**DS6372 – Project 1**

# Introduction

Price prediction is pivotal for real estate. Home sellers want to know the appropriate time to sell and how much profit they can expect from their efforts. Home buyers want to know whether they are getting a fair price, where to look for homes in their budget, and various trade-offs that accompany a purchasing decision. Real estate companies navigate both sides of real estate; hence, they too are a key stakeholder. These stakeholders utilize multiple factors related to real estate to determine the fair price for the property. These same factors can be built into a model for price prediction that assists in taking some of the guess work out of property pricing.

The purpose of this paper is to provide a predictive statistical analysis of house sales in King County, Washington, USA using the King County housing dataset. The analysis has two main objectives with the dataset. The first objective is to build a regression model using the dataset providing metrics off the model and to interpret the regression model. The second objective is to demonstrate an understanding of an advanced analysis workflow.

## Data Description

The data used for this analysis, described in the sections below, comes from the Kaggle website. The dataset may be found on the Kaggle website (<https://www.kaggle.com/harlfoxem/housesalesprediction>). The dataset contains housing related data for King County, Washington, USA representing homes sold from May 2014 through 2015. A detailed listing of King County Housing Sales variables and what they represent may be found on the King County Assessor website (<http://your.kingcounty.gov/assessor/eRealProperty/ResGlossaryOfTerms.html>). The total data set contains 21,613 observations with 20 features or variables. These 20 features contain information quantity and quality-based attributes of a physical property that may interest any of the key stakeholders (prospective home buyer, home seller, real estate company/agent). For example, the data provides answers to questions such as: “How many rooms in the property?”, “What is the condition of the property?”, “Is the property on the waterfront?”, “What is the location of the property?”.

The dataset contains a mix of categorical variables and numeric variables. The categorical variables indicate information a prospective stakeholder would like to understand such as a property grade, number of floors, and condition. The numerical variables indicate information such as square footage that is above or below ground, price, square footage of the interior living space and square footage of the property lot.

## Exploratory Data Analysis (EDA)

The first step in the EDA was to determine which features, if any, could be removed from the dataset due to being not being relevant/valuable for the analysis. It was determined that the following features could be removed from the data set for regression analysis:

* ID – this is a unique field for each record
* Date – this field contains a date/time format which will not be applicable in the analysis

Data quality checks were performed across the data. It was determined via summary statistics (Table 1) that there were no missing values within the data set that needed to be addressed. The data set was clean and contained no missing values.

Analysis was performed to evaluate data plots for all features (Figure 1) to quickly review data normality. Results showed some skew and thus both log (Figure 2) and square root (Figure 3) based transformations were performed. The log transformed variables were looked at to visually satisfy the assumption of linear trend, constant variance and conditional normality. It is assumed that these data are independent due to the nature of the data.

Additional analysis was performed to investigate feature correlation. Through a correlation plot (Figure 4) it was determined that the features: bathrooms, sqft\_living, grade, sqft\_above and sqft\_living15 are predictors with higher correlation to the price target when compared to other factors.

## Analysis Question 1

## Problem Statement

The goal of analysis question 1 is to perform a regression analysis and report on the predictive ability of the model using either a test data set, or some means through CV. This regression analysis should include an interpretation of the regression model coefficients, confidence intervals and hypothesis testing.

## Model Selection

To assist the reader in understanding the models named in this section refer to Table 1 which details out the model name used for reference and a description of the model intent.

In the initial phases of model analysis, the team built Model 0 utilizing the correlated values found in the initial EDA. Upon initial inspection the model had an Adjusted R-Squared value of 0.5441. This value showed there is some correlation between the variables however the team believed the model could be better. The team also reviewed the residual plots for Model 0 (Figure 5) to determine how the data fit for the model and whether all necessary assumptions were met. The residual plots for Model 0 show the following information:

* Residual Plot – There is a slight suspicion of non-constant variance, as we can see a slight funnel shape as the predicted values increase. This is not too serious, and further analysis will proceed with caution
* Studentized Residual Plot – This plot is very similar to the Residual plot although this plot identifies potential outlying observations. This may provide some evidence against the normality assumption. Further analysis may be necessary on these points.
* Q-Q Plot of Residuals - The QQ Plot of residuals displayed below provides some evidence that the residuals are not normally distributed. There are high curves at the tales, in indication of a possible logarithmic curve.
* Histogram of Residuals - The histogram of residuals displayed does not provide strong evidence that the residuals are not normally distributed.

The Q-Q Plot of Residuals was the most concerning of this residual analysis. The team elected to log transform the price in order to determine a better fitting model. This change resulted in Model 0 LT. As seen in Table 4 the Adjusted R-Squared value increases to 0.5698, which indicates a better fit. The residual plots (Figure 6) show some smoothing out of the Q-Q plot curve however there is still curvature.

To create a better Adjusted R-Squared value for modeling and to remove the curvature on the Q-Q plot the team created another set of models (Model 1 and Model 1 LT). These models utilized all the available data and all features in the data set. The models build in this phase resulted in 15 of the available variables being statistically significant. For Model 1, the features: floor, sqft\_basement and sqft\_lot were deemed not statistically significant. The coefficient selection results for Model 1 may be seen in Table 5. Leveraging this model did create a better Adjusted R-Squared of 0.6995 as expected. Figure 7 shows that the same residual issues encountered with Model 0 are present with Model 1 and that log transformation of price would be an applicable solution.

This log transformation resulted in Model 1 LT. As with Model 1, Model 1 LT utilized all available dates and all features in the data set. Model 1 LT built in this phase resulted in 15 of the available variables being significant. The sqft\_above, sqft\_basement and sqft\_lot15 features were deemed not statistically significant, however the floor feature was deemed significant. The results of this initial model run can be seen in Table 6. The log transformation in Model 1 LT resulted in an Adjusted R-Squared of 0.7703. Figure 8 shows that the log transformation has smoothed out the Q-Q plot to a more normal view. The remaining residual plots still look acceptable for analysis.

To create an even stronger model the team elected to build a third model set utilizing all available data and a stepwise feature selection model. As with the other models a log transformation of price was necessary to ensure normality within the data set. For reference Model 2 coefficients may be seen in Table 7 and residual plots in Figure 9. Model 2 LT coefficients may be seen in Table 8 and residual plots in Figure 10. Model 2 LT resulted in an Adjusted R-Squared of 0.7703. The stepwise model selected 17 features for the model, with 15 being statistically significant. In the case of this model the features sqft\_lot15 and sqft\_above were not deemed statistically significant.

In order to determine the predictive ability of Model 2 LT, the model (Model 2 LT TD) was seeded with training data. As with the other models 15 features were deemed statistically significant. As with Model 2 LT, the features sqft\_lot15 and sqft\_above were not deemed statistically significant. The model run against this training set resulted in an Adjusted R-Squared of 0.771. Coefficient selection for Model 2 LT PD can be seen in Table 9. Residual plot (Figure 11) review for this model remained consistent as with the other models. An actual vs predicted plot (Figure 12) was generated against the test and training data sets. From this plot it can been seen that the model does a good job of prediction except for some of the outliers in the data.

Verifying the model outcome, the ASE plot (Figure 13) shows that the test and training models follow similar paths. From the graphic we can see that approximately 17 features are the selection criteria for these models.

## Parameter Interpretation

Interpretation of Parameters will be performed on Model 2-LT as this model was similar to Model 1-LT. The resulting equation for Model 2-LT is:

A sample of the parameter interpretation, using the condition feature, is as follows:

Holding all other variables constant we expect that a 1 unit change in condition will have a 1.065 multiplicative increase in price for the home. There is a 95% confidence that the parameter estimate for condition is between [1.058502338, 1.070812965]

For a full listing of factor impacts see Table 11, which contains the feature, estimate, confidence intervals and converted confidence intervals.

## Conclusion

As can be seen from the Model Selection section above multiple models have been created and verified with various factors to build a model that performs well at prediction of housing prices in King County, Washington, USA. In addition to looking at the Adjust R-Squared metrics for the models the AIC and BIC values were also reviewed. These values, shown in Table 10 add additional validation to the results seen in the Model Selection section. Model 1-LT and Model 2-LT are very similar in all aspects and either one can be used in this case for predictive modeling.

## Analysis Question 2

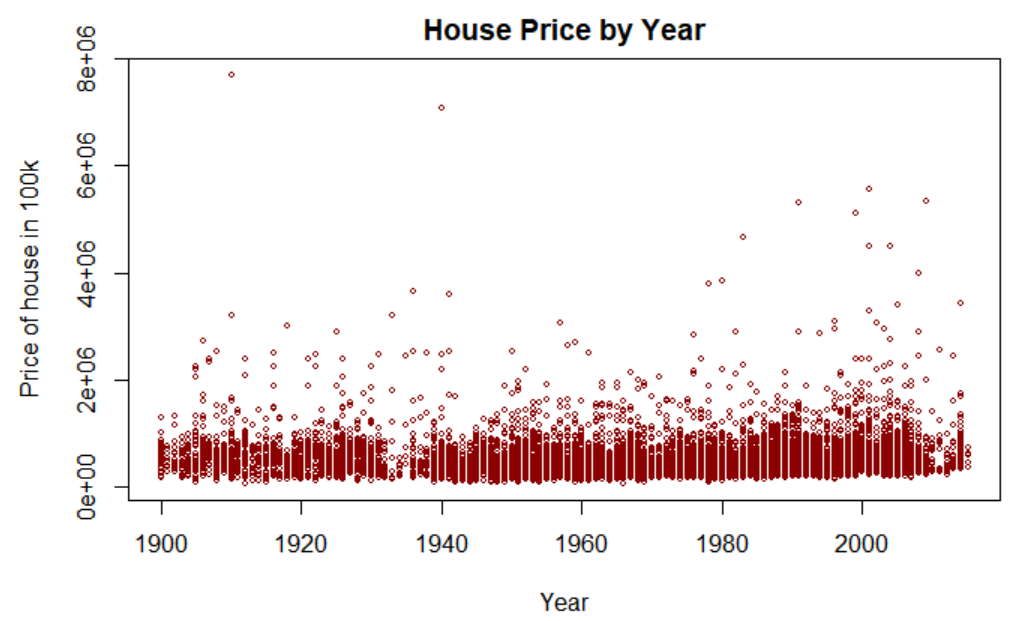
## Problem Statement

While analyzing the house price over this past 100 years, we’ve noticed that housing price in King County has a spike in the 90s, 2000, and 2018. We want to further investigate and see if there’s a housing price change every 20 years by using Anova. We want to validate the statement that housing price has a spike at 20-year increments.

## Main Analysis

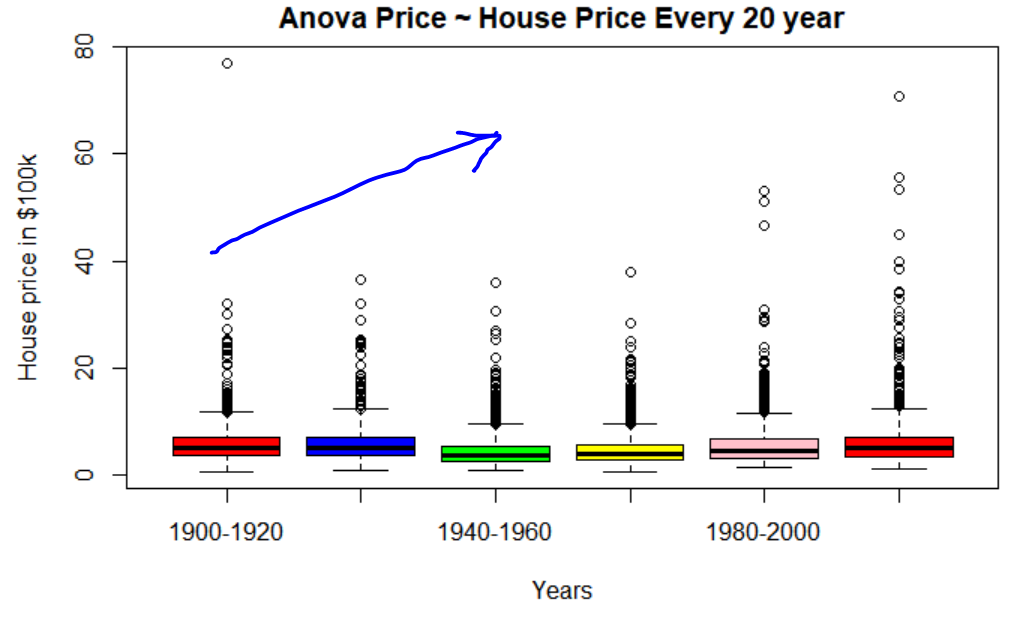
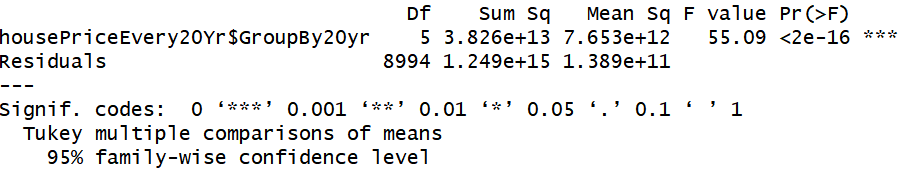
Here, we divide the year build by every 20 years and separate them into categories and run it against the housing price using Anova and TukeyHSD. The hypotheses of interest in our ANOVA are as follows:

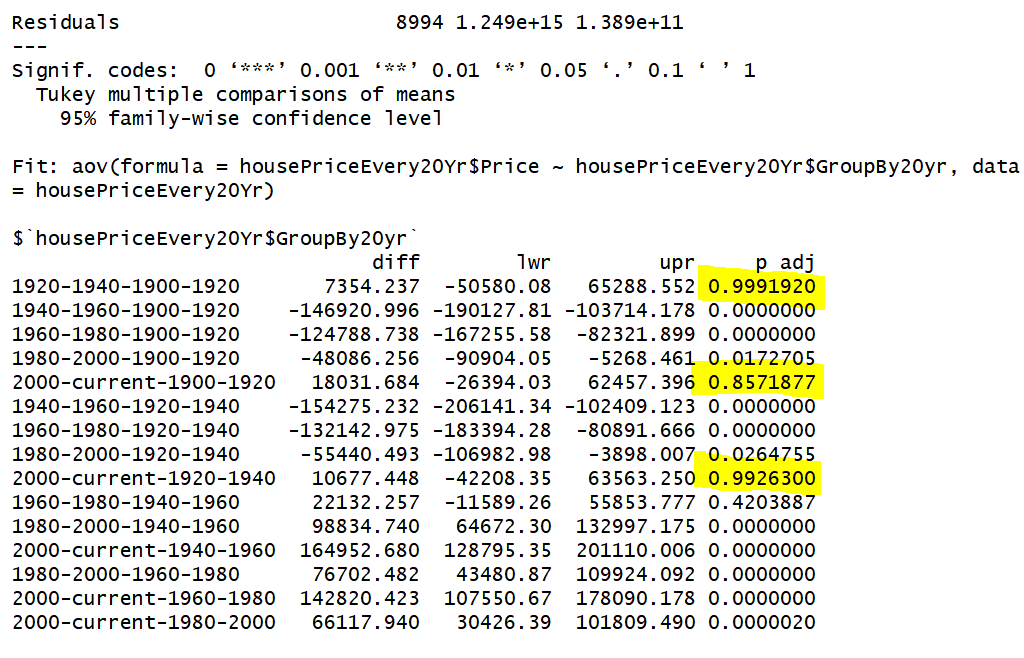
* H0: μ1 = μ2 = μ3 ... = μk  (The housing price average mean are same every 20 years)
* H1: Means are not all equal. (The housing price average mean are not same every 20 years)



## Conclusion

By using Anova, we can see the housing price for the most part on average is stable throughout the century. However, there is a trend showing that housing price is moving upward in the chart (Anova Price ~ House Price Every 20 year) below. The p-value from the Anovo test is less than 0.05 indicating that the time (every 20 years) have significant effect on the housing price. Furthermore, by using TukeyHSD method, the output of TukeyHSD indicates that the differences between all groups are significant, except 3 groups (1920-1940 vs 1900-1920, 2000-Current vs 1900-1920, 2000-Current vs 1920-1940). This is possible due to the great depression and 2008 recession which is something we cannot conclude but would want to future investigate in our next research.



## Appendix

## List of Tables

**Table 1 – EDA Summary Statistics**

## id date price   
## Min. :1.000e+06 20140623T000000: 142 Min. : 75000   
## 1st Qu.:2.123e+09 20140625T000000: 131 1st Qu.: 321950   
## Median :3.905e+09 20140626T000000: 131 Median : 450000   
## Mean :4.580e+09 20140708T000000: 127 Mean : 540088   
## 3rd Qu.:7.309e+09 20150427T000000: 126 3rd Qu.: 645000   
## Max. :9.900e+09 20150325T000000: 123 Max. :7700000   
## (Other) :20833   
## bedrooms bathrooms sqft\_living sqft\_lot   
## Min. : 0.000 Min. :0.000 Min. : 290 Min. : 520   
## 1st Qu.: 3.000 1st Qu.:1.750 1st Qu.: 1427 1st Qu.: 5040   
## Median : 3.000 Median :2.250 Median : 1910 Median : 7618   
## Mean : 3.371 Mean :2.115 Mean : 2080 Mean : 15107   
## 3rd Qu.: 4.000 3rd Qu.:2.500 3rd Qu.: 2550 3rd Qu.: 10688   
## Max. :33.000 Max. :8.000 Max. :13540 Max. :1651359   
##   
## floors waterfront view condition   
## Min. :1.000 Min. :0.000000 Min. :0.0000 Min. :1.000   
## 1st Qu.:1.000 1st Qu.:0.000000 1st Qu.:0.0000 1st Qu.:3.000   
## Median :1.500 Median :0.000000 Median :0.0000 Median :3.000   
## Mean :1.494 Mean :0.007542 Mean :0.2343 Mean :3.409   
## 3rd Qu.:2.000 3rd Qu.:0.000000 3rd Qu.:0.0000 3rd Qu.:4.000   
## Max. :3.500 Max. :1.000000 Max. :4.0000 Max. :5.000   
##   
## grade sqft\_above sqft\_basement yr\_built   
## Min. : 1.000 Min. : 290 Min. : 0.0 Min. :1900   
## 1st Qu.: 7.000 1st Qu.:1190 1st Qu.: 0.0 1st Qu.:1951   
## Median : 7.000 Median :1560 Median : 0.0 Median :1975   
## Mean : 7.657 Mean :1788 Mean : 291.5 Mean :1971   
## 3rd Qu.: 8.000 3rd Qu.:2210 3rd Qu.: 560.0 3rd Qu.:1997   
## Max. :13.000 Max. :9410 Max. :4820.0 Max. :2015   
##   
## yr\_renovated zipcode lat long   
## Min. : 0.0 Min. :98001 Min. :47.16 Min. :-122.5   
## 1st Qu.: 0.0 1st Qu.:98033 1st Qu.:47.47 1st Qu.:-122.3   
## Median : 0.0 Median :98065 Median :47.57 Median :-122.2   
## Mean : 84.4 Mean :98078 Mean :47.56 Mean :-122.2   
## 3rd Qu.: 0.0 3rd Qu.:98118 3rd Qu.:47.68 3rd Qu.:-122.1   
## Max. :2015.0 Max. :98199 Max. :47.78 Max. :-121.3   
##   
## sqft\_living15 sqft\_lot15   
## Min. : 399 Min. : 651   
## 1st Qu.:1490 1st Qu.: 5100   
## Median :1840 Median : 7620   
## Mean :1987 Mean : 12768   
## 3rd Qu.:2360 3rd Qu.: 10083   
## Max. :6210 Max. :871200   
##

**Table 2 – Model Naming Conventions and Descriptions**

|  |  |
| --- | --- |
| **Model Code** | **Model Description** |
| Model 0 | Model based on EDA correlation discovery. Price is not log transformed |
| Model 0 - LT | Model based on EDA correlation discovery. Price is log transformed |
| Model 1 | Model based on reviewing statistically significant outputs. Price is not log transformed |
| Model 1 LT | Model based on reviewing statistically significant outputs. Price is log transformed |
| Model 2 | Model based on reviewing stepwise selection. Price is not log transformed |
| Model 2 - LT | Model based on reviewing stepwise selection. Price is log transformed |
| Model 2 – LT - PD | Model based on reviewing stepwise selection. Price is log transformed and model is executed against training data |

**Table 3 – Model 0 – Coefficient Selection**

##   
## Call:  
## lm(formula = kc\_cleanData\_df$price ~ bathrooms + sqft\_living +   
## grade + sqft\_above + sqft\_living15, data = kc\_cleanData\_df)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1026038 -135316 -22098 98701 4829774   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -6.469e+05 1.351e+04 -47.870 < 2e-16 \*\*\*  
## bathrooms -3.546e+04 3.426e+03 -10.353 < 2e-16 \*\*\*  
## sqft\_living 2.454e+02 4.524e+00 54.251 < 2e-16 \*\*\*  
## grade 1.110e+05 2.462e+03 45.090 < 2e-16 \*\*\*  
## sqft\_above -8.048e+01 4.455e+00 -18.067 < 2e-16 \*\*\*  
## sqft\_living15 2.282e+01 4.027e+00 5.667 1.47e-08 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 247900 on 21607 degrees of freedom  
## Multiple R-squared: 0.5442, Adjusted R-squared: 0.5441   
## F-statistic: 5160 on 5 and 21607 DF, p-value: < 2.2e-16

**Table 4 – Model 0 - LT – Coefficient Selection**

##   
## Call:  
## lm(formula = log(kc\_cleanData\_df$price) ~ bathrooms + sqft\_living +   
## grade + sqft\_above + sqft\_living15, data = kc\_cleanData\_df)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.56303 -0.24679 0.00267 0.23149 1.40650   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.110e+01 1.883e-02 589.326 <2e-16 \*\*\*  
## bathrooms -1.012e-02 4.774e-03 -2.119 0.0341 \*   
## sqft\_living 2.817e-04 6.305e-06 44.680 <2e-16 \*\*\*  
## grade 1.910e-01 3.432e-03 55.662 <2e-16 \*\*\*  
## sqft\_above -1.450e-04 6.209e-06 -23.356 <2e-16 \*\*\*  
## sqft\_living15 9.142e-05 5.612e-06 16.290 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.3455 on 21607 degrees of freedom  
## Multiple R-squared: 0.5699, Adjusted R-squared: 0.5698   
## F-statistic: 5725 on 5 and 21607 DF, p-value: < 2.2e-16

**Table 5 – Model 1 - Coefficient Selection**

##   
## Call:  
## lm(formula = kc\_cleanData\_df$price ~ ., data = kc\_cleanData\_df)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1291725 -99229 -9739 77583 4333222   
##   
## Coefficients: (1 not defined because of singularities)  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 6.690e+06 2.931e+06 2.282 0.02249 \*   
## bedrooms -3.577e+04 1.892e+03 -18.906 < 2e-16 \*\*\*  
## bathrooms 4.114e+04 3.254e+03 12.645 < 2e-16 \*\*\*  
## sqft\_living 1.501e+02 4.385e+00 34.227 < 2e-16 \*\*\*  
## sqft\_lot 1.286e-01 4.792e-02 2.683 0.00729 \*\*   
## floors 6.690e+03 3.596e+03 1.860 0.06285 .   
## waterfront 5.830e+05 1.736e+04 33.580 < 2e-16 \*\*\*  
## view 5.287e+04 2.140e+03 24.705 < 2e-16 \*\*\*  
## condition 2.639e+04 2.351e+03 11.221 < 2e-16 \*\*\*  
## grade 9.589e+04 2.153e+03 44.542 < 2e-16 \*\*\*  
## sqft\_above 3.113e+01 4.360e+00 7.139 9.71e-13 \*\*\*  
## sqft\_basement NA NA NA NA   
## yr\_built -2.620e+03 7.266e+01 -36.062 < 2e-16 \*\*\*  
## yr\_renovated 1.981e+01 3.656e+00 5.420 6.03e-08 \*\*\*  
## zipcode -5.824e+02 3.299e+01 -17.657 < 2e-16 \*\*\*  
## lat 6.027e+05 1.073e+04 56.149 < 2e-16 \*\*\*  
## long -2.147e+05 1.313e+04 -16.349 < 2e-16 \*\*\*  
## sqft\_living15 2.168e+01 3.448e+00 6.289 3.26e-10 \*\*\*  
## sqft\_lot15 -3.826e-01 7.327e-02 -5.222 1.78e-07 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 201200 on 21595 degrees of freedom  
## Multiple R-squared: 0.6997, Adjusted R-squared: 0.6995   
## F-statistic: 2960 on 17 and 21595 DF, p-value: < 2.2e-16

**Table 6 – Model 1 - LT – Coefficient Selection**

##   
## Call:  
## lm(formula = log(kc\_cleanData\_df$price) ~ ., data = kc\_cleanData\_df)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.78817 -0.16139 0.00316 0.15887 1.19290   
##   
## Coefficients: (1 not defined because of singularities)  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -5.073e+00 3.677e+00 -1.379 0.16776   
## bedrooms -1.221e-02 2.373e-03 -5.144 2.71e-07 \*\*\*  
## bathrooms 6.912e-02 4.081e-03 16.936 < 2e-16 \*\*\*  
## sqft\_living 1.512e-04 5.501e-06 27.494 < 2e-16 \*\*\*  
## sqft\_lot 4.712e-07 6.011e-08 7.838 4.78e-15 \*\*\*  
## floors 7.515e-02 4.511e-03 16.661 < 2e-16 \*\*\*  
## waterfront 3.712e-01 2.178e-02 17.046 < 2e-16 \*\*\*  
## view 6.040e-02 2.684e-03 22.501 < 2e-16 \*\*\*  
## condition 6.264e-02 2.950e-03 21.235 < 2e-16 \*\*\*  
## grade 1.589e-01 2.700e-03 58.855 < 2e-16 \*\*\*  
## sqft\_above -1.529e-05 5.470e-06 -2.795 0.00520 \*\*   
## sqft\_basement NA NA NA NA   
## yr\_built -3.411e-03 9.114e-05 -37.419 < 2e-16 \*\*\*  
## yr\_renovated 3.659e-05 4.586e-06 7.979 1.54e-15 \*\*\*  
## zipcode -6.459e-04 4.138e-05 -15.610 < 2e-16 \*\*\*  
## lat 1.400e+00 1.347e-02 103.968 < 2e-16 \*\*\*  
## long -1.592e-01 1.648e-02 -9.660 < 2e-16 \*\*\*  
## sqft\_living15 9.857e-05 4.325e-06 22.791 < 2e-16 \*\*\*  
## sqft\_lot15 -2.610e-07 9.191e-08 -2.840 0.00452 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.2524 on 21595 degrees of freedom  
## Multiple R-squared: 0.7704, Adjusted R-squared: 0.7703   
## F-statistic: 4263 on 17 and 21595 DF, p-value: < 2.2e-16

**Table 7 – Model 2 – Coefficient Selection**

##   
## Call:  
## lm(formula = price ~ sqft\_living + lat + view + grade + yr\_built +   
## waterfront + bedrooms + bathrooms + zipcode + long + condition +   
## sqft\_above + sqft\_living15 + yr\_renovated + sqft\_lot15 +   
## sqft\_lot + floors, data = kc\_cleanData\_df)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1291725 -99229 -9739 77583 4333222   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 6.690e+06 2.931e+06 2.282 0.02249 \*   
## sqft\_living 1.501e+02 4.385e+00 34.227 < 2e-16 \*\*\*  
## lat 6.027e+05 1.073e+04 56.149 < 2e-16 \*\*\*  
## view 5.287e+04 2.140e+03 24.705 < 2e-16 \*\*\*  
## grade 9.589e+04 2.153e+03 44.542 < 2e-16 \*\*\*  
## yr\_built -2.620e+03 7.266e+01 -36.062 < 2e-16 \*\*\*  
## waterfront 5.830e+05 1.736e+04 33.580 < 2e-16 \*\*\*  
## bedrooms -3.577e+04 1.892e+03 -18.906 < 2e-16 \*\*\*  
## bathrooms 4.114e+04 3.254e+03 12.645 < 2e-16 \*\*\*  
## zipcode -5.824e+02 3.299e+01 -17.657 < 2e-16 \*\*\*  
## long -2.147e+05 1.313e+04 -16.349 < 2e-16 \*\*\*  
## condition 2.639e+04 2.351e+03 11.221 < 2e-16 \*\*\*  
## sqft\_above 3.113e+01 4.360e+00 7.139 9.71e-13 \*\*\*  
## sqft\_living15 2.168e+01 3.448e+00 6.289 3.26e-10 \*\*\*  
## yr\_renovated 1.981e+01 3.656e+00 5.420 6.03e-08 \*\*\*  
## sqft\_lot15 -3.826e-01 7.327e-02 -5.222 1.78e-07 \*\*\*  
## sqft\_lot 1.286e-01 4.792e-02 2.683 0.00729 \*\*   
## floors 6.690e+03 3.596e+03 1.860 0.06285 .   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 201200 on 21595 degrees of freedom  
## Multiple R-squared: 0.6997, Adjusted R-squared: 0.6995   
## F-statistic: 2960 on 17 and 21595 DF, p-value: < 2.2e-16

**Table 8 – Model 2 - LT – Coefficient Selection**

##   
## Call:  
## lm(formula = log(price) ~ grade + lat + sqft\_living + yr\_built +   
## view + bathrooms + sqft\_living15 + condition + waterfront +   
## floors + zipcode + long + sqft\_lot + yr\_renovated + bedrooms +   
## sqft\_lot15 + sqft\_above, data = kc\_cleanData\_df)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.78817 -0.16139 0.00316 0.15887 1.19290   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -5.073e+00 3.677e+00 -1.379 0.16776   
## grade 1.589e-01 2.700e-03 58.855 < 2e-16 \*\*\*  
## lat 1.400e+00 1.347e-02 103.968 < 2e-16 \*\*\*  
## sqft\_living 1.512e-04 5.501e-06 27.494 < 2e-16 \*\*\*  
## yr\_built -3.411e-03 9.114e-05 -37.419 < 2e-16 \*\*\*  
## view 6.040e-02 2.684e-03 22.501 < 2e-16 \*\*\*  
## bathrooms 6.912e-02 4.081e-03 16.936 < 2e-16 \*\*\*  
## sqft\_living15 9.857e-05 4.325e-06 22.791 < 2e-16 \*\*\*  
## condition 6.264e-02 2.950e-03 21.235 < 2e-16 \*\*\*  
## waterfront 3.712e-01 2.178e-02 17.046 < 2e-16 \*\*\*  
## floors 7.515e-02 4.511e-03 16.661 < 2e-16 \*\*\*  
## zipcode -6.459e-04 4.138e-05 -15.610 < 2e-16 \*\*\*  
## long -1.592e-01 1.648e-02 -9.660 < 2e-16 \*\*\*  
## sqft\_lot 4.712e-07 6.011e-08 7.838 4.78e-15 \*\*\*  
## yr\_renovated 3.659e-05 4.586e-06 7.979 1.54e-15 \*\*\*  
## bedrooms -1.221e-02 2.373e-03 -5.144 2.71e-07 \*\*\*  
## sqft\_lot15 -2.610e-07 9.191e-08 -2.840 0.00452 \*\*   
## sqft\_above -1.529e-05 5.470e-06 -2.795 0.00520 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.2524 on 21595 degrees of freedom  
## Multiple R-squared: 0.7704, Adjusted R-squared: 0.7703   
## F-statistic: 4263 on 17 and 21595 DF, p-value: < 2.2e-16

**Table 9 – Model 2 – LT - PD – Coefficient Selection**

##   
## Call:  
## lm(formula = log(price) ~ grade + lat + sqft\_living + yr\_built +   
## view + condition + bathrooms + sqft\_living15 + waterfront +   
## floors + zipcode + long + yr\_renovated + sqft\_lot + bedrooms +   
## sqft\_lot15 + sqft\_basement, data = trainData)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.75820 -0.16151 0.00316 0.15904 1.20367   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -4.160e+00 4.146e+00 -1.003 0.31577   
## grade 1.595e-01 3.023e-03 52.762 < 2e-16 \*\*\*  
## lat 1.402e+00 1.509e-02 92.938 < 2e-16 \*\*\*  
## sqft\_living 1.372e-04 5.151e-06 26.638 < 2e-16 \*\*\*  
## yr\_built -3.402e-03 1.022e-04 -33.280 < 2e-16 \*\*\*  
## view 5.905e-02 3.032e-03 19.474 < 2e-16 \*\*\*  
## condition 6.694e-02 3.303e-03 20.271 < 2e-16 \*\*\*  
## bathrooms 6.721e-02 4.584e-03 14.662 < 2e-16 \*\*\*  
## sqft\_living15 9.853e-05 4.841e-06 20.351 < 2e-16 \*\*\*  
## waterfront 3.771e-01 2.509e-02 15.031 < 2e-16 \*\*\*  
## floors 7.590e-02 5.071e-03 14.968 < 2e-16 \*\*\*  
## zipcode -6.559e-04 4.666e-05 -14.056 < 2e-16 \*\*\*  
## long -1.586e-01 1.841e-02 -8.615 < 2e-16 \*\*\*  
## yr\_renovated 3.592e-05 5.156e-06 6.966 3.39e-12 \*\*\*  
## sqft\_lot 4.207e-07 6.826e-08 6.163 7.29e-10 \*\*\*  
## bedrooms -1.132e-02 2.648e-03 -4.275 1.92e-05 \*\*\*  
## sqft\_lot15 -2.691e-07 1.014e-07 -2.653 0.00798 \*\*   
## sqft\_basement 1.087e-05 6.166e-06 1.762 0.07805 .   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.2528 on 17274 degrees of freedom  
## Multiple R-squared: 0.7676, Adjusted R-squared: 0.7674   
## F-statistic: 3356 on 17 and 17274 DF, p-value: < 2.2e-16

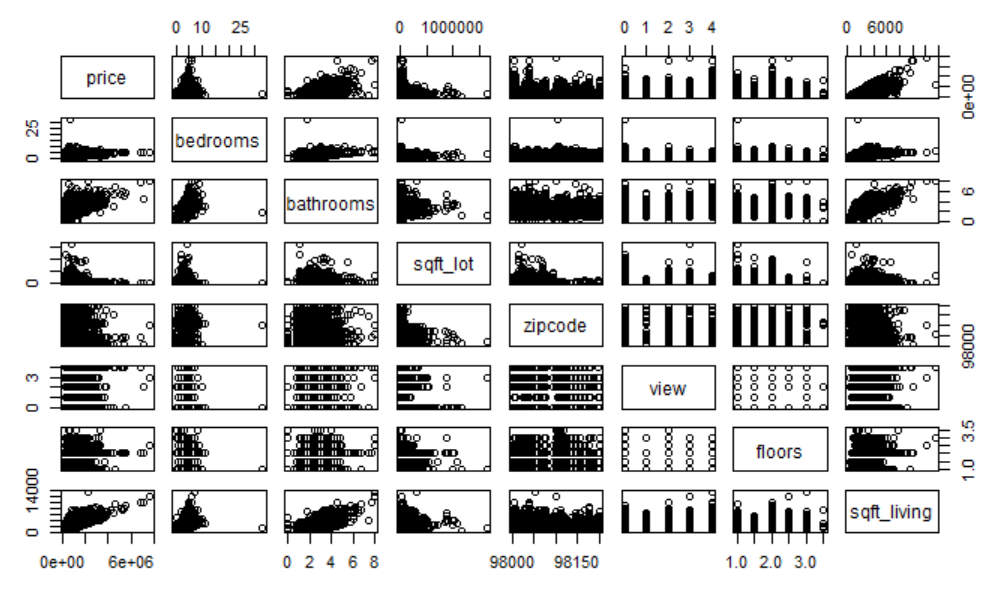
**Table 10 – AIC, BIC and Adjusted R-Squared Model Comparison**

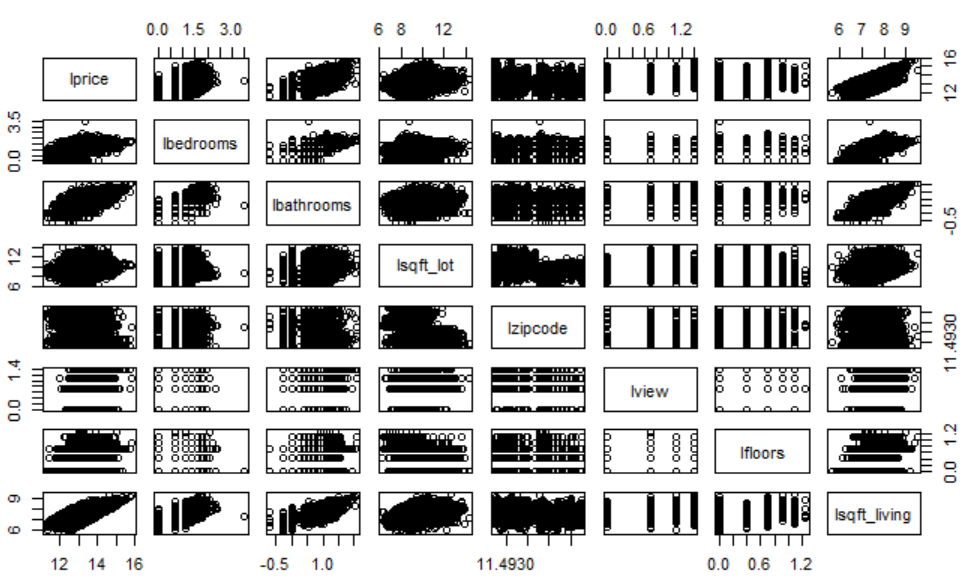
|  |  |  |  |
| --- | --- | --- | --- |
| **Model** | **AIC** | **BIC** | **Adj. R-Squared (%)** |
| Model 0 | 598240.50 | 598296.40 | 0.5491 (54.91%) |
| Model 0-LT | 15399.97 | 15455.83 | 0.5698 (56.98%) |
| Model 1 | 589243.60 | 589395.20 | 0.6995 (69.95%) |
| Model 1-LT | 1851.661 | 2003.301 | 0.7703 (77.03%) |
| Model 2 | 589243.60 | 589395.20 | 0.6995 (69.95%) |
| Model 2-LT | 1851.661 | 203.301 | 0.7703 (77.03%) |
| Model 2-LT-PD | 1531.877 | 1670.279 | 0.7674 (76.74%) |

**Table 11 – Model 2-LT Parameter Estimates**

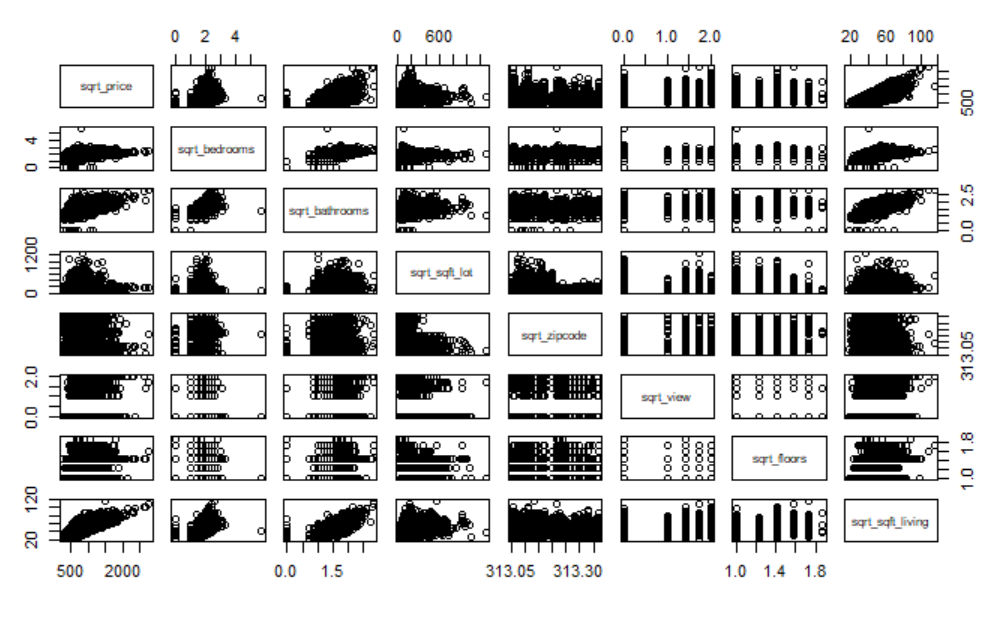
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Feature** | **Estimate** | **Confidence Interval** | | **Converted Estimate** | **Converted CI** | |
|  |  | **2.50%** | **97.50%** |  | **2.50%** | **97.50%** |
| Intercept | -5.07E+00 | -1.23E+01 | 2.13E+00 | 0.006263601 | 4.64202E-06 | 8.456851997 |
| Grade | 1.59E-01 | 1.54E-01 | 1.64E-01 | 1.172220719 | 1.166071842 | 1.178481564 |
| Lat | 1.40E+00 | 1.37E+00 | 1.43E+00 | 4.055199967 | 3.94954349 | 4.163628793 |
| Sqft\_Living | 1.51E-04 | 1.40E-04 | 1.62E-04 | 1.000151211 | 1.000140473 | 1.000162041 |
| yr\_built | -3.41E-03 | -3.59E-03 | -3.23E-03 | 0.996594811 | 0.996417265 | 0.996773344 |
| view | 6.04E-02 | 5.51E-02 | 6.57E-02 | 1.062261366 | 1.056691595 | 1.067870448 |
| bathrooms | 6.91E-02 | 6.11E-02 | 7.71E-02 | 1.071564789 | 1.063030562 | 1.080175513 |
| sqft\_living15 | 9.86E-05 | 9.01E-05 | 1.07E-04 | 1.000098575 | 1.000090093 | 1.000107049 |
| condition | 6.26E-02 | 5.69E-02 | 6.84E-02 | 1.064643499 | 1.058502338 | 1.070812965 |
| waterfront | 3.71E-01 | 3.29E-01 | 4.14E-01 | 1.449472939 | 1.38889949 | 1.512673477 |
| floors | 7.52E-02 | 6.63E-02 | 8.40E-02 | 1.078045846 | 1.068560248 | 1.087622944 |
| zipcode | -6.46E-04 | -7.27E-04 | -5.65E-04 | 0.999354309 | 0.999273278 | 0.999435378 |
| long | -1.59E-01 | -1.91E-01 | -1.27E-01 | 0.852825777 | 0.82576182 | 0.880853372 |
| sqft\_lot | 4.71E-07 | 3.53E-07 | 5.89E-07 | 1.000000471 | 1.000000353 | 1.000000589 |
| yr\_renovated | 3.66E-05 | 2.76E-05 | 4.56E-05 | 1.000036591 | 1.000027603 | 1.00004558 |
| bedrooms | -1.22E-02 | -1.69E-02 | -7.56E-03 | 0.98786424 | 0.983281639 | 0.992471774 |
| sqft\_lot15 | -2.61E-07 | -4.41E-07 | -8.09E-08 | 0.999999739 | 0.999999559 | 0.999999919 |
| sqft\_above | -1.53E-05 | -2.60E-05 | -4.57E-06 | 0.99998471 | 0.999973993 | 0.999995434 |

## List of Figures

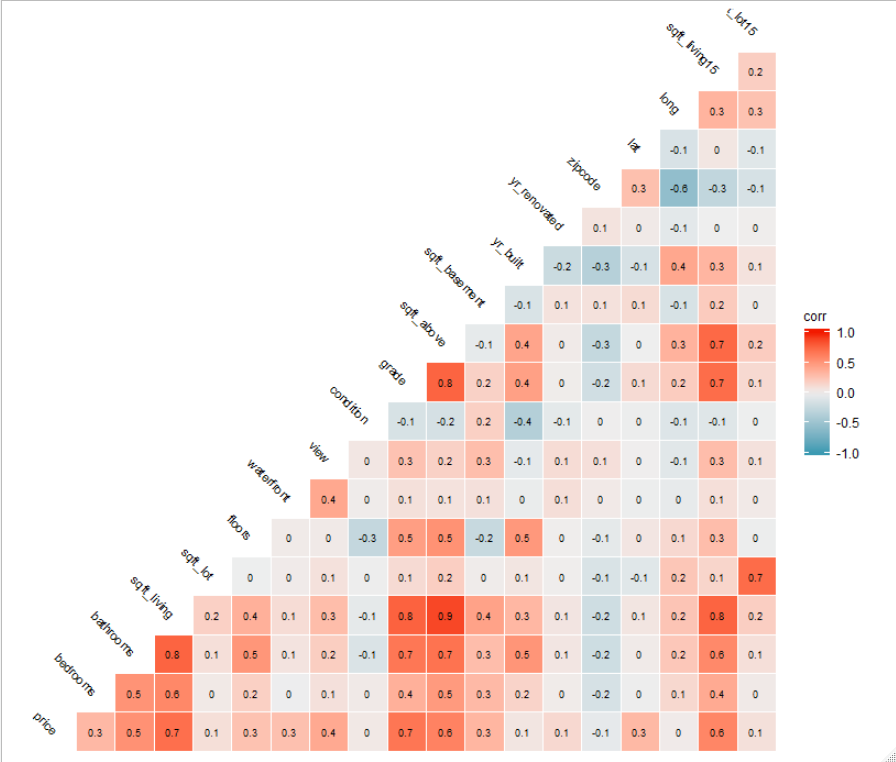
**Figure 1 – Plots of Original Data**

**Figure 2 – Plots of Log Transformed Data**

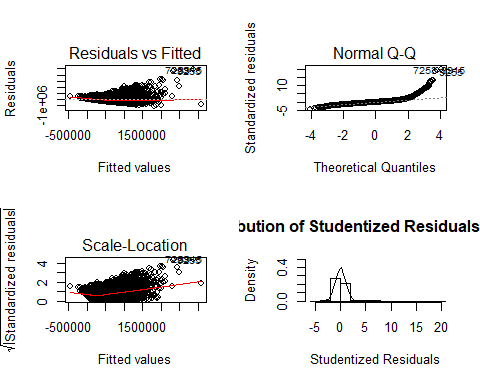
**Figure 3 – Plots of Square Root Transformed Data**



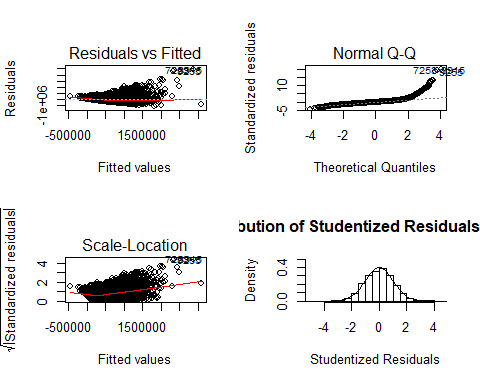
**Figure 4 – Correlation Matrix of Features**



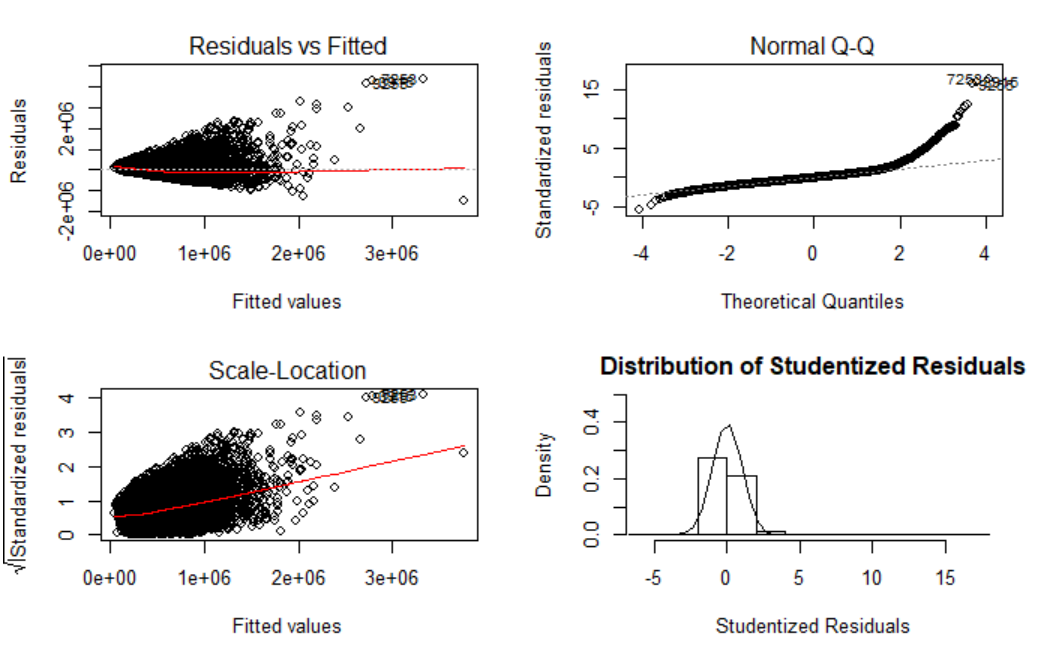
**Figure 5 – Residual Plots for Model 0**



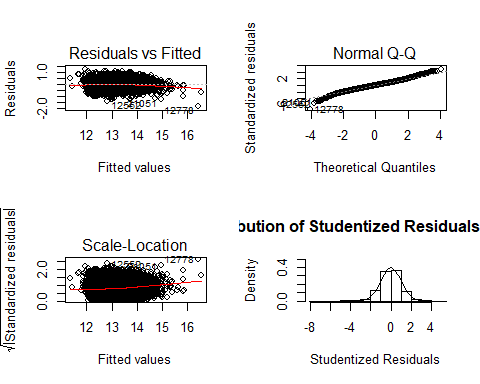
**Figure 6 - Residual Plots for Model 0-LT**



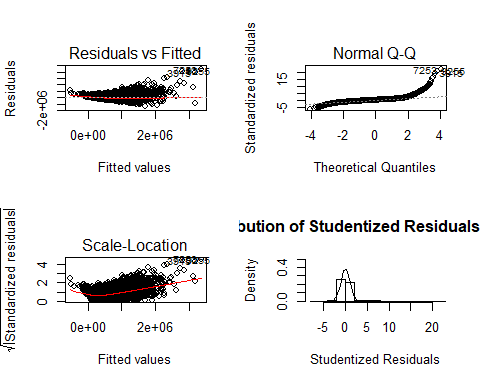
**Figure 7 – Residual Plots for Model 1**



**Figure 8 – Residual Plots for Model 1-LT**



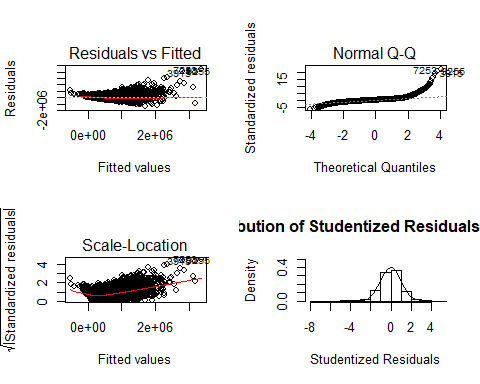
**Figure 9 – Residual Plots for Model 2**



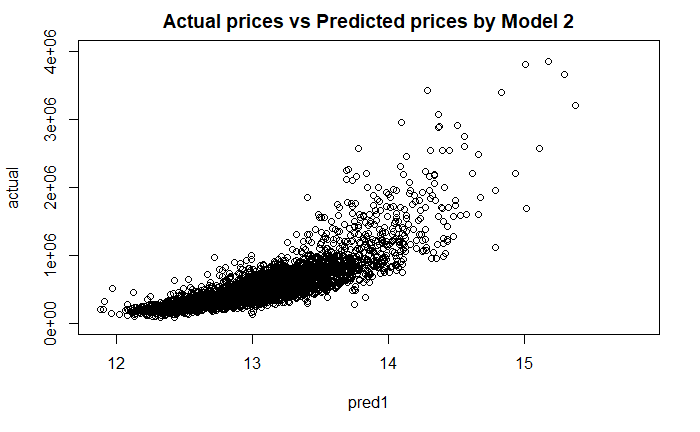
**Figure 10 – Model 2 – Residual Plots for Model 2-LT**

## 

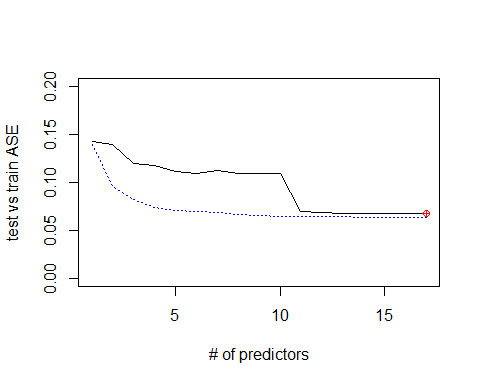
**Figure 11 – Model 2 – Residual Plots for Model 2-LT-PD**



**Figure 12 – Actual Prices vs Predicted Prices**



**Figure 13 – ASE Plot**



## Completed Code

All source code and materials may be found in the following GitHub repository:

<https://github.com/chiawang/DS6372Project1_Group5.git>

A copy of the source code from the R-Markdown used for analysis is below.

---

title: "DS 6372 - Project 1"

author: "Brandon Croom, Queena Wang"

date: "September 15, 2019"

output:

word\_document: default

html\_document: default

---

```{r setup, include=FALSE}

knitr::opts\_chunk$set(echo = TRUE)

```

## R Markdown

## Initial setup of data and library loading

```{r load libraries and data}

library(lubridate)

library(dplyr)

library(ggplot2)

library(dichromat)

library(leaflet)

library(GGally)

library(gridExtra)

library(DT)

library(caret)

library(ISLR)

library(leaps)

library(tseries)

library(forecast)

library(ggmap)

## Directories listed below for easy copy paste

#qw\_directory = "C:/Users/chiawa/DS6372Project1\_Group5/

kc\_house\_data.csv"

#bc\_directory = "C:/Users/croomb/OneDrive - BAT/Desktop/Personal

Training/SMU/DS 6372 - Applied Statistics/Projects/

DS6372Project1\_Group5/"

## Set the working directory

setwd("C:/Users/croomb/OneDrive - BAT/Desktop/Personal Training/

SMU/DS 6372 - Applied Statistics/Projects/

DS6372Project1\_Group5/")

## Read in the CSV file

kc\_data\_df <- read.csv(file="kc\_house\_data.csv", header=TRUE,

lbathrooms<-log(kc\_cleanData\_df$bathrooms)

lsqft\_lot<-log(kc\_cleanData\_df$sqft\_lot)

lzipcode<-log(kc\_cleanData\_df$zipcode)

lview <- log(kc\_cleanData\_df$view)

lfloors<-log(kc\_cleanData\_df$floors)

lsqft\_living<-log(kc\_cleanData\_df$sqft\_living)

pairs (~lprice + lbedrooms + lbathrooms + lsqft\_lot + lzipcode +

lview + lfloors + lsqft\_living, data = kc\_cleanData\_df)

## Try a sqrt transformation with every relevant variable logged.

sqrt\_price <- sqrt(kc\_cleanData\_df$price)

sqrt\_bedrooms<-sqrt(kc\_cleanData\_df$bedrooms)

sqrt\_bathrooms<-sqrt(kc\_cleanData\_df$bathrooms)

sqrt\_sqft\_lot<-sqrt(kc\_cleanData\_df$sqft\_lot)

sqrt\_zipcode<-sqrt(kc\_cleanData\_df$zipcode)

sqrt\_view <- sqrt(kc\_cleanData\_df$view)

sqrt\_floors<-sqrt(kc\_cleanData\_df$floors)

sqrt\_sqft\_living<-sqrt(kc\_cleanData\_df$sqft\_living)

pairs (~sqrt\_price + sqrt\_bedrooms + sqrt\_bathrooms +

sqrt\_sqft\_lot + sqrt\_zipcode + sqrt\_view + sqrt\_floors +

sqrt\_sqft\_living, data = kc\_cleanData\_df)

## Number of Bedroom vs Price

qplot(kc\_cleanData\_df$bedrooms, kc\_cleanData\_df$price,

data=kc\_cleanData\_df)

cor.test(kc\_cleanData\_df$bedrooms, kc\_cleanData\_df$price)

## Number of Bathrooms vs Price

qplot(kc\_cleanData\_df$bathrooms,

kc\_cleanData\_df$price,data=kc\_cleanData\_df)

cor.test(kc\_cleanData\_df$bathrooms, kc\_cleanData\_df$price)

## sqft\_lot of Bedroom vs Price

qplot(kc\_cleanData\_df$sqft\_lot, kc\_cleanData\_df$price,

data=kc\_cleanData\_df)

cor.test(kc\_cleanData\_df$sqft\_lot, kc\_cleanData\_df$price)

## zipcode vs price

qplot(kc\_cleanData\_df$zipcode, kc\_cleanData\_df$price,

data=kc\_cleanData\_df)

cor.test(kc\_cleanData\_df$zipcode, kc\_cleanData\_df$price)

## Create training and test data sets for model predictions

## Create training and test data sets

set.seed(1234)

trainIndex = createDataPartition(kc\_cleanData\_df$price,p=.

8,list=FALSE,times=1)

trainData = kc\_cleanData\_df[trainIndex,]

testdata = kc\_cleanData\_df[-trainIndex,]

```

## Model 0 - Initial Attempt Correlated Data - All Data

```{r model 0 Initial Correlated Data }

## model 0: build model based off correlated values in EDA

model\_0 <- lm(kc\_cleanData\_df$price ~ bathrooms + sqft\_living +

grade + sqft\_above + sqft\_living15 , data = kc\_cleanData\_df)

#Get model summary

summary(model\_0)

#Get model confidence intervals

confint(model\_0)

#Get model AIC and BIC

AIC(model\_0)

BIC(model\_0)

```

```{r model 0 residual analysis initial}

##plotting the model fit

par(mfrow=c(2,2))

plot(model\_0 , which=c(1:3))

##Histogram with normal curve

##Store studentized residuals

model\_0\_studresbrain <- rstudent(model\_0)

##Histogram

hist(model\_0\_studresbrain, freq=FALSE, main="Distribution of

Studentized Residuals(Model 1) ",

xlab="Studentized Residuals", ylab="Density", ylim=c(0,0.5))

##Create range of x-values for normal curve

xfit2 <- seq(min(model\_0\_studresbrain)-1,

max(model\_0\_studresbrain)+1, length=40)

BIC(model\_0\_LT)

```

```{r model 0 residual analysis initial log}

##plotting the model fit

par(mfrow=c(2,2))

plot(model\_0 , which=c(1:3))

##Histogram with normal curve

##Store studentized residuals

model\_0\_studresbrain <- rstudent(model\_0\_LT)

##Histogram

hist(model\_0\_studresbrain, freq=FALSE, main="Distribution of

Studentized Residuals(Model 1) ",

xlab="Studentized Residuals", ylab="Density", ylim=c(0,0.5))

##Create range of x-values for normal curve

xfit2 <- seq(min(model\_0\_studresbrain)-1,

max(model\_0\_studresbrain)+1, length=40)

##Generate values from the normal distribution at the specified

values

yfit2 <- (dnorm(xfit2))

##Add the normal curve

lines(xfit2, yfit2, ylim=c(0,0.5))

```

## Model 1 - Initial Attempt - All Data

```{r model 1 Initial }

## model 1: build model based off of all from EDA

model\_1 <- lm(kc\_cleanData\_df$price ~. , data = kc\_cleanData\_df)

#Get model summary

summary(model\_1)

#Get model confidence intervals

confint(model\_1)

#Get model AIC and BIC

AIC(model\_1)

BIC(model\_1)

```

##Add the normal curve

lines(xfit2, yfit2, ylim=c(0,0.5))

```

## Model 1 - Log Transform of Price - All Data

```{r model 1 LogTransform}

## model 1: build model based off of all values from EDA

model\_1\_LT <- lm(log(kc\_cleanData\_df$price) ~. , data =

kc\_cleanData\_df)

#Get model summary

summary(model\_1\_LT)

#Get model confidence intervals

confint(model\_1\_LT)

#Get model AIC and BIC

AIC(model\_1\_LT)

BIC(model\_1\_LT)

```

```{r model 1 residual analysis log transform}

##plotting the model fit

par(mfrow=c(2,2))

plot(model\_1\_LT, which=c(1:3))

##Histogram with normal curve

##Store studentized residuals

model\_1\_studresbrain <- rstudent(model\_1\_LT)

##Histogram

hist(model\_1\_studresbrain, freq=FALSE, main="Distribution of

Studentized Residuals(Model 1) ",

xlab="Studentized Residuals", ylab="Density", ylim=c(0,0.5))

##Create range of x-values for normal curve

xfit2 <- seq(min(model\_1\_studresbrain)-1,

max(model\_1\_studresbrain)+1, length=40)

##Generate values from the normal distribution at the specified

values

yfit2 <- (dnorm(xfit2))

##Add the normal curve

sqft\_lot + floors, data = kc\_cleanData\_df)

#Get model summary

summary(model\_2)

#Get model confidence intervals

confint(model\_2)

#Get model AIC and BIC

AIC(model\_2)

BIC(model\_2)

```

```{r model 2 residual analysis}

##plotting the model fit

par(mfrow=c(2,2))

plot(model\_2 , which=c(1:3))

##Histogram with normal curve

##Store studentized residuals

model\_2\_studresbrain <- rstudent(model\_2)

##Histogram

hist(model\_2\_studresbrain, freq=FALSE, main="Distribution of

Studentized Residuals(Model 2) ",

xlab="Studentized Residuals", ylab="Density", ylim=c(0,0.5))

##Create range of x-values for normal curve

xfit2 <- seq(min(model\_2\_studresbrain)-1,

max(model\_2\_studresbrain)+1, length=40)

##Generate values from the normal distribution at the specified

values

yfit2 <- (dnorm(xfit2))

##Add the normal curve

lines(xfit2, yfit2, ylim=c(0,0.5))

```

## Model 2 - Stepwise Prediction Function (Log Price) - All Data

```{r - use backward, forward, and stepwise to find the best

model log price}

## adjusted R^2 - higher is better

## MSPE (Mean Square Prediction Error) - lower is better as it

measure the distance the prediction are from the acutual value

#Get model AIC and BIC

AIC(model\_2\_LT)

BIC(model\_2\_LT)

```

```{r model 2 residual analysis all}

##plotting the model fit

par(mfrow=c(2,2))

plot(model\_2\_LT , which=c(1:3))

##Histogram with normal curve

##Store studentized residuals

model\_2\_studresbrain <- rstudent(model\_2\_LT)

##Histogram

hist(model\_2\_studresbrain, freq=FALSE, main="Distribution of

Studentized Residuals(Model 2) ",

xlab="Studentized Residuals", ylab="Density", ylim=c(0,0.5))

##Create range of x-values for normal curve

xfit2 <- seq(min(model\_2\_studresbrain)-1,

max(model\_2\_studresbrain)+1, length=40)

##Generate values from the normal distribution at the specified

values

yfit2 <- (dnorm(xfit2))

##Add the normal curve

lines(xfit2, yfit2, ylim=c(0,0.5))

```

Based on running all the analysis on all data for both a linear

regression model and a stepwise model, the better fit comes with

taking a log transformation of the price when looking at the

residual plots. Keeping that in mind let's move forward with

testing how well the models predict for price

## Model 2 - Stepwise Prediction Function (Log Price) -

Predictive Data

```{r - use backward, forward, and stepwise to find the best

model predictive}

## adjusted R^2 - higher is better

## MSPE (Mean Square Prediction Error) - lower is better as it

measure the distance the prediction are from the acutual value

#Get model AIC and BIC

AIC(model\_2\_PD)

BIC(model\_2\_PD)

```

```{r model 2 residual analysis predictive}

##plotting the model fit

par(mfrow=c(2,2))

plot(model\_2 , which=c(1:3))

##Histogram with normal curve

##Store studentized residuals

model\_2\_studresbrain <- rstudent(model\_2\_PD)

##Histogram

hist(model\_2\_studresbrain, freq=FALSE, main="Distribution of

Studentized Residuals(Model 2) ",

xlab="Studentized Residuals", ylab="Density", ylim=c(0,0.5))

##Create range of x-values for normal curve

xfit2 <- seq(min(model\_2\_studresbrain)-1,

max(model\_2\_studresbrain)+1, length=40)

##Generate values from the normal distribution at the specified

values

yfit2 <- (dnorm(xfit2))

##Add the normal curve

lines(xfit2, yfit2, ylim=c(0,0.5))

#Test the model prediction based on the training data

pred1 = predict(object=model\_2\_PD,newdata=testdata)

res1 = cbind(testdata$price,pred1)

colnames(res1) <- c("actual", "pred1")

res1 <- as.data.frame(res1)

plot(actual~pred1, data=res1,ylim=c(0,4000000),

main = "Actual prices vs Predicted prices by Model 2")

```

```{r model prediction}

#Look at an exhaustive method for model build with an NVMax of 18

(the maximumum number of variables)

which.min(testASE)

#Display model coefficients based on testASE

coef(model\_2\_Final,which.min(testASE))

#Build ASE graph

par(mfrow=c(1,1))

plot(1:17,testASE,type="l",xlab="# of predictors",ylab="test vs

train ASE",ylim=c(0,0.2))

index<-which(testASE==min(testASE))

points(index,testASE[index],col="red",pch=10)

rss<-summary(model\_2\_Final)$rss

lines(1:17,rss/(nrow(trainData)),lty=3,col="blue") #Dividing by

training data sample side since ASE=RSS/sample size

```

## Object 2: Anova

```{r correlation, echo=FALSE}

# Compare models

# anova(model\_1, model\_2) - compare House Price every 20 years

kc\_cleanData\_df <-kc\_data\_df[rowSums(is.na(kc\_data\_df)) == 0,]

# Drop Date From model

kc\_cleanData\_df <- kc\_cleanData\_df[,c(1,3:21)]

kc\_cleanData\_df

# Grab a smaller set of the data

kc\_cleanData\_df <- kc\_cleanData\_df[1:9000,]

attach(kc\_cleanData\_df)

glimpse(kc\_cleanData\_df)

pricesIn100k <- kc\_cleanData\_df$price/100000

plot(kc\_cleanData\_df$price ~ kc\_cleanData\_df$yr\_built,

data=kc\_cleanData\_df,

cex =.5,

col ='dark red',

main = 'House Price by Year',

xlab ='Year',

ylab ='Price of house in 100k')

# Divide House Price by every 20 years

housePriceEvery20Yr <- data.frame(Price = kc\_cleanData\_df$price,

else {

housePriceEvery20Yr$GroupBy20yr[i] <- '2000-current'

}

}

housePriceEvery20Yr$GroupBy20yr <-

as.factor(housePriceEvery20Yr$GroupBy20yr)

# We are diving years into factors of 20 year spans

# Run Anova

anova <- aov(housePriceEvery20Yr$Price ~

housePriceEvery20Yr$GroupBy20yr, data=housePriceEvery20Yr )

# Summary stats on Anova

summary(anova)

# Run Tukey's test on Anova

TukeyHSD(anova)

plot(pricesIn100k ~ housePriceEvery20Yr$GroupBy20yr,

data = housePriceEvery20Yr ,

main = 'Anova Price ~ House Price Every 20 year',

xlab = 'Years',

ylab = 'House price in $100k',

col = c ('red', 'blue', 'green', 'yellow', 'pink')

)

```

Note that the `echo = FALSE` parameter was added to the code

chunk to prevent printing of the R code that generated the plot.