House Prices: Advanced Regression Techniques

By: Kosi Okeke and Victoria Hernandez

Introduction

A prominent real estate firm operating in Ames, Iowa seeks to better understand its knowledge of how the living area influences the sales prices of houses. The firm is specifically interested in the NAmes, Edwards, and BrkSide neighborhoods. This analysis aims to equip Century 21 Ames with precise, quantifiable insights that could significantly enhance its sales strategy. By focusing on these neighborhoods, the study aims to provide tailored information that reflects the unique characteristics and market demands of these areas. Century 21 Ames is also interested in using predictive modeling for the sales prices in Ames, Iowa encompassing all neighborhoods. In our second analysis we hope to provide a well-calibrated model to serve potential buyers, sellers, and investors in making informed decisions and providing insights into the dynamic housing market of Ames.

Data Description

The dataset is derived from the "Ames Housing dataset," originally prepared by Dean De Cock. It has 2390 observations, each representing individual property sales in Ames, Iowa from 2006 to 2010. The subset for analysis 1 includes 383 observations with 81 features ranging from sale price to type of alley access. The variables of interest to analysis 1 are SalePrice, GrLivArea (above-ground living area in square feet), and Neighborhood. For analysis 2 the training set has 1460 observations each with 81 features including the target variable of SalePrice. The test set contains 1459 observations with 80 features excluding the SalePrice variable.

Analysis Question 1

Century 21 Ames hopes to refine its market strategies by seeking to understand how the living area (square footage) of houses influences their sales process in the NAmes, Edwards, and BrkSide neighborhoods.

The primary objective is to construct a multiple linear regression model to estimate the relationship between living area (GrLivArea) and sale price of houses (SalePrice), while also taking into account the influence of different neighborhoods, specifically, the NAmes, Edwards, and BrkSide neighborhoods. To do this, we filtered the train.csv data set for only the NAmes, Edwards, and BrkSIde neighborhoods and removed NA values. This resulted in a filtered data set (filtered_data). Initial exploratory data analysis shows the histograms (Figures 1 & 2) of the distribution of sales price and distribution of living area are both quite right-skewed. In the scatter plots (Figures 3 &4), you can see there are positive linear relationships between SalePrice and GrLivArea in each of the neighborhoods.

An initial simple linear model is constructed to estimate how SalePrice is related to the GrLivArea in the three neighborhoods of interest. The initial model yields a root mean squared error (RMSE) of $28,550 and an adjusted R squared score of 0.44. These metrics suggest that the model explains about 44% of the variance in house sale prices within the NAmes, Edwards, and BrkSide neighborhoods, The scatter plot titled “Residuals vs Fitted Values” (Figure 5) does not show a clear pattern of variance however the points are trending towards one area and ideally we’d like to see a random scatter throughout around the horizontal line. The QQ residual plot shows a slight linear relationship but has several outliers. Next, any outliers or influential observations should be identified using Cook’s Distance. The Cook’s Distance plot (Figure 7) shows an analysis shows 23 potentially influential observations in the training data as indicated by the Cook’s Distance greater than the threshold of $4/n$, where n is the number of observations.

After addressing and removing the influential points the model was refit. This complex model created a residual scatter plot (figure 8) that has a more random scatter suggesting homoscedasticity. The QQ plot (figure 9) deviates slightly from the line at the tails but follows the line more closely. The adjusted R squared score is 0.5141 and an RMSE of $24,920. This suggests the model without the high influence points predicts 51% of the variability in house sale prices within the selected neighborhoods.

After comparing the complex model to transformed models the complex model with the influential points was removed, and no log transformations were chosen. The model estimates the intercept to be $24,290.93, this is the baseline in BrkSide. For each additional sq ft of living area, the sale prices increase by about $82.19 in BrkSide.

The houses in the Edwards neighborhood have a starting sales price of $43,625.33 higher than BrkSide, holding the living area constant. Similar to Edwards, the homes in NAmes start at $58,599.38 higher than BrkSide, holding the living area constant.

The interaction term of GrLivArea:NeighborhoodEdwards indicates that the additional price per square foot in the Edwards neighborhood is $39.78 less compared to BrkSide. This suggests that while larger houses in Edwards are still more than homes in BrkSide the price per square foot increases at a slower rate than in BrkSide. Similarly, the interaction term of GrLivArea: NeighborhoodNAmes indicates the price per square foot is $35.01 less compared to BrkSide. This suggests the marginal price increase per square foot in NAmes is lower compared to BrkSide.

For this complex model (model2) the standard deviation of residuals or RMSE is about $21,370. This gives an idea of typical errors in predictions of sale prices. The Adjusted R-squared is 0.5141 indicating that about 51.41% of the variability of sale price is explained by the model and this accounts for the number of predictors. The F-statistic of 76.97 and p-value of <2.2e-16 strongly suggest the model is statistically significant and there is enough evidence to suggest GrLivArea and Neighborhood do have an effect on predicting the sale price that is different from zero.

In conclusion, the analysis yielded a robust model that captures the nuanced impact of living area on sale prices, influenced significantly by neighborhood. The significant interaction terms indicate the importance of considering neighborhood context in price estimation and reveal different price sensitivity to living area across neighborhoods. The model provides Century 21 Ames with valuable insights for advising clients and adjusting sales strategies.

R Shiny: Price v. Living Area Chart

At this link, you can view our shiny app that allows you to select any combination of neighborhoods and log transform SalePrice or GrLivArea to view the relationship in a scatterplot. https://torih1541.shinyapps.io/Project2App/

Analysis Question 2

Restatement of Problem

The objective of Analysis 2 is to develop the most predictive model for estimating sales prices of homes across all neighborhoods in Ames, Iowa. Analysis limited to techniques covered Course 6371 (excluding methods such as random forests or other advanced techniques). We would like to produce four models based on Forward Selection, Backwards Elimination, Stepwise Selection, and then lastly a custom model that is up to us.

Model Selection

As an aside, I ran the selection processes on *all* the columns that did not include missing values or NAs. I wanted to let R choose variables for me, and from there I would analyze the models. The initial custom model was a combination of running a forward selection process to narrow down variables coupled with intuition.

Type of Selection

Forward

Variables were sequentially added to the model based on their individual contribution to explaining the variance in the target variable. The model achieved an adjusted R-squared of 0.9298 and a cross-validated prediction error (CV PRESS) of 0.1805.

Backward

Starting with a model containing all available predictor variables, non-significant variables were systematically removed. This approach yielded an adjusted R-squared of 0.9302 and a CV PRESS of 0.1799.

Stepwise

This method involves iteratively adding or removing variables from the model based on their statistical significance, as determined by criteria such as p-values or information criteria. My criterion was the BIC. The resulting model had an adjusted R-squared of 0.9148 and a CV PRESS of 0.2004.

Custom

I initially ran a simple forward selection, then from there, removed variables based on VIF. After removing these variables and analyzing the scores, I found that the best model to come up with was the original forward-selected variables. They happened to give the second lowest CV PRESS which can be used as a measure of how well the prediction model will run on new, unseen data.

Checking Assumptions

It is worth mentioning that the distribution of the variable “SalePrice” which we are looking to Predict is right-skewed. So, to counteract this, we will log SalePrice to bring the distribution to a normal one. Also, this whole section of assumptions can be found in the Appendix under the section titled “Checking Assumptions!”.

- We can assume independence of variables. If I created a variable named “TotalSF" that consisted of both 1st Floor SF + 2nd Floor SF but still left in those individual variables, then it would make sense that our variables were not independent of each other. I left all variables as is to counteract this phenomenon and any mistakes I could potentially make.
- The residuals follow a normal distribution generally, we can see (In the Appendix: Checking Assumptions!) that there is a small amount of points that pull the end of the distribution off the Q-Q line, but it is minimal in scope compared to the overall amount of observations.
- There is a linearity between most of the variables and the response variable “Log_SalePrice”.
- In the appendix, you can see the VIF values for the initial model named “log_forward”. There seems to be a few variables (Neighborhood, MSZoning and Sale Condition) that have VIF’s > 5. It makes sense that “Neighborhood” would have some collinearity with other variables seeing as usually neighborhoods have houses that are in a given range of Pricing Values. So, if there are houses that seem to share common qualities, we would assume those houses to be priced around the same amount. But of course, in a neighborhood, there is also a possibility the range in prices between houses can vary greatly depending on how populous or big the Neighborhood may be.
- When running the Studentized Breusch-Pagan test, our respective p-value for each of the models is < 2.2e-16. This extremely small p-value provides evidence against Heteroscedasticity, meaning, the variance across variables is constant (again found in the Appendix under “Checking Assumptions!”.

Comparing Competing Models

Predictive Models	Adjusted R2	CV PRESS	Kaggle Score
Forward	0.9298	0.1805	0.14538
Backward	0.9302	0.1799	0.14745
Stepwise	0.9148	0.2004	0.14954
CUSTOM (using Forward model again)	0.9298	0.1805	0.14538

Conclusion

In this analysis, we explored various linear regression models to predict housing prices based on a set of predictor variables. We began by preprocessing the data, including handling missing values and converting categorical variables into factors.

We then experimented with different variable selection techniques, including forward selection, backward elimination, and stepwise selection. These techniques helped us identify a subset of predictor variables that showed the strongest associations with the target variable, Log_SalePrice.

After selecting our models, we assessed their performance using cross-validation and diagnostic checks to ensure they met the assumptions of linear regression. The models demonstrated good predictive performance, with low cross-validated prediction errors and no significant violations of regression assumptions such as multicollinearity and heteroscedasticity.

We also compared the performance of our models on a test dataset and found that the forward-selected model consistently outperformed the others in terms of predictive accuracy. Additionally, we explored the possibility of creating a custom linear model based on our domain knowledge. However, our initial custom model did not outperform the forward-selected model, leading us to settle on the variables chosen through forward selection which were far and wide the most effective predictors of housing prices in this dataset.

The analysis demonstrates that the forward/custom and backward selection methods produced the most predictive models as both models outperformed the stepwise selection approach. Finally, the forward and backward selection methods offer reliable yet simple approaches for building predictive models of housing prices in Ames, Iowa based on the dataset provided.

Appendix

Analysis 1 R- Code

Install and load necessary packages

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##    filter, lag
```

```
## The following objects are masked from 'package:base':
##
##    intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(car)
```

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

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

```
## The following object is masked from 'package:dplyr':
##
##    recode
```

```
library(magrittr)
library(readr)
library(MASS)
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##    select
```

```
library(tidyverse)
```

```
## ── Attaching core tidyverse packages ──────────────────────────────────
tidyverse 2.0.0 ──
## ✔ forcats   1.0.0   ✔ stringr   1.5.1
## ✔ lubridate 1.9.2   ✔ tibble    3.2.1
## ✔ purrr 1.0.1       ✔ tidyr     1.3.1

## ── Conflicts ────────────────────────────────────────────────────────────
tidyverse_conflicts() ──
## ✖ tidyr::extract()   masks magrittr::extract()
## ✖ dplyr::filter()    masks stats::filter()
## ✖ dplyr::lag()       masks stats::lag()
## ✖ car::recode()      masks dplyr::recode()
## ✖ MASS::select()     masks dplyr::select()
## ✖ purrr::set_names() masks magrittr::set_names()
## ✖ purrr::some()      masks car::some()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all
conflicts to become errors
```

Load the data

```
data <- read_csv("house-prices-advanced-regression-techniques/train.csv")

## Rows: 1460 Columns: 81
## ── Column specification
──────────────────────────────────────────────────────────────────────────
## Delimiter: ","
## chr (43): MSZoning, Street, Alley, LotShape, LandContour, Utilities,
LotConf...
## dbl (38): Id, MSSubClass, LotFrontage, LotArea, OverallQual, OverallCond,
Ye...
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this
message.
```

Filter the data for specific neighborhoods & omit NA

```
filtered_data <- data %>%
  filter(Neighborhood %in% c("NAmes", "Edwards", "BrkSide")) %>%
  drop_na(SalePrice, GrLivArea, Neighborhood)
```

Summary statistics

```
summary_stats <- summary(filtered_data[c("SalePrice", "GrLivArea")])
summary_stats
```

```
##    SalePrice         GrLivArea
##  Min.   : 39300   Min.   : 334
##  1st Qu.:116000   1st Qu.:1003
##  Median :135500   Median :1200
##  Mean   :138062   Mean   :1302
##  3rd Qu.:155000   3rd Qu.:1496
##  Max.   :345000   Max.   :5642
```

Plotting distributions

Figure 1

```
# Histogram of Sale Prices
ggplot(filtered_data, aes(x = SalePrice)) + geom_histogram(bins = 30,fill =
"blue",alpha = 0.7) +
  ggtitle("Distribution of Sale Prices")

# Histogram of Living Area
ggplot(filtered_data, aes(x = GrLivArea)) + geom_histogram(bins = 30, fill =
"red",alpha = 0.7) +
  ggtitle("Distribution of Living Area")
```

Figure 2

```
# SalePrice vs GrLivArea scatterplot by Neighborhood
ggplot(filtered_data, aes(x = GrLivArea, y = SalePrice)) +
geom_point() +
facet_wrap(~ Neighborhood) +
labs(title = "SalePrice vs GrLivArea in Selected Neighborhoods",
x = "Living Area (GrLivArea)", y = "Sale Price")
```

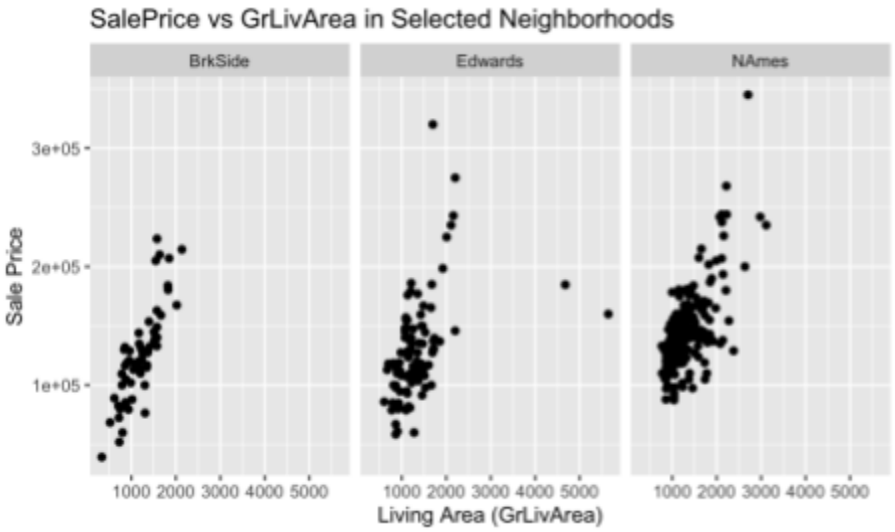


Figure 3

```
# SalePrice vs GrLivArea Scatterplot
ggplot(filtered_data, aes(x = GrLivArea, y = SalePrice, color =
Neighborhood)) +
geom_point() +
labs(title = "SalePrice vs GrLivArea in Selected Neighborhoods",
```

```
x = "Living Area", y = "Sale Price") +
theme_minimal()
```

Figure 4

Fit the SLR model

```
model <-
  lm(SalePrice ~ GrLivArea * Neighborhood, data = filtered_data)

# Summary of the model
modelsum <- summary(model)
modelsum

##
## Call:
## lm(formula = SalePrice ~ GrLivArea * Neighborhood, data = filtered_data)
##
## Residuals:
##    Min   1Q Median   3Q    Max
## -96204 -14568   -310  12601 181131
##
## Coefficients:
##                                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   19971.514  12351.125   1.617  0.10672
## GrLivArea                        87.163      9.782   8.911  < 2e-16 ***
## NeighborhoodEdwards           68381.591  13969.511   4.895 1.46e-06 ***
## NeighborhoodNAmes             54704.888  13882.334   3.941 9.69e-05 ***
## GrLivArea:NeighborhoodEdwards   -57.412     10.718  -5.357 1.48e-07 ***
## GrLivArea:NeighborhoodNAmes     -32.847     10.815  -3.037  0.00256 **
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 28550 on 377 degrees of freedom
## Multiple R-squared:  0.4474, Adjusted R-squared:   0.44
## F-statistic: 61.04 on 5 and 377 DF,  p-value: < 2.2e-16
```

```r
# Diagnostic plots
par(mfrow = c(1, 1))
plot(model)
```

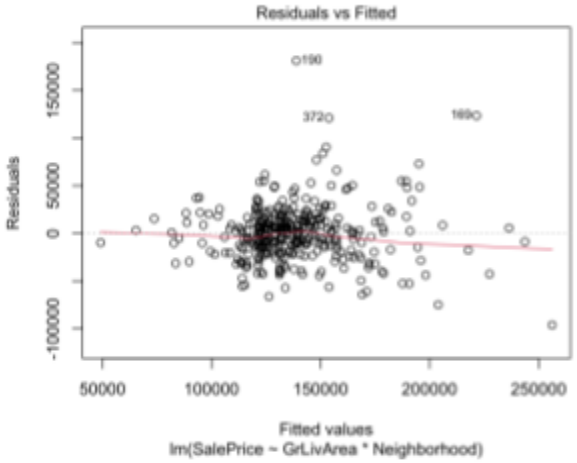


Figure 5

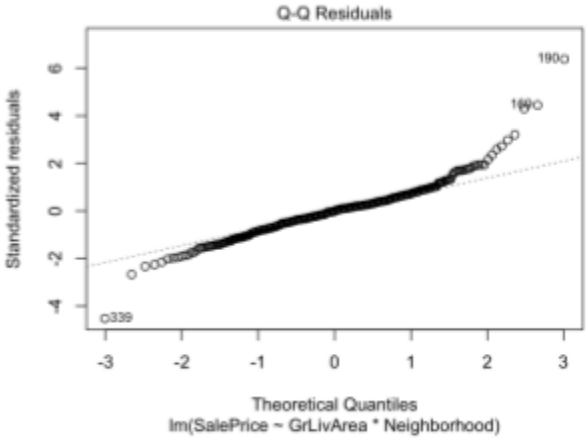


Figure 6

Influence measures

```r
influence_measures <- cooks.distance(model)

# Plot Cook's distance
plot(influence_measures,
      type = "h",
      main = "Cook's Distance",
      ylab = "Cook's Distance")
```

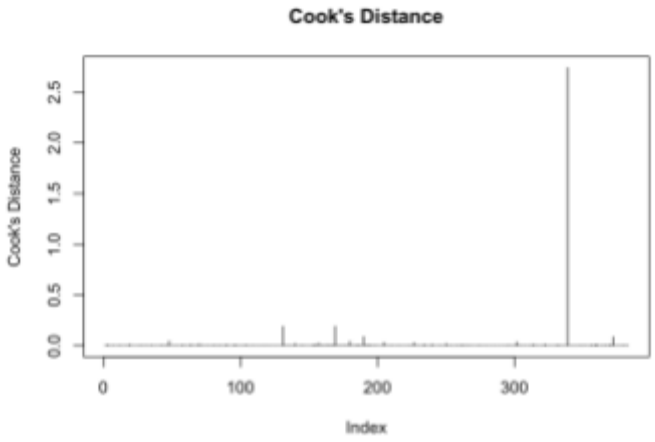


Figure 7

Identify high leverage points

```
influential_points <-
  which(influence_measures > (4/383))
influential_points
```

```
##  19  48  64  70  90 131 140 157 164 169 180 190 205 227 234 240 250 302
314 322
##  19  48  64  70  90 131 140 157 164 169 180 190 205 227 234 240 250 302
314 322
## 339 360 372
## 339 360 372
```

SLR plot with influential points removed

```
# Remove influential points
refined_data <- filtered_data[-influential_points, ]

# New model with high leverage points removed
model2 <-
  lm(SalePrice ~ GrLivArea * Neighborhood, data = refined_data)
summodel2 <- summary(model2)

# Summary of model
summodel2
```

```
##
## Call:
```

```
## lm(formula = SalePrice ~ GrLivArea * Neighborhood, data = refined_data)
##
## Residuals:
##    Min   1Q Median   3Q    Max
## -62030 -13040  981  13115  66684
##
## Coefficients:
##                                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   24290.931   9651.230   2.517 0.012281 *
## GrLivArea                        82.187      7.898  10.405  < 2e-16 ***
## NeighborhoodEdwards           43625.330  13320.955   3.275 0.001161 **
## NeighborhoodNAmes             58599.377  10976.283   5.339 1.68e-07 ***
## GrLivArea:NeighborhoodEdwards   -39.778     10.785  -3.688 0.000261 ***
## GrLivArea:NeighborhoodNAmes     -35.006      8.836  -3.962 9.00e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 21370 on 354 degrees of freedom
## Multiple R-squared:  0.5209, Adjusted R-squared:  0.5141
## F-statistic: 76.97 on 5 and 354 DF,  p-value: < 2.2e-16
```

```
#Plot of model
plot(model2)
```

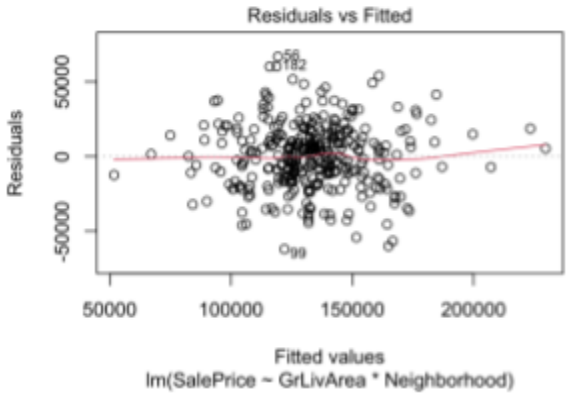


Figure 8

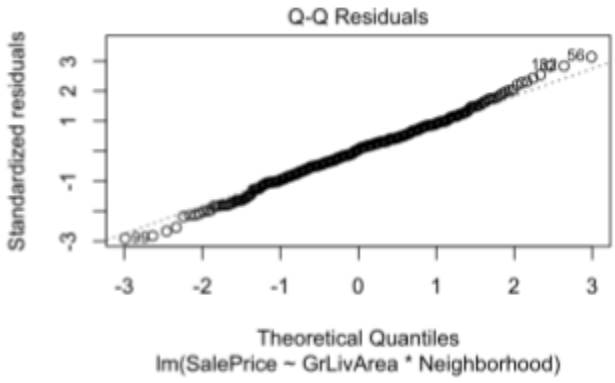


Figure 9

Log transformation of Variables

```
refined_data$LogSalePrice <- log(refined_data$SalePrice)
refined_data$LogGrLivArea <- log(refined_data$GrLivArea)
```

Model with LogSalePrice only

```
logSP_model <-
  lm(LogSalePrice ~ GrLivArea * Neighborhood, data = refined_data)

# Summary of model
sumlogSP_model <- summary(logSP_model)
sumlogSP_model

##
## Call:
## lm(formula = LogSalePrice ~ GrLivArea * Neighborhood, data = refined_data)
##
## Residuals:
##     Min          1Q   Median       3Q     Max
## -0.68104 -0.09390  0.01498  0.11110  0.47484
##
## Coefficients:
##                                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)                      1.080e+01  7.877e-02 137.165  < 2e-16 ***
## GrLivArea                        7.253e-04  6.446e-05  11.252  < 2e-16 ***
## NeighborhoodEdwards              3.914e-01  1.087e-01   3.600 0.000364 ***
## NeighborhoodNAmes                6.632e-01  8.958e-02   7.403 9.77e-13 ***
## GrLivArea:NeighborhoodEdwards -3.429e-04  8.802e-05  -3.895 0.000117 ***
## GrLivArea:NeighborhoodNAmes   -4.218e-04  7.211e-05  -5.849 1.13e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1744 on 354 degrees of freedom
## Multiple R-squared:  0.5082, Adjusted R-squared:  0.5012
## F-statistic: 73.15 on 5 and 354 DF,  p-value: < 2.2e-16

# Plot of model
plot(logSP_model)
```

Model with LogLivArea only

```
logLA_model <-
  lm(SalePrice ~ LogGrLivArea * Neighborhood, data = refined_data)

#Summary of model
sumlogLA_model <- summary(logLA_model)
sumlogLA_model
```

```
##
## Call:
## lm(formula = SalePrice ~ LogGrLivArea * Neighborhood, data = refined_data)
##
## Residuals:
##    Min   1Q Median   3Q    Max
## -63722 -12519  908  13151  65018
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     -472619     59503  -7.943 2.65e-14 ***
## LogGrLivArea                    84613       8486   9.971  < 2e-16 ***
## NeighborhoodEdwards             215608      86789   2.484   0.0134 *
## NeighborhoodNAmes               144273      71922   2.006   0.0456 *
## LogGrLivArea:NeighborhoodEdwards   -31375    12320  -2.547   0.0113 *
## LogGrLivArea:NeighborhoodNAmes   -18354      10211  -1.798   0.0731 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 21580 on 354 degrees of freedom
## Multiple R-squared:  0.5114, Adjusted R-squared:  0.5045
## F-statistic:  74.1 on 5 and 354 DF,  p-value: < 2.2e-16
```

```
# Plot of model
plot(logLA_model)
```

Model with both log

```
logboth_model <-
  lm(LogSalePrice ~ LogGrLivArea * Neighborhood, data = refined_data)

# Summary of model
sumlogboth_model <- summary(logboth_model)
sumlogboth_model
```

```
##
## Call:
## lm(formula = LogSalePrice ~ LogGrLivArea * Neighborhood, data =
refined_data)
##
## Residuals:
##    Min         1Q   Median       3Q    Max
## -0.69620 -0.09036  0.02169  0.10294  0.45985
##
```

```
## Coefficients:
##                                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)                      6.09842    0.47243  12.909  < 2e-16 ***
## LogGrLivArea                     0.79242    0.06737  11.762  < 2e-16 ***
## NeighborhoodEdwards              2.17461    0.68907   3.156  0.00174 **
## NeighborhoodNAmes                2.65458    0.57104   4.649 4.72e-06 ***
## LogGrLivArea:NeighborhoodEdwards -0.31346    0.09781  -3.205  0.00148 **
## LogGrLivArea:NeighborhoodNAmes   -0.35654    0.08107  -4.398 1.45e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1713 on 354 degrees of freedom
## Multiple R-squared:  0.5252, Adjusted R-squared:  0.5185
## F-statistic: 78.33 on 5 and 354 DF,  p-value: < 2.2e-16
```

```r
# Plot of model
plot(logboth_model)
```

Internal CV Press

```r
# Function to calculate PRESS
calculate_press <- function(model, data) {
  n <- nrow(data)
  press <- 0

  for (i in 1:n) {
      # Fit model without the ith observation
      model_loo <- update(model, subset = -i)

      # Predict the ith observation
      pred <- predict(model_loo, data[i, , drop = FALSE])

      # Calculate squared prediction error and add to PRESS
      press <- press + (data$LogSalePrice[i] - pred)^2
  }

  return(press)
}

# Calculate PRESS statistics
press_original <- calculate_press(model, refined_data)
press_logSP <- calculate_press(logSP_model, refined_data)
press_logLA <- calculate_press(logLA_model, refined_data)
press_bothlog <- calculate_press(logboth_model, refined_data)
```

Compare Adjusted R- squared and Internal CV Press for models

```
cat("Adjusted R-squared for the original model:", modelsum$adj.r.squared,
"\n")
```

```
## Adjusted R-squared for the original model: 0.4400466
```

```
cat("Adjusted R-squared for the LogSalePrice model:",
sumlogSP_model$adj.r.squared, "\n")
```

```
## Adjusted R-squared for the LogSalePrice model: 0.5012172
```

```
cat("Adjusted R-squared for the LogLivingArea model:",
sumlogLA_model$adj.r.squared, "\n")
```

```
## Adjusted R-squared for the LogLivingArea model: 0.5044703
```

```
cat("Adjusted R-squared for the both log model:",
sumlogboth_model$adj.r.squared, "\n")
```

```
## Adjusted R-squared for the both log model: 0.5185443
```

```
cat("PRESS for original model:", press_original, "\n")
```

```
## PRESS for original model: 6.782684e+12
```

```
cat("PRESS for LogSalePrice model:", press_logSP, "\n")
```

```
## PRESS for LogSalePrice model: 11.17583
```

```
cat("PRESS for LogLivingArea model:", press_logLA, "\n")
```

```
## PRESS for LogLivingArea model: 6.573203e+12
```

```
cat("PRESS for log both model:", press_bothlog, "\n")
```

```
## PRESS for log both model: 10.76139
```

Parameters

```
sumlogboth_model
```

```
##
## Call:
## lm(formula = LogSalePrice ~ LogGrLivArea * Neighborhood, data =
refined_data)
##
## Residuals:
##     Min       1Q   Median       3Q      Max
```

```
## -0.69620 -0.09036  0.02169  0.10294  0.45985
##
## Coefficients:
##                                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)                      6.09842    0.47243  12.909  < 2e-16 ***
## LogGrLivArea                     0.79242    0.06737  11.762  < 2e-16 ***
## NeighborhoodEdwards              2.17461    0.68907   3.156  0.00174 **
## NeighborhoodNAmes                2.65458    0.57104   4.649 4.72e-06 ***
## LogGrLivArea:NeighborhoodEdwards -0.31346    0.09781  -3.205  0.00148 **
## LogGrLivArea:NeighborhoodNAmes   -0.35654    0.08107  -4.398 1.45e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1713 on 354 degrees of freedom
## Multiple R-squared:  0.5252, Adjusted R-squared:  0.5185
## F-statistic: 78.33 on 5 and 354 DF,  p-value: < 2.2e-16
```

```
model_CI <- confint(logboth_model, level = 0.95)
model_CI
```

```
##                                      2.5 %       97.5 %
## (Intercept)                      5.1692864  7.0275447
## LogGrLivArea                     0.6599205  0.9249283
## NeighborhoodEdwards              0.8194168  3.5298045
## NeighborhoodNAmes                1.5315274  3.7776375
## LogGrLivArea:NeighborhoodEdwards -0.5058356 -0.1210931
## LogGrLivArea:NeighborhoodNAmes   -0.5159846 -0.1971023
```

Shiny App Code

```
library(shiny)
library(ggplot2)
library(rsconnect)

##
## Attaching package: 'rsconnect'

## The following object is masked from 'package:shiny':
##
##    serverInfo

library(readr)
library(magrittr)
library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##    filter, lag

## The following objects are masked from 'package:base':
##
##    intersect, setdiff, setequal, union

data <- read_csv("train.csv")

## Rows: 1460 Columns: 81

## ── Column specification
────────────────────────────────────────────────────────
## Delimiter: ","
## chr (43): MSZoning, Street, Alley, LotShape, LandContour, Utilities,
LotConf...
## dbl (38): Id, MSSubClass, LotFrontage, LotArea, OverallQual, OverallCond,
Ye...
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this
message.
```

```r
# UI
ui <- fluidPage(
  titlePanel("Interactive Price vs. Living Area Chart"),
  sidebarLayout(
      sidebarPanel(
      helpText("Interactive chart displaying the relationship between sale
price and living area."),
      # Checkbox group for neighborhoods
      checkboxGroupInput(
      "neighborhood",
      "Neighborhood:",
      choices = unique(data$Neighborhood),
      selected = unique(data$Neighborhood)
      ),
      # Options for log transformation
      checkboxInput("logScale", "Log-transform Sale Price", value = FALSE),
      checkboxInput("logGrLivArea", "Log-transform Living Area", value =
FALSE)
      ),
      mainPanel(
      plotOutput("priceLivingAreaPlot")
      )
  )
)

# Server
server <- function(input, output) {
  filtered_data <- reactive({
      if (is.null(input$neighborhood) || identical(input$neighborhood, "")) {
      dat <- data
      } else {
      dat <- data %>% filter(Neighborhood %in% input$neighborhood)
      }
      dat
  })

  # Render the plot
  output$priceLivingAreaPlot <- renderPlot({
      plot_data <- filtered_data()

      # Apply log transformations if selected
      if (input$logScale) {
      plot_data$SalePrice <- log(plot_data$SalePrice)
      }
      if (input$logGrLivArea) {
      plot_data$GrLivArea <- log(plot_data$GrLivArea)
      }

      # Generate the plot
      ggplot(plot_data, aes(x = GrLivArea, y = SalePrice)) +
```

```
      geom_point(alpha = 0.5) +
      labs(
      x = ifelse(input$logGrLivArea, "Log of Living Area (sq ft)", "Living
Area (sq ft)"),
      y = ifelse(input$logScale, "Log of Sale Price ($)", "Sale Price ($)"),
      title = "Sale Price vs. Living Area"
      ) +
      theme_minimal()
  })
}

# Run the app
shinyApp(ui = ui, server = server)
```

Analysis 2 R-Code

Loading Packages

```
library(MASS)

## Warning: package 'MASS' was built under R version 4.3.3

library(glmnet)

## Warning: package 'glmnet' was built under R version 4.3.3

## Loading required package: Matrix

## Loaded glmnet 4.1-8

library(ggplot2)
library(leaps)

## Warning: package 'leaps' was built under R version 4.3.3

library(olsrr)

## Warning: package 'olsrr' was built under R version 4.3.3

##
## Attaching package: 'olsrr'

## The following object is masked from 'package:MASS':
##
##     cement

## The following object is masked from 'package:datasets':
##
##     rivers

library(plyr)
library(forecast)
```

```
## Warning: package 'forecast' was built under R version 4.3.3

## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo

library(caret)

## Loading required package: lattice

library(car)

## Loading required package: carData

library(lmtest)

## Warning: package 'lmtest' was built under R version 4.3.3

## Loading required package: zoo

## Warning: package 'zoo' was built under R version 4.3.3

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

Loading Data

```
train_df = read.csv(choose.files(), header = TRUE)
test_df = read.csv(choose.files(), header = TRUE)

View(train_df)
View(test_df)
names(train_df)

##  [1] "Id"            "MSSubClass"    "MSZoning"      "LotFrontage"
##  [5] "LotArea"       "Street"        "Alley"         "LotShape"
##  [9] "LandContour"   "Utilities"     "LotConfig"     "LandSlope"
## [13] "Neighborhood"  "Condition1"    "Condition2"    "BldgType"
## [17] "HouseStyle"    "OverallQual"   "OverallCond"   "YearBuilt"
## [21] "YearRemodAdd"  "RoofStyle"     "RoofMatl"      "Exterior1st"
## [25] "Exterior2nd"   "MasVnrType"    "MasVnrArea"    "ExterQual"
## [29] "ExterCond"     "Foundation"    "BsmtQual"      "BsmtCond"
## [33] "BsmtExposure"  "BsmtFinType1"  "BsmtFinSF1"    "BsmtFinType2"
## [37] "BsmtFinSF2"    "BsmtUnfSF"     "TotalBsmtSF"   "Heating"
## [41] "HeatingQC"     "CentralAir"    "Electrical"    "X1stFlrSF"
## [45] "X2ndFlrSF"     "LowQualFinSF"  "GrLivArea"     "BsmtFullBath"
## [49] "BsmtHalfBath"  "FullBath"      "HalfBath"      "BedroomAbvGr"
## [53] "KitchenAbvGr"  "KitchenQual"   "TotRmsAbvGrd"  "Functional"
## [57] "Fireplaces"    "FireplaceQu"   "GarageType"    "GarageYrBlt"
```

```
## [61] "GarageFinish"  "GarageCars"    "GarageArea"    "GarageQual"
## [65] "GarageCond"    "PavedDrive"    "WoodDeckSF"    "OpenPorchSF"
## [69] "EnclosedPorch" "X3SsnPorch"    "ScreenPorch"   "PoolArea"
## [73] "PoolQC"        "Fence"         "MiscFeature"   "MiscVal"
## [77] "MoSold"        "YrSold"        "SaleType"      "SaleCondition"
## [81] "SalePrice"
```

```
str(train_df)
```

```
## 'data.frame':    1460 obs. of  81 variables:
##  $ Id           : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ MSSubClass   : int  60 20 60 70 60 50 20 60 50 190 ...
##  $ MSZoning     : chr  "RL" "RL" "RL" "RL" ...
##  $ LotFrontage  : int  65 80 68 60 84 85 75 NA 51 50 ...
##  $ LotArea      : int  8450 9600 11250 9550 14260 14115 10084 10382 6120
7420 ...
##  $ Street       : chr  "Pave" "Pave" "Pave" "Pave" ...
##  $ Alley        : chr  NA NA NA NA ...
##  $ LotShape     : chr  "Reg" "Reg" "IR1" "IR1" ...
##  $ LandContour  : chr  "Lvl" "Lvl" "Lvl" "Lvl" ...
##  $ Utilities    : chr  "AllPub" "AllPub" "AllPub" "AllPub" ...
##  $ LotConfig    : chr  "Inside" "FR2" "Inside" "Corner" ...
##  $ LandSlope    : chr  "Gtl" "Gtl" "Gtl" "Gtl" ...
##  $ Neighborhood : chr  "CollgCr" "Veenker" "CollgCr" "Crawfor" ...
##  $ Condition1   : chr  "Norm" "Feedr" "Norm" "Norm" ...
##  $ Condition2   : chr  "Norm" "Norm" "Norm" "Norm" ...
##  $ BldgType     : chr  "1Fam" "1Fam" "1Fam" "1Fam" ...
##  $ HouseStyle   : chr  "2Story" "1Story" "2Story" "2Story" ...
##  $ OverallQual  : int  7 6 7 7 8 5 8 7 7 5 ...
##  $ OverallCond  : int  5 8 5 5 5 5 5 6 5 6 ...
##  $ YearBuilt    : int  2003 1976 2001 1915 2000 1993 2004 1973 1931 1939
...
##  $ YearRemodAdd : int  2003 1976 2002 1970 2000 1995 2005 1973 1950 1950
...
##  $ RoofStyle    : chr  "Gable" "Gable" "Gable" "Gable" ...
##  $ RoofMatl     : chr  "CompShg" "CompShg" "CompShg" "CompShg" ...
##  $ Exterior1st  : chr  "VinylSd" "MetalSd" "VinylSd" "Wd Sdng" ...
##  $ Exterior2nd  : chr  "VinylSd" "MetalSd" "VinylSd" "Wd Shng" ...
##  $ MasVnrType   : chr  "BrkFace" "None" "BrkFace" "None" ...
##  $ MasVnrArea   : int  196 0 162 0 350 0 186 240 0 0 ...
##  $ ExterQual    : chr  "Gd" "TA" "Gd" "TA" ...
##  $ ExterCond    : chr  "TA" "TA" "TA" "TA" ...
##  $ Foundation   : chr  "PConc" "CBlock" "PConc" "BrkTil" ...
##  $ BsmtQual     : chr  "Gd" "Gd" "Gd" "TA" ...
##  $ BsmtCond     : chr  "TA" "TA" "TA" "Gd" ...
##  $ BsmtExposure : chr  "No" "Gd" "Mn" "No" ...
##  $ BsmtFinType1 : chr  "GLQ" "ALQ" "GLQ" "ALQ" ...
##  $ BsmtFinSF1   : int  706 978 486 216 655 732 1369 859 0 851 ...
##  $ BsmtFinType2 : chr  "Unf" "Unf" "Unf" "Unf" ...
##  $ BsmtFinSF2   : int  0 0 0 0 0 0 0 32 0 0 ...
```

```
##  $ BsmtUnfSF    : int  150 284 434 540 490 64 317 216 952 140 ...
##  $ TotalBsmtSF  : int  856 1262 920 756 1145 796 1686 1107 952 991 ...
##  $ Heating      : chr  "GasA" "GasA" "GasA" "GasA" ...
##  $ HeatingQC    : chr  "Ex" "Ex" "Ex" "Gd" ...
##  $ CentralAir   : chr  "Y" "Y" "Y" "Y" ...
##  $ Electrical   : chr  "SBrkr" "SBrkr" "SBrkr" "SBrkr" ...
##  $ X1stFlrSF    : int  856 1262 920 961 1145 796 1694 1107 1022 1077 ...
##  $ X2ndFlrSF    : int  854 0 866 756 1053 566 0 983 752 0 ...
##  $ LowQualFinSF : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ GrLivArea    : int  1710 1262 1786 1717 2198 1362 1694 2090 1774 1077
...
##  $ BsmtFullBath : int  1 0 1 1 1 1 1 1 0 1 ...
##  $ BsmtHalfBath : int  0 1 0 0 0 0 0 0 0 0 ...
##  $ FullBath     : int  2 2 2 1 2 1 2 2 2 1 ...
##  $ HalfBath     : int  1 0 1 0 1 1 0 1 0 0 ...
##  $ BedroomAbvGr : int  3 3 3 3 4 1 3 3 2 2 ...
##  $ KitchenAbvGr : int  1 1 1 1 1 1 1 1 2 2 ...
##  $ KitchenQual  : chr  "Gd" "TA" "Gd" "Gd" ...
##  $ TotRmsAbvGrd : int  8 6 6 7 9 5 7 7 8 5 ...
##  $ Functional   : chr  "Typ" "Typ" "Typ" "Typ" ...
##  $ Fireplaces   : int  0 1 1 1 1 0 1 2 2 2 ...
##  $ FireplaceQu  : chr  NA "TA" "TA" "Gd" ...
##  $ GarageType   : chr  "Attchd" "Attchd" "Attchd" "Detchd" ...
##  $ GarageYrBlt  : int  2003 1976 2001 1998 2000 1993 2004 1973 1931 1939
...
##  $ GarageFinish : chr  "RFn" "RFn" "RFn" "Unf" ...
##  $ GarageCars   : int  2 2 2 3 3 2 2 2 2 1 ...
##  $ GarageArea   : int  548 460 608 642 836 480 636 484 468 205 ...
##  $ GarageQual   : chr  "TA" "TA" "TA" "TA" ...
##  $ GarageCond   : chr  "TA" "TA" "TA" "TA" ...
##  $ PavedDrive   : chr  "Y" "Y" "Y" "Y" ...
##  $ WoodDeckSF   : int  0 298 0 0 192 40 255 235 90 0 ...
##  $ OpenPorchSF  : int  61 0 42 35 84 30 57 204 0 4 ...
##  $ EnclosedPorch: int  0 0 0 272 0 0 0 228 205 0 ...
##  $ X3SsnPorch   : int  0 0 0 0 0 320 0 0 0 0 ...
##  $ ScreenPorch  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ PoolArea     : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ PoolQC       : chr  NA NA NA NA ...
##  $ Fence        : chr  NA NA NA NA ...
##  $ MiscFeature  : chr  NA NA NA NA ...
##  $ MiscVal      : int  0 0 0 0 0 700 0 350 0 0 ...
##  $ MoSold       : int  2 5 9 2 12 10 8 11 4 1 ...
##  $ YrSold       : int  2008 2007 2008 2006 2008 2009 2007 2009 2008 2008
...
##  $ SaleType     : chr  "WD" "WD" "WD" "WD" ...
##  $ SaleCondition: chr  "Normal" "Normal" "Normal" "Abnorml" ...
##  $ SalePrice    : int  208500 181500 223500 140000 250000 143000 307000
200000 129900 118000 ...
```

Data Processing and Cleaning

First we will do a count of NA values per column.

```
## Count the number of NA values in each column
na_summary <- colSums(is.na(train_df))
sum(na_summary > 0) # 19 columns w NA. I would like to filter them out to
ease

## [1] 19

#choosing variables
# Filter out columns with NA values
na_summary <- na_summary[na_summary > 0]
# Print the summary, should be the names of the 19 and how many.
print(na_summary)

##  LotFrontage        Alley   MasVnrType   MasVnrArea     BsmtQual
BsmtCond
##          259         1369            8            8           37
37
## BsmtExposure BsmtFinType1 BsmtFinType2   Electrical  FireplaceQu
GarageType
##           38           37           38            1          690
81
##  GarageYrBlt GarageFinish   GarageQual   GarageCond       PoolQC
Fence
##           81           81           81           81         1453
1179
##  MiscFeature
##         1406
```

Here we will obtain the column names then check a summary of our clean datasets.

```
# Get the column names with NA values
na_cols <- names(na_summary)
# Create a new dataframe without the columns containing NA values
train_clean <- train_df[, !(names(train_df) %in% na_cols)]
test_clean <- test_df[, !(names(test_df) %in% na_cols)]
#Checking summary
summary(train_clean)

##        Id           MSSubClass      MSZoning           LotArea
##  Min.   :   1.0   Min.   : 20.0   Length:1460        Min.   :  1300
##  1st Qu.: 365.8   1st Qu.: 20.0   Class :character   1st Qu.:  7554
##  Median : 730.5   Median : 50.0   Mode  :character   Median :  9478
##  Mean   : 730.5   Mean   : 56.9                      Mean   : 10517
##  3rd Qu.:1095.2   3rd Qu.: 70.0                      3rd Qu.: 11602
##  Max.   :1460.0   Max.   :190.0                      Max.   :215245
##     Street            LotShape         LandContour         Utilities
##  Length:1460        Length:1460        Length:1460        Length:1460
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
```

```
##
##
##
##   LotConfig          LandSlope         Neighborhood        Condition1
##  Length:1460        Length:1460        Length:1460        Length:1460
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##   Condition2          BldgType          HouseStyle         OverallQual
##  Length:1460        Length:1460        Length:1460        Min.   : 1.000
##  Class :character   Class :character   Class :character   1st Qu.: 5.000
##  Mode  :character   Mode  :character   Mode  :character   Median : 6.000
##                                                           Mean   : 6.099
##                                                           3rd Qu.: 7.000
##                                                           Max.   :10.000
##   OverallCond      YearBuilt     YearRemodAdd   RoofStyle
##  Min.   :1.000   Min.   :1872   Min.   :1950   Length:1460
##  1st Qu.:5.000   1st Qu.:1954   1st Qu.:1967   Class :character
##  Median :5.000   Median :1973   Median :1994   Mode  :character
##  Mean   :5.575   Mean   :1971   Mean   :1985
##  3rd Qu.:6.000   3rd Qu.:2000   3rd Qu.:2004
##  Max.   :9.000   Max.   :2010   Max.   :2010
##    RoofMatl         Exterior1st        Exterior2nd         ExterQual
##  Length:1460        Length:1460        Length:1460        Length:1460
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##   ExterCond          Foundation          BsmtFinSF1       BsmtFinSF2
##  Length:1460        Length:1460        Min.   :   0.0   Min.   :   0.00
##  Class :character   Class :character   1st Qu.:   0.0   1st Qu.:   0.00
##  Mode  :character   Mode  :character   Median : 383.5   Median :   0.00
##                                        Mean   : 443.6   Mean   :  46.55
##                                        3rd Qu.: 712.2   3rd Qu.:   0.00
##                                        Max.   :5644.0   Max.   :1474.00
##    BsmtUnfSF       TotalBsmtSF       Heating           HeatingQC
##  Min.   :   0.0   Min.   :   0.0   Length:1460        Length:1460
##  1st Qu.: 223.0   1st Qu.: 795.8   Class :character   Class :character
##  Median : 477.5   Median : 991.5   Mode  :character   Mode  :character
##  Mean   : 567.2   Mean   :1057.4
##  3rd Qu.: 808.0   3rd Qu.:1298.2
##  Max.   :2336.0   Max.   :6110.0
##   CentralAir          X1stFlrSF      X2ndFlrSF     LowQualFinSF
##  Length:1460        Min.   : 334   Min.   :   0   Min.   :  0.000
##  Class :character   1st Qu.: 882   1st Qu.:   0   1st Qu.:  0.000
##  Mode  :character   Median :1087   Median :   0   Median :  0.000
##                     Mean   :1163   Mean   : 347   Mean   :  5.845
```

```
##                    3rd Qu.:1391   3rd Qu.: 728   3rd Qu.:  0.000
##                    Max.   :4692   Max.   :2065   Max.   :572.000
##    GrLivArea     BsmtFullBath     BsmtHalfBath        FullBath
##  Min.   : 334   Min.   :0.0000   Min.   :0.00000   Min.   :0.000
##  1st Qu.:1130   1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.:1.000
##  Median :1464   Median :0.0000   Median :0.00000   Median :2.000
##  Mean   :1515   Mean   :0.4253   Mean   :0.05753   Mean   :1.565
##  3rd Qu.:1777   3rd Qu.:1.0000   3rd Qu.:0.00000   3rd Qu.:2.000
##  Max.   :5642   Max.   :3.0000   Max.   :2.00000   Max.   :3.000
##     HalfBath       BedroomAbvGr    KitchenAbvGr   KitchenQual
##  Min.   :0.0000   Min.   :0.000   Min.   :0.000   Length:1460
##  1st Qu.:0.0000   1st Qu.:2.000   1st Qu.:1.000   Class :character
##  Median :0.0000   Median :3.000   Median :1.000   Mode  :character
##  Mean   :0.3829   Mean   :2.866   Mean   :1.047
##  3rd Qu.:1.0000   3rd Qu.:3.000   3rd Qu.:1.000
##  Max.   :2.0000   Max.   :8.000   Max.   :3.000
##   TotRmsAbvGrd     Functional          Fireplaces      GarageCars
##  Min.   : 2.000   Length:1460        Min.   :0.000   Min.   :0.000
##  1st Qu.: 5.000   Class :character   1st Qu.:0.000   1st Qu.:1.000
##  Median : 6.000   Mode  :character   Median :1.000   Median :2.000
##  Mean   : 6.518                      Mean   :0.613   Mean   :1.767
##  3rd Qu.: 7.000                      3rd Qu.:1.000   3rd Qu.:2.000
##  Max.   :14.000                      Max.   :3.000   Max.   :4.000
##    GarageArea      PavedDrive          WoodDeckSF      OpenPorchSF
##  Min.   :   0.0   Length:1460        Min.   :  0.00   Min.   :  0.00
##  1st Qu.: 334.5   Class :character   1st Qu.:  0.00   1st Qu.:  0.00
##  Median : 480.0   Mode  :character   Median :  0.00   Median : 25.00
##  Mean   : 473.0                      Mean   : 94.24   Mean   : 46.66
##  3rd Qu.: 576.0                      3rd Qu.:168.00   3rd Qu.: 68.00
##  Max.   :1418.0                      Max.   :857.00   Max.   :547.00
##  EnclosedPorch      X3SsnPorch      ScreenPorch        PoolArea
##  Min.   :  0.00   Min.   :  0.00   Min.   :  0.00   Min.   :  0.000
##  1st Qu.:  0.00   1st Qu.:  0.00   1st Qu.:  0.00   1st Qu.:  0.000
##  Median :  0.00   Median :  0.00   Median :  0.00   Median :  0.000
##  Mean   : 21.95   Mean   :  3.41   Mean   : 15.06   Mean   :  2.759
##  3rd Qu.:  0.00   3rd Qu.:  0.00   3rd Qu.:  0.00   3rd Qu.:  0.000
##  Max.   :552.00   Max.   :508.00   Max.   :480.00   Max.   :738.000
##     MiscVal             MoSold           YrSold       SaleType
##  Min.   :    0.00   Min.   : 1.000   Min.   :2006   Length:1460
##  1st Qu.:    0.00   1st Qu.: 5.000   1st Qu.:2007   Class :character
##  Median :    0.00   Median : 6.000   Median :2008   Mode  :character
##  Mean   :   43.49   Mean   : 6.322   Mean   :2008
##  3rd Qu.:    0.00   3rd Qu.: 8.000   3rd Qu.:2009
##  Max.   :15500.00   Max.   :12.000   Max.   :2010
##  SaleCondition        SalePrice
##  Length:1460        Min.   : 34900
##  Class :character   1st Qu.:129975
##  Mode  :character   Median :163000
##                     Mean   :180921
```

```
##                    3rd Qu.:214000
##                    Max.   :755000
```

```
summary(test_clean)
```

```
##        Id         MSSubClass       MSZoning            LotArea
##  Min.   :1461   Min.   : 20.00   Length:1459        Min.   : 1470
##  1st Qu.:1826   1st Qu.: 20.00   Class :character   1st Qu.: 7391
##  Median :2190   Median : 50.00   Mode  :character   Median : 9399
##  Mean   :2190   Mean   : 57.38                      Mean   : 9819
##  3rd Qu.:2554   3rd Qu.: 70.00                      3rd Qu.:11518
##  Max.   :2919   Max.   :190.00                      Max.   :56600
##
##     Street            LotShape         LandContour         Utilities
##  Length:1459        Length:1459        Length:1459        Length:1459
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##   LotConfig          LandSlope         Neighborhood        Condition1
##  Length:1459        Length:1459        Length:1459        Length:1459
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##   Condition2          BldgType          HouseStyle         OverallQual
##  Length:1459        Length:1459        Length:1459        Min.   : 1.000
##  Class :character   Class :character   Class :character   1st Qu.: 5.000
##  Mode  :character   Mode  :character   Mode  :character   Median : 6.000
##                                                           Mean   : 6.079
##                                                           3rd Qu.: 7.000
##                                                           Max.   :10.000
##
##   OverallCond      YearBuilt     YearRemodAdd   RoofStyle
##  Min.   :1.000   Min.   :1879   Min.   :1950   Length:1459
##  1st Qu.:5.000   1st Qu.:1953   1st Qu.:1963   Class :character
##  Median :5.000   Median :1973   Median :1992   Mode  :character
##  Mean   :5.554   Mean   :1971   Mean   :1984
##  3rd Qu.:6.000   3rd Qu.:2001   3rd Qu.:2004
##  Max.   :9.000   Max.   :2010   Max.   :2010
##
##    RoofMatl         Exterior1st        Exterior2nd         ExterQual
##  Length:1459        Length:1459        Length:1459        Length:1459
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
```

```
##
##
##
##   ExterCond          Foundation          BsmtFinSF1       BsmtFinSF2
##  Length:1459        Length:1459        Min.   :   0.0   Min.   :   0.00
##  Class :character   Class :character   1st Qu.:   0.0   1st Qu.:   0.00
##  Mode  :character   Mode  :character   Median : 350.5   Median :   0.00
##                                        Mean   : 439.2   Mean   :  52.62
##                                        3rd Qu.: 753.5   3rd Qu.:   0.00
##                                        Max.   :4010.0   Max.   :1526.00
##                                        NA's   :1        NA's   :1
##    BsmtUnfSF       TotalBsmtSF     Heating           HeatingQC
##  Min.   :   0.0   Min.   :   0   Length:1459        Length:1459
##  1st Qu.: 219.2   1st Qu.: 784   Class :character   Class :character
##  Median : 460.0   Median : 988   Mode  :character   Mode  :character
##  Mean   : 554.3   Mean   :1046
##  3rd Qu.: 797.8   3rd Qu.:1305
##  Max.   :2140.0   Max.   :5095
##  NA's   :1        NA's   :1
##   CentralAir          X1stFlrSF        X2ndFlrSF     LowQualFinSF
##  Length:1459        Min.   : 407.0   Min.   :   0   Min.   :   0.000
##  Class :character   1st Qu.: 873.5   1st Qu.:   0   1st Qu.:   0.000
##  Mode  :character   Median :1079.0   Median :   0   Median :   0.000
##                     Mean   :1156.5   Mean   : 326   Mean   :   3.543
##                     3rd Qu.:1382.5   3rd Qu.: 676   3rd Qu.:   0.000
##                     Max.   :5095.0   Max.   :1862   Max.   :1064.000
##
##    GrLivArea     BsmtFullBath     BsmtHalfBath       FullBath
##  Min.   : 407   Min.   :0.0000   Min.   :0.0000   Min.   :0.000
##  1st Qu.:1118   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:1.000
##  Median :1432   Median :0.0000   Median :0.0000   Median :2.000
##  Mean   :1486   Mean   :0.4345   Mean   :0.0652   Mean   :1.571
##  3rd Qu.:1721   3rd Qu.:1.0000   3rd Qu.:0.0000   3rd Qu.:2.000
##  Max.   :5095   Max.   :3.0000   Max.   :2.0000   Max.   :4.000
##                 NA's   :2        NA's   :2
##     HalfBath       BedroomAbvGr    KitchenAbvGr   KitchenQual
##  Min.   :0.0000   Min.   :0.000   Min.   :0.000   Length:1459
##  1st Qu.:0.0000   1st Qu.:2.000   1st Qu.:1.000   Class :character
##  Median :0.0000   Median :3.000   Median :1.000   Mode  :character
##  Mean   :0.3777   Mean   :2.854   Mean   :1.042
##  3rd Qu.:1.0000   3rd Qu.:3.000   3rd Qu.:1.000
##  Max.   :2.0000   Max.   :6.000   Max.   :2.000
##
##   TotRmsAbvGrd     Functional          Fireplaces       GarageCars
##  Min.   : 3.000   Length:1459        Min.   :0.0000   Min.   :0.000
##  1st Qu.: 5.000   Class :character   1st Qu.:0.0000   1st Qu.:1.000
##  Median : 6.000   Mode  :character   Median :0.0000   Median :2.000
##  Mean   : 6.385                      Mean   :0.5812   Mean   :1.766
##  3rd Qu.: 7.000                      3rd Qu.:1.0000   3rd Qu.:2.000
##  Max.   :15.000                      Max.   :4.0000   Max.   :5.000
```

```
##                                                       NA's   :1
##    GarageArea      PavedDrive          WoodDeckSF       OpenPorchSF
##  Min.   :   0.0   Length:1459        Min.   :   0.00   Min.   :  0.00
##  1st Qu.: 318.0   Class :character   1st Qu.:   0.00   1st Qu.:  0.00
##  Median : 480.0   Mode  :character   Median :   0.00   Median : 28.00
##  Mean   : 472.8                      Mean   :  93.17   Mean   : 48.31
##  3rd Qu.: 576.0                      3rd Qu.: 168.00   3rd Qu.: 72.00
##  Max.   :1488.0                      Max.   :1424.00   Max.   :742.00
##  NA's   :1
##  EnclosedPorch       X3SsnPorch       ScreenPorch        PoolArea
##  Min.   :   0.00   Min.   :  0.000   Min.   :  0.00   Min.   :  0.000
##  1st Qu.:   0.00   1st Qu.:  0.000   1st Qu.:  0.00   1st Qu.:  0.000
##  Median :   0.00   Median :  0.000   Median :  0.00   Median :  0.000
##  Mean   :  24.24   Mean   :  1.794   Mean   : 17.06   Mean   :  1.744
##  3rd Qu.:   0.00   3rd Qu.:  0.000   3rd Qu.:  0.00   3rd Qu.:  0.000
##  Max.   :1012.00   Max.   :360.000   Max.   :576.00   Max.   :800.000
##
##     MiscVal             MoSold          YrSold       SaleType
##  Min.   :    0.00   Min.   : 1.000   Min.   :2006   Length:1459
##  1st Qu.:    0.00   1st Qu.: 4.000   1st Qu.:2007   Class :character
##  Median :    0.00   Median : 6.000   Median :2008   Mode  :character
##  Mean   :   58.17   Mean   : 6.104   Mean   :2008
##  3rd Qu.:    0.00   3rd Qu.: 8.000   3rd Qu.:2009
##  Max.   :17000.00   Max.   :12.000   Max.   :2010
##
##  SaleCondition
##  Length:1459
##  Class :character
##  Mode  :character
##
##
##
##
```

From our summary we see that our character columns would make more sense if they were changed to factor values. After converting to characters, we will double check for NA's below.

```
# we see that there are many character columns that can be changed into a
factor
# of multiple levels
# Identify character columns
character_columns <- sapply(train_clean, is.character)
character_columns <- sapply(test_clean, is.character)
# Get the names of columns identified as character columns
character_column_names <- names(character_columns)[character_columns]
# Convert character columns to factors
train_clean[character_column_names] <-
lapply(train_clean[character_column_names], as.factor)
test_clean[character_column_names] <-
```

```
lapply(test_clean[character_column_names], as.factor)

# Double checking for NA's:
missing_values <- colSums(is.na(train_clean))
missing_val2 <- colSums(is.na(test_clean))
# Display variables with missing values and their counts
missing_values <- missing_values[missing_values > 0]
missing_val2 <- missing_val2[missing_val2 > 0]
print(missing_values) #There should be ZERO NA's.

## named numeric(0)

print(missing_val2) #There are NA's!! We can impute to deal w them.

##     MSZoning    Utilities  Exterior1st  Exterior2nd   BsmtFinSF1
BsmtFinSF2
##            4            2            1            1            1
1
##    BsmtUnfSF  TotalBsmtSF BsmtFullBath BsmtHalfBath  KitchenQual
Functional
##            1            1            2            2            1
2
##   GarageCars   GarageArea     SaleType
##            1            1            1
```

Now we see that there are no longer NAs in our Train_clean dataset, we want to eliminate NAs from the Test_clean dataset as well by imputing. For categorical variable columns we impute NAs along the mode, for numeric variable columns we impute along the mean.

```
#TO DEAL WITH MISSING VALUES WE WILL IMPUTE ALONG MEAN (NUMERIC)/MODE
(CATEGORICAL)
# Define a function to calculate the mode
Mode <- function(x) {
  ux <- unique(x)
  ux[which.max(tabulate(match(x, ux)))]
}

# Identify columns with missing values
missing_cols <- colnames(test_clean)[colSums(is.na(test_clean)) > 0]

# Impute categorical variables with mode and numerical variables with mean
for (col in missing_cols) {
  if (is.factor(test_clean[[col]])) {
    # Impute categorical variables with mode
    test_clean[[col]][is.na(test_clean[[col]])] <-
Mode(test_clean[[col]][!is.na(test_clean[[col]])])
  } else {
    # Impute numerical variables with mean
    test_clean[[col]][is.na(test_clean[[col]])] <- mean(test_clean[[col]],
na.rm = TRUE)
```

```
  }
}

# Verify if all missing values have been imputed
colSums(is.na(test_clean)) #NO MORE MISSING VALUES
```

```
##            Id    MSSubClass      MSZoning       LotArea        Street
##             0             0             0             0             0
##      LotShape   LandContour     Utilities     LotConfig     LandSlope
##             0             0             0             0             0
##  Neighborhood    Condition1    Condition2      BldgType    HouseStyle
##             0             0             0             0             0
##   OverallQual   OverallCond     YearBuilt  YearRemodAdd     RoofStyle
##             0             0             0             0             0
##      RoofMatl   Exterior1st   Exterior2nd     ExterQual     ExterCond
##             0             0             0             0             0
##    Foundation    BsmtFinSF1    BsmtFinSF2     BsmtUnfSF   TotalBsmtSF
##             0             0             0             0             0
##       Heating     HeatingQC    CentralAir     X1stFlrSF     X2ndFlrSF
##             0             0             0             0             0
##  LowQualFinSF     GrLivArea  BsmtFullBath  BsmtHalfBath      FullBath
##             0             0             0             0             0
##      HalfBath  BedroomAbvGr  KitchenAbvGr   KitchenQual  TotRmsAbvGrd
##             0             0             0             0             0
##    Functional    Fireplaces    GarageCars    GarageArea    PavedDrive
##             0             0             0             0             0
##    WoodDeckSF   OpenPorchSF EnclosedPorch    X3SsnPorch   ScreenPorch
##             0             0             0             0             0
##      PoolArea       MiscVal        MoSold        YrSold      SaleType
##             0             0             0             0             0
## SaleCondition
##             0
```

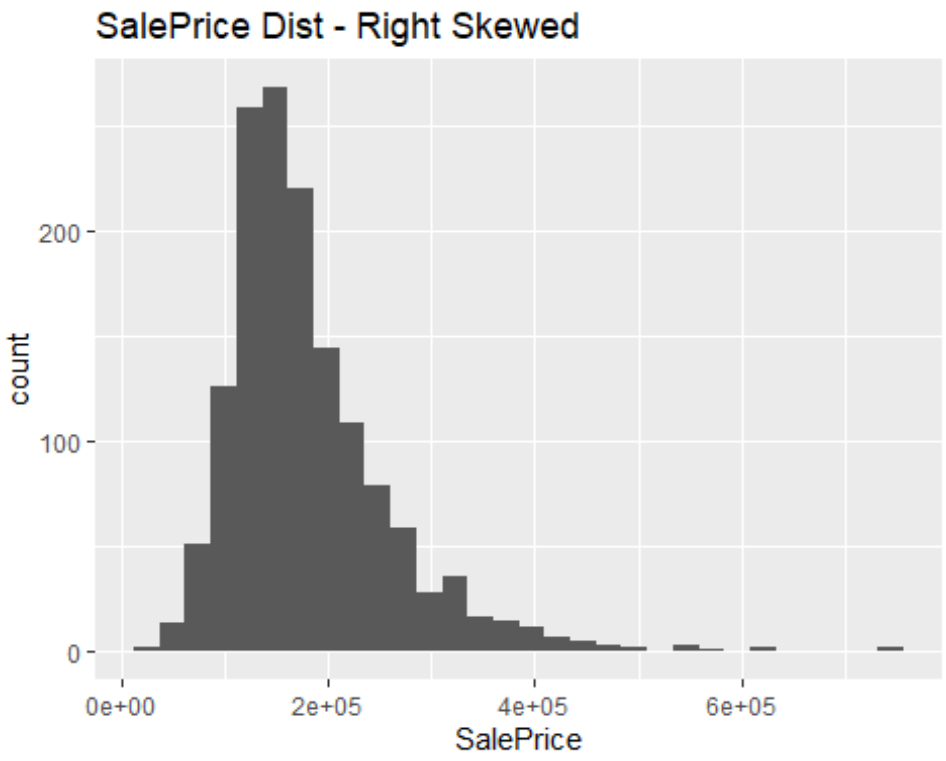
NO MORE MISSING VALUES!

Variable Analysis

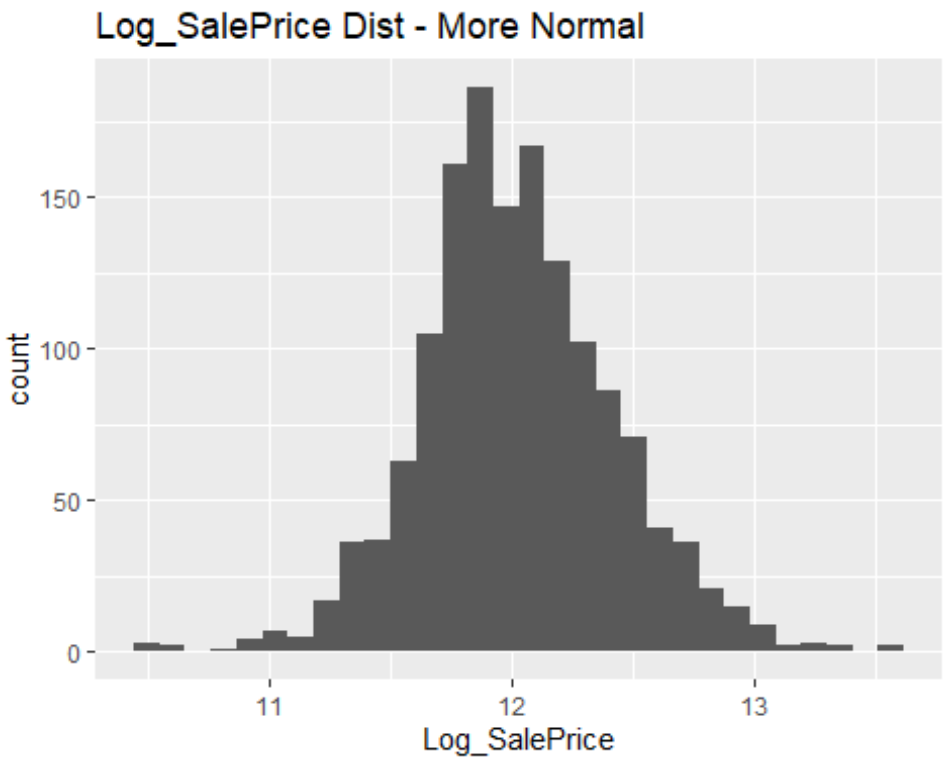
We will now analyze our response variable and check its distribution.

```
#Distribution of SalePrice is right-skewed:
ggplot(data = train_clean, aes(x = SalePrice)) +
  geom_histogram() +
  labs(title = "SalePrice Dist - Right Skewed")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
#Logging the SalePrice column because it is very right-skewed (non normal
distribution)
train_clean$Log_SalePrice = log(train_clean$SalePrice)

#Distribution of Log_SalePrice:
ggplot(data = train_clean, aes(x = Log_SalePrice)) +
  geom_histogram() +
  labs(title = "Log_SalePrice Dist - More Normal")

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Model Selection

Here we will do different selection techniques and store our models under different variable names (log_forward, log_backward, log_stepwise).

Forward Selection

```
#another way to do forward selection:
#int only model
log_intercept_only = lm(Log_SalePrice ~ 1, data = train_clean)
#model w all predictors
log_all = lm(Log_SalePrice ~.-SalePrice, data = train_clean)
#forward selection
log_forward = step(log_intercept_only, direction = "forward", scope =
formula(log_all), trace = 0)
log_forward # to show results

##
## Call:
## lm(formula = Log_SalePