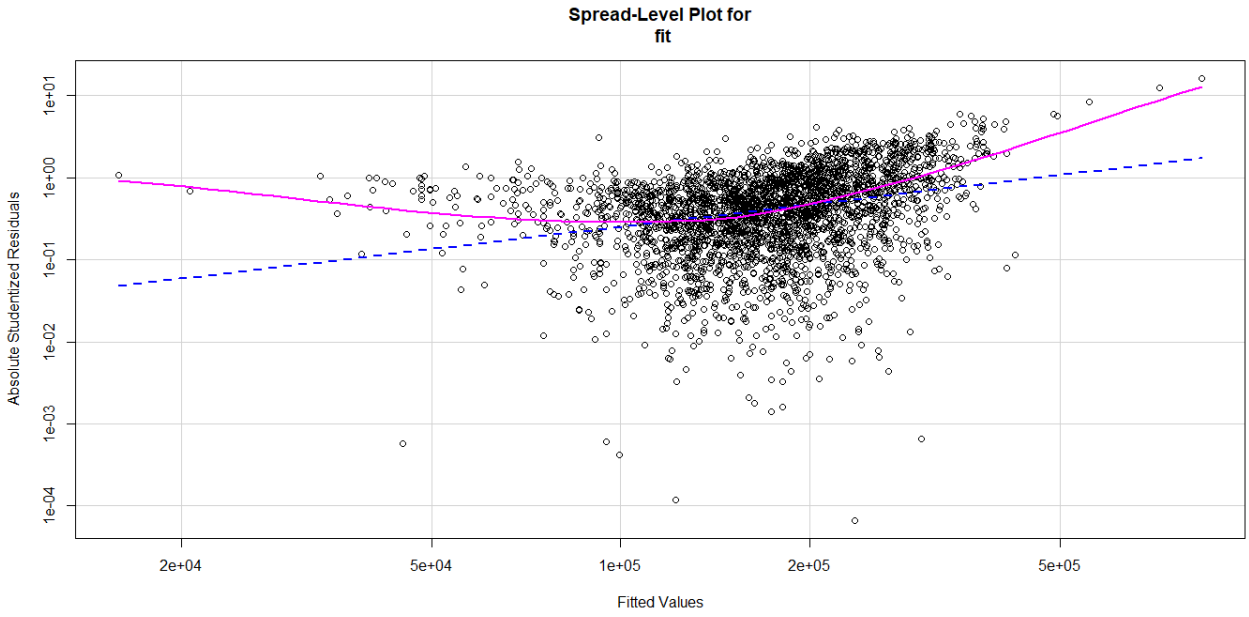


Figure A3

>qqPlot(fit1, labels = row.names(ameshousing3), simulate = TRUE, main = 'Q-Q Plot')

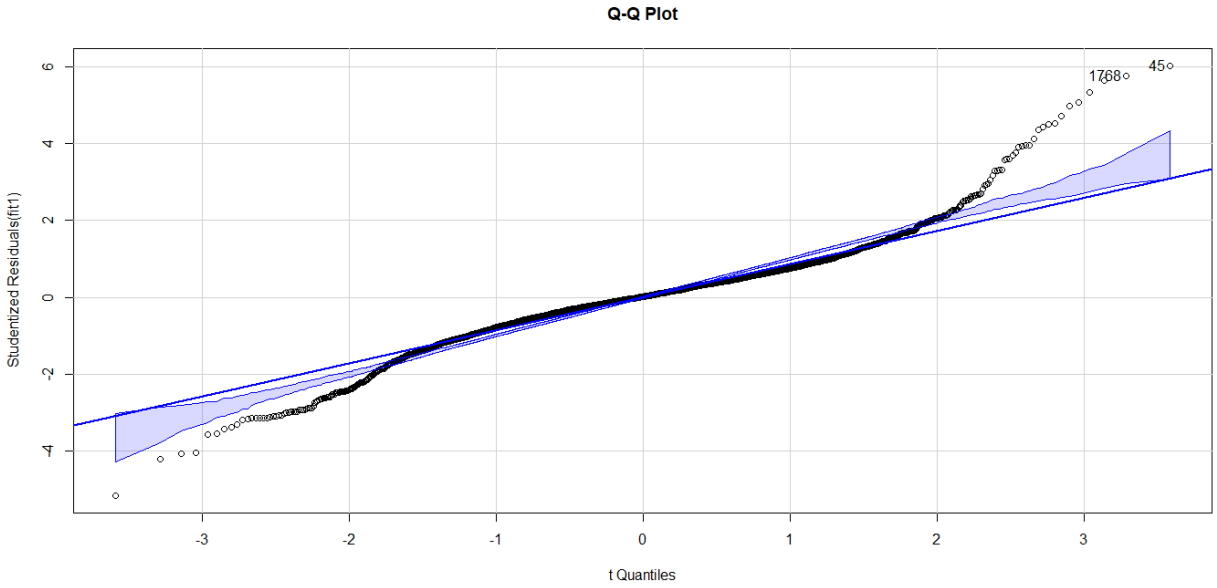


Figure A4

>crPlots(model=fit1)

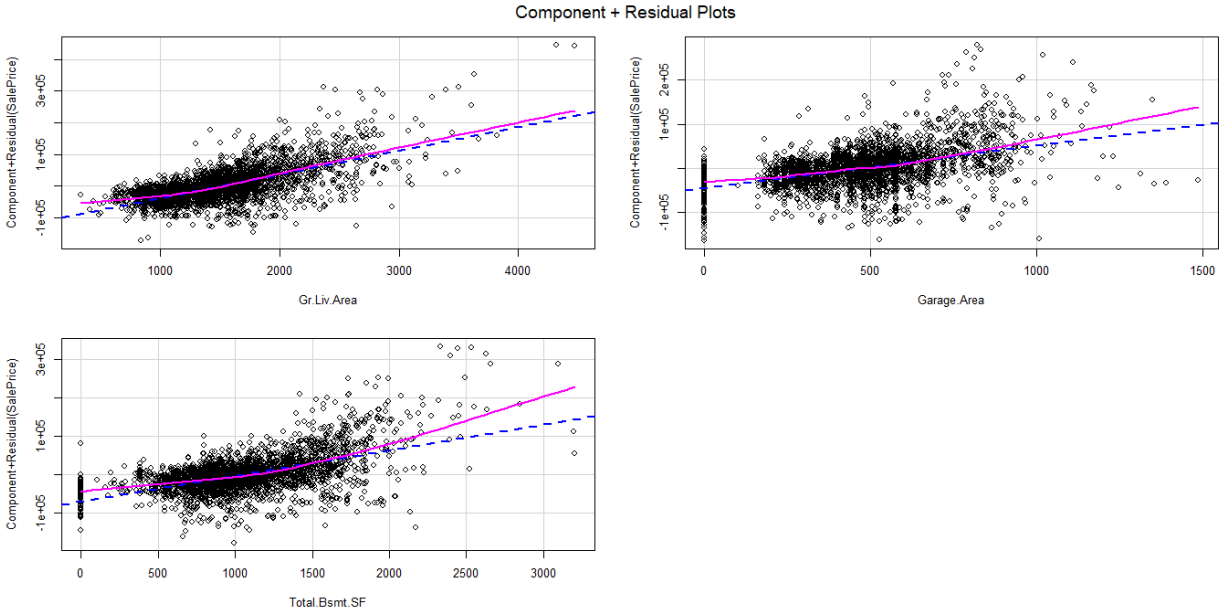


Figure A5

>spreadLevelPlot(fit1)

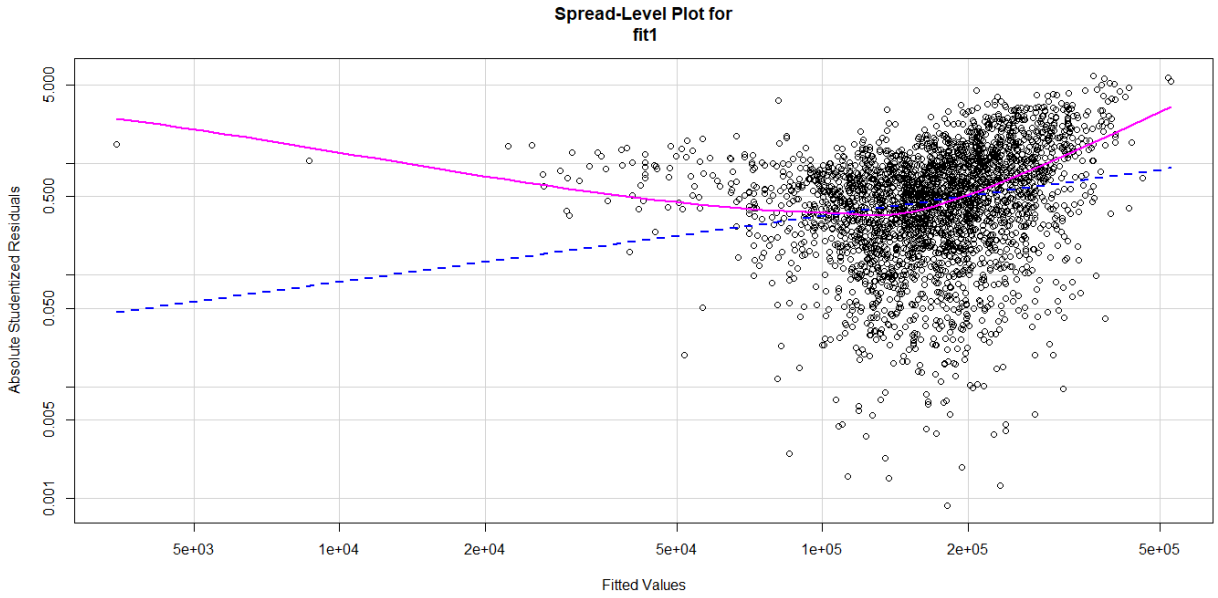


Figure A6

> qqPlot(fit3, labels = row.names(ameshousing4), simulate = TRUE, main = 'Q-Q Plot')

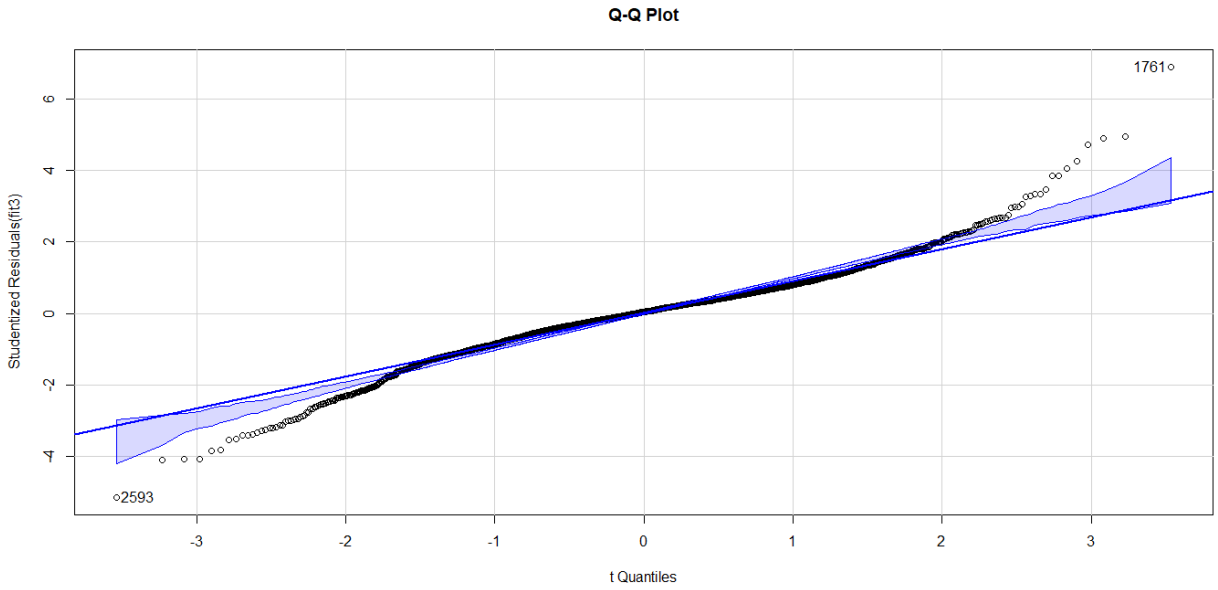


Figure A7

> *crPlots(model=fit3)*

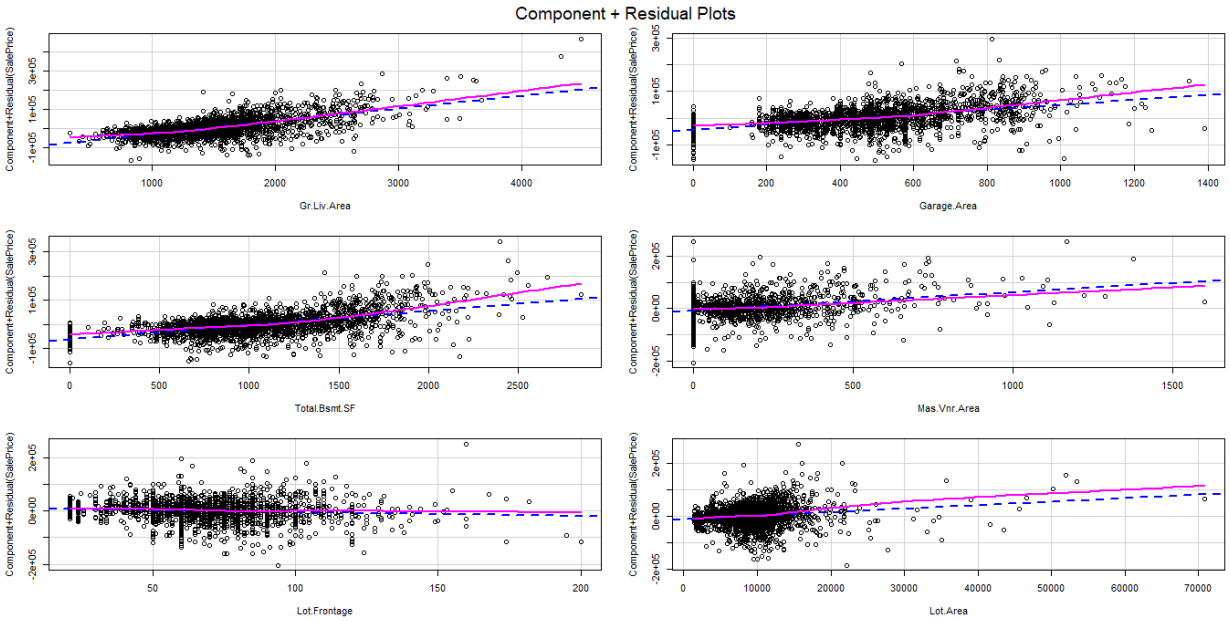
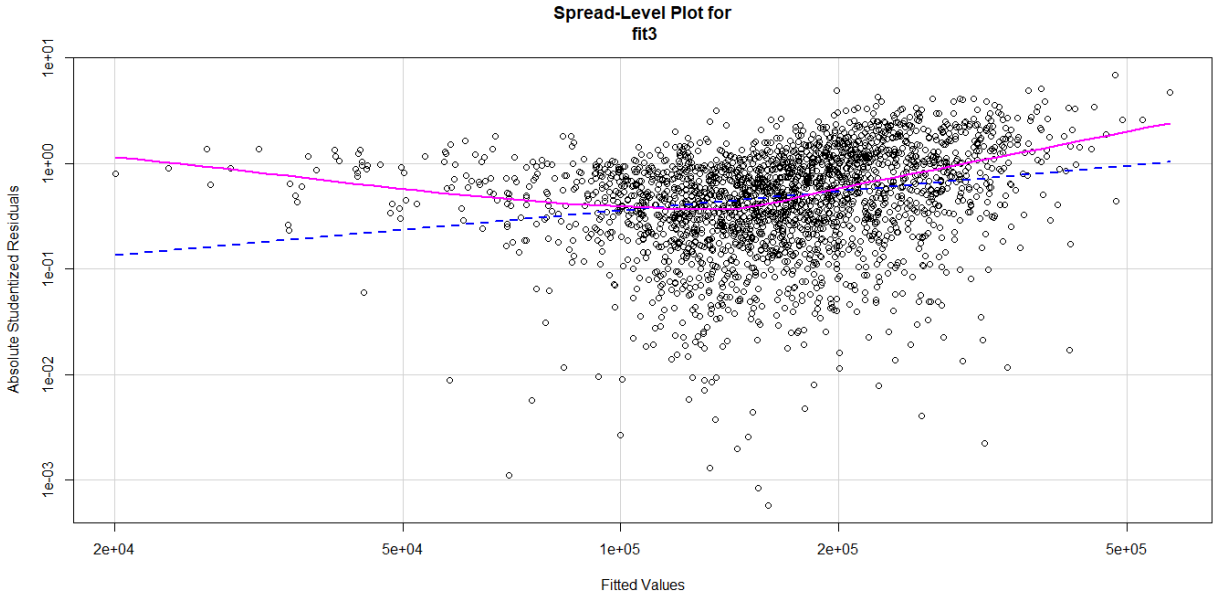


Figure A8

>*spreadLevelPlot(fit3)*



Appendix B: R Code

```
library(dplyr)

library(ggplot2)
```

```
library(corrplot)
library(RColorBrewer)
library(car)
library(pastecs)
library(utils)
library(MASS)
library(leaps)
#################################
##1. Load the Ames Housing DataSet
#################################
setwd("C:/Users/Scott/Desktop/ALY6015")

ameshousing = read.csv("DataSets/AmesHousing.csv")

#################################
##2. Perform Exploratory Data Analysis and use descriptive statistics to describe the data
#################################

####Number of columns and rows
ncol(ameshousing)
nrow(ameshousing)

####View/Glimpse Dataset
glimpse(ameshousing)
View(ameshousing)

####Summary of Dataset
summary(ameshousing)
```

```
####View Histograms of variables
hist(ameshousing$SalePrice/100000,
    main = 'Sale Price Histogram',
    xlab="Hundreds of Thousands of Dollars",
    xaxp=c(0,8,16),
    col = brewer.pal(12, "Set3"),
    labels=T)

####View boxplots of variables
boxplot(ameshousing$SalePrice/100000,
       main = 'Sale Price Boxplot',
       ylab='Hundreds of Thousands of Dollars',
       yaxp=c(0,8,16))
text(y = boxplot.stats(ameshousing$SalePrice/100000)$stats,
    labels = round(boxplot.stats(ameshousing$SalePrice/100000)$stats,2),
    x=1.23,
    cex = 0.8,
    col = "#A11515")
abline(h = mean(ameshousing$SalePrice/100000),
      col = "#7127D1",
      lwd = 1)
text(y=mean(ameshousing$SalePrice/100000)+.2,
    x=0.5,
    paste("Mean:", round(mean(ameshousing$SalePrice/100000),2)),
    col = "#7127D1",
    cex = 0.8,
    pos = 4)
```

```
####View Scatter plots
scatterplot(SalePrice ~ Year.Built, data = ameshousing)

####Selecting only numeric columns
ameshousing1=select_if(ameshousing, is.numeric)
View(ameshousing1)
ameshousing1 = subset(ameshousing1, select = -c(ï..Order,PID))
View(ameshousing1)

scatterplotMatrix(ameshousing1, spread=F, smoother.args = list(lty = 2), main = "Scatter Plot
Matrix")

####Previous Scatter Plot Matrix was illegible, so creating a subset

ameshousing2 <-
ameshousing[c('SalePrice','Lot.Area','Yr.Sold','Gr.Liv.Area','Year.Built','Overall.Qual','Overall.C
ond')]
glimpse(ameshousing2)

scatterplotMatrix(ameshousing2, spread=F, smoother.args = list(lty = 2), main = "Scatter Plot
Matrix")

###################################
##3. Prepare the dataset for modeling by imputing missing values with the variable's mean
##value or any other value that you prefer.
###################################

####Lot.Frontage NA values to 0
ameshousing$Lot.Frontage[is.na(ameshousing$Lot.Frontage)] <- 0
```

```
ameshousing1$Lot.Frontage[is.na(ameshousing1$Lot.Frontage)] <- 0

####Mas.Vnr.Area NA values to 0

ameshousing$Mas.Vnr.Area[is.na(ameshousing$Mas.Vnr.Area)] <- 0

ameshousing1$Mas.Vnr.Area[is.na(ameshousing1$Lot.Frontage)] <- 0


#################################

##4. Use the "cor()" function to produce a correlation matrix of the numeric values.

#################################

cors <- cor(ameshousing1, use = 'pairwise')

View(cors)


#################################

##5. Produce a plot of the correlation matrix, and explain how to interpret it. (hint - check the corrplot or ggcorrplot plot libraries)

#################################

corrplot(cors, type = 'upper',col=brewer.pal(n=8,name="RdYlBu"))


#################################

##6.a. Make a scatter plot for the X continuous variable with the highest correlation with SalePrice.

#################################

scatterplot(SalePrice/100000 ~ Gr.Liv.Area,

        data = ameshousing1,

        main = 'Sale Price by Above Grade (Ground) Living Area Scatter Plot',

        ylab='Sale Price (Hundreds of Thousands of Dollars)',

        xlab='Above Grade Living Area (Square Feet)',

        yaxp=c(0,8,8))

#################################

##6. b Do the same for the X variable that has the lowest correlation with SalePrice.
```

```
#################################
scatterplot(SalePrice/100000 ~ BsmtFin.SF.2,
        data=ameshousing1,
        main='Sale Price by Type 2 Basement Finished Area Scatter Plot',
        ylab='Sale Price (Hundreds of Thousands of Dollars)',
        xlab='Type 2 Basement Finished (Square Feet)',
        yaxp=c(0,8,8))
#################################
##6.c Finally, make a scatter plot between X and SalePrice with the correlation closest to 0.5.
#################################
scatterplot(SalePrice/100000 ~TotRms.AbvGrd,
        data=ameshousing1,
        main='Sale Price by Year Built Scatter Plot',
        ylab='Sale Price (Hundreds of Thousands of Dollars)',
        xlab='Total Rooms Above Grade (Ground)',
        yaxp=c(0,8,8))
#################################
##6. d Interpret the scatter plots and describe how the patterns differ.
#################################

#################################
##7. Using at least 3 continuous variables, fit a regression model in R.
#################################
####I select the 4 continuous variables with the highest correlation values associated with
SalePrice
####For my formula
```

```
fit <-lm(formula = SalePrice ~ Gr.Liv.Area + Garage.Area + Total.Bsmt.SF + X1st.Flr.SF, data=ameshousing1)

summary(fit)

AIC(fit)

####71093.51

BIC(fit)

####71129.4


####It appears that X1st.Flr.SF may be irrelevant to my formula with a Pr(>|t|) of 0.552

####I will run stepAIC() to make sure

stepAIC(fit,direction = "both")

####Start: AIC=62,782.2

####Step: AIC=62,780.56


####Our fit model is just a bit more accurate without X1st.Flr.SF in the the formula

####Thus, I create a new fit formula:

fit <-lm(formula = SalePrice ~ Gr.Liv.Area + Garage.Area + Total.Bsmt.SF, data=ameshousing1)

summary(fit)

AIC(fit)

####71091.86

BIC(fit)

####71121.77


##################################

##8. Report the model in equation form and interpret each coefficient

##of the model in the context of this problem.

##################################

#Y= -29,536.070 + 68.860*X1 + 105.091*X2 + 54.586*X3
```

```
##################################
##9. Use the "plot()" function to plot your regression model.
##Interpret the four graphs that are produced.
##################################

par(mfrow=c(2,2))
plot(fit)
dev.off()
#### There is random dispersion in the Residual vs Fitted plot as hoped for.
#### However, the assumption of linearity appears to not be true,
#### as the plot and the slope of our line increases; our fitted value and residuals
#### increase faster beginning at a fitted value of 2e+5.
####
#### We will look more closely at the other Q-Q plot shortly.
#### One note, before we do so is that 1499,2181, and 2182 all appear to be outliers
#### considering their unusual dispersion.

#### Our scale location plot shows constant variance or homosdasticity.
#### We want to see a random, not distinctive, pattern.
#### Although somewhat random, slope increases exponentionally,starting when
#### our fitted value is roughly 3e+05.  This could be attributed to our outliers however.

####Residuals vs Leverage
#### We can use this plot to identify unusual observations. Again we have entries
#### 1499, 2181, and 2182 as standouts.
```

```
#### I would like to point out that none of these plots are great and I will examine this more
#### in detail later on in the analysis. As a analyst student, not a housing professional, I
#### cannot accurately attribute these flaws in the plots.  A subject matter expert may
#### provide more insight.

####Q-Q Plot
qqPlot(fit, labels = row.names(ameshousing), simulate = TRUE, main = 'Q-Q Plot')
#### The Q-Q Plot evaluates normality.
#### Ideally, we want these plots to fall on a diagonal line.
#### ALthough the points somewhat resemble a diagonal line, it appears most points are
#### outside our 95% confidence envelope.
#### Especially, towards the beginning and end of the plot.
#### This causes some concern, however, most points are focused around the center
#### and there are thousands of entries in this dataset.
#### There are many outliers that are far from the diagonal and 95% envelope.
#### We will keep this in mind when we examine the following graphs


####Component + Residuals Plots
crPlots(model=fit)
#### We use this plot to check for linearity.
#### We have the dotted line, which is the suggested regression line and the solid line
#### which is a lowess line showing our general trends.
#### We can see that the SalesPrice of the Gr.Liv.Area and Garage.Area follow the
#### suggested regression line however SalesPrice begins to increase at a faster rate
#### at around 2000 for Gr.Liv.Area and 800 for Garage.Area.
#### The SalePrice comparatively to Total.Bsmt.SF begins increasing
#### significantly at 2000 Total.Bsmt.Sf
```

```
####Spread-Level Plot for fit - homoscedasticity
spreadLevelPlot(fit)

#### Points are pretty randomly distributed around the horizontal line as we hope to see
#### The lowess line shows the general trend of the data.

###################################
##10. Check your model for multicollinearity and report your findings.
##What steps would you take to correct multicollinearity if it exists?
###################################
####Checking for multicollinearity using the variable inflation factors
vif(fit)
####multicollinearity is relatively low for our predictors
####(in general anything less than 5 is not of concern;
####10 or greater would be cause for concern)

###################################
##11. Check your model for outliers and report your findings.
##Should these observations be removed from the model?
###################################
##24min
outlierTest(model=fit)
####Identified 1499, 2181, 2182 as potential outliers which is something we have
#### been seeing while looking at our plots.


####High-leverage observation
```

```
hat.plot <-function(fit){
  p <- length(coefficients(fit))
  n <- length(fitted(fit))
  plot(hatvalues(fit),main = "Index Plot of Hat Values")
      abline(h=c(2,3)*p/n, col = "red", lty = 2)
      identify(1:n, hatvalues(fit),names(hatvalues(fit)))
}
hat.plot(fit)
###### Creates a plot that identifies high leverage observations
######******Kabikof R in action ()
######Values that fall under .2 are all good according to lab

####Influential observations
cutoff <- 4/(nrow(ameshousing) - length(fit$coefficients)-2)
plot(fit, which = 4, cook.levels = cutoff)
abline(h = cutoff, lty = 2, col = "red")
#### This cooks d plot shows us which observations are most likely influential.
#### Notice 1499, 2181, and 2182 are show up as influential observations and labeled.




##################################
##12. Attempt to correct any issues that you have discovered in your model.
##Did your changes improve the model, why or why not?
##################################

####In all plots that identified outliers, 1499,2181, and 2182 were present
ameshousing3 = ameshousing1[-c(1499, 2181, 2182),]
```

```
fit1 <-lm(formula = SalePrice ~ Gr.Liv.Area + Garage.Area + Total.Bsmt.SF,
data=ameshousing3)

summary(fit1)

###new formula: Y= -46,101.585 + 75.089*X1 + 96.149*X2 + 66.049*X3

AIC(fit1)

###new AIC = 70,492.39

BIC(fit1)

###new BIC = 70,522.3

####After removing the outliers, we identified in the previous step, both the AIC and BIC

#### have lowered suggesting a more accurate model


par(mfrow=c(2,2))

plot(fit1)

dev.off()


####Q-Q Plot

qqPlot(fit1, labels = row.names(ameshousing3), simulate = TRUE, main = 'Q-Q Plot')

####Component + Residuals Plots

crPlots(model=fit1)

####Spread-Level Plot for fit - homoscedasticity

spreadLevelPlot(fit1)

####multicollinearity

vif(fit1)

###vif: 1.367761, 1.477458, 1.370315


####outliers

outlierTest(model=fit1)

###outliers include 45,1768, 1064, 1761, 2593,433,434,2446,2333,2335
```

```
####High-leverage observation
hat.plot <-function(fit1){
  p <- length(coefficients(fit1))
  n <- length(fitted(fit1))
  plot(hatvalues(fit1),main = "Index Plot of Hat Values")
  abline(h=c(2,3)*p/n, col = "red", lty = 2)
  identify(1:n, hatvalues(fit1),names(hatvalues(fit1)))
}
hat.plot(fit1)

####Influential observations
cutoff <- 4/(nrow(ameshousing3) - length(fit1$coefficients)-2)
plot(fit1, which = 4, cook.levels = cutoff)
abline(h = cutoff, lty = 2, col = "red")


###Plenty outliers and entries above the acceptable hat value causes a bit of
###concern so I will add another four variables to fit to see if the model improves
### and reset the dataset

fit2 <-lm(formula = SalePrice ~ Gr.Liv.Area + Garage.Area + Total.Bsmt.SF + Mas.Vnr.Area +
Lot.Frontage + Lot.Area , data=ameshousing)
summary(fit2)
####that looks better, all Pr(>|t|) are close to 0.
#Y = -19,620 + 63.50*X1 + 96.94*X2 + 48.83*X3 + 61.05*X4 + 14.57 *X5 + 0.09979*X6
AIC(fit2)
####AIC=70964.02
```

```
BIC(fit2)
####BIC=71011.87
stepAIC(fit2, direction = "backward")

#### stepAIC() tells us the AIC is lowest with these 4 variables
fit2 <- lm(formula = SalePrice ~ Gr.Liv.Area + Garage.Area + Total.Bsmt.SF +
    Mas.Vnr.Area, data = ameshousing)
summary(fit2)
#### Y = -18,977.922 + 63.912*X1 + 97.318*X2 + 49.242*X3 + 60.873*X4
AIC(fit2)
####AIC=70961.25
BIC(fit2)
####BIC=70997.15

par(mfrow=c(2,2))
plot(fit2)
dev.off()

####Q-Q Plot
qqPlot(fit2, labels = row.names(ameshousing), simulate = TRUE, main = 'Q-Q Plot')
####Component + Residuals Plots
crPlots(model=fit2)
####Spread-Level Plot for fit - homoscedasticity
spreadLevelPlot(fit2)
####multicollinearity
vif(fit2)
####outliers
outlierTest(model=fit2)
```

```
####High-leverage observation
hat.plot <-function(fit2){
  p <- length(coefficients(fit2))
  n <- length(fitted(fit2))
  plot(hatvalues(fit2),main = "Index Plot of Hat Values")
  abline(h=c(2,3)*p/n, col = "red", lty = 2)
  identify(1:n, hatvalues(fit2),names(hatvalues(fit2)))
}
hat.plot(fit)

####Influential observations
cutoff <- 4/(nrow(ameshousing) - length(fit2$coefficients)-2)
plot(fit2, which = 4, cook.levels = cutoff)
abline(h = cutoff, lty = 2, col = "red")

####We get the same significant outliers of entries 1499, 2181, and 2182 so rerun
####with the ameshousing3 dataset which discludes these outliers
fit2 <- lm(formula = SalePrice ~ Gr.Liv.Area + Garage.Area + Total.Bsmt.SF +
           Mas.Vnr.Area, data = ameshousing3)
summary(fit2)
####that looks better, all Pr(>|t|) are close to 0.
#Y = -34904.882 + 69.97*X1 + 86.709*X2 + 60.491*X3 + 64.665*X4
AIC(fit2)
####AIC=69748.16
BIC(fit2)
####BIC=69784
```

```
par(mfrow=c(2,2))
plot(fit2)
dev.off()

####Q-Q Plot
qqPlot(fit2, labels = row.names(ameshousing3), simulate = TRUE, main = 'Q-Q Plot')
####Component + Residuals Plots
crPlots(model=fit2)
####Spread-Level Plot for fit - homoscedasticity
spreadLevelPlot(fit2)
####multicollinearity
vif(fit2)
####outliers
outlierTest(model=fit2)

####High-leverage observation
hat.plot <-function(fit2){
  p <- length(coefficients(fit2))
  n <- length(fitted(fit2))
  plot(hatvalues(fit2),main = "Index Plot of Hat Values")
  abline(h=c(2,3)*p/n, col = "red", lty = 2)
  identify(1:n, hatvalues(fit2),names(hatvalues(fit2)))
}
hat.plot(fit2)
####957, 1266, and 2767 have significantly high hat values
```

```
####Influential observations
cutoff <- 4/(nrow(ameshousing3) - length(fit2$coefficients)-2)
plot(fit2, which = 4, cook.levels = cutoff)
abline(h = cutoff, lty = 2, col = "red")

####In every model fit model we ran with the subset, there were similar outliers
####so we will run the last model with a subset that discludes all outliers
####not just the significant ones
ameshousing4 = ameshousing3[-c(957, 1266, 2767, 1761, 45, 1064, 2593, 433, 434, 1641, 2333,
1768, 2446, 1498, 1761, 1773, 2195, 1768, 1259, 2046, 445, 1558, 291, 1946, 2279, 291, 424,
1538),]
fit3 <- lm(formula = SalePrice ~ Gr.Liv.Area + Garage.Area + Total.Bsmt.SF + Mas.Vnr.Area +
Lot.Frontage + Lot.Area , data=ameshousing4)
stepAIC(fit3, direction = "both")
summary(fit3)
####that looks better, all Pr(>|t|) are close to 0.
#Y = -32,352.437 + 69.197*X1 + 87.506*X2 + 58.718*X3 + 61.322*X4
AIC(fit3)
####AIC=69343.18
BIC(fit3)
####BIC=69378.99


par(mfrow=c(2,2))
plot(fit3)
dev.off()

####Q-Q Plot
qqPlot(fit3, labels = row.names(ameshousing4), simulate = TRUE, main = 'Q-Q Plot')
```

```
####Component + Residuals Plots
crPlots(model=fit3)
####Spread-Level Plot for fit - homoscedasticity
spreadLevelPlot(fit3)
####multicollinearity
vif(fit3)
####outliers
outlierTest(model=fit3)
####only 5 outliers (1761, 2593, 1641, 2333, 1768)
####High-leverage observation
hat.plot <-function(fit3){
  p <- length(coefficients(fit3))
  n <- length(fitted(fit3))
  plot(hatvalues(fit3),main = "Index Plot of Hat Values")
  abline(h=c(2,3)*p/n, col = "red", lty = 2)
  identify(1:n, hatvalues(fit3),names(hatvalues(fit3)))
}
hat.plot(fit3)
####957, 1266, and 2767 have significantly high hat values


####Influential observations
cutoff <- 4/(nrow(ameshousing4) - length(fit3$coefficients)-2)
plot(fit3, which = 4, cook.levels = cutoff)
abline(h = cutoff, lty = 2, col = "red")
```

```
##################################

##13. Use the all subsets regression method to identify the "best" model.

## State the preferred model in equation form.

##################################

leaps <- regsubsets(SalePrice ~ Gr.Liv.Area + Garage.Area + Total.Bsmt.SF + Mas.Vnr.Area +
Lot.Frontage + Lot.Area , data=ameshousing4, nbest = 6)

summary(leaps)

#### Y = -18,977.922 + 63.912*X1 + 97.318*X2 + 49.242*X3 + 60.873*X4


##################################

##14. Compare the preferred model from step 13 with your model from step 12.

##How do they differ? Which model do you prefer and why?

##################################

####I prefer the final model from step 12 as it includes the same variables,

#### However, the subset removed outliers. If the model in step 13

#### did not include these outliers, we would have the same model.
```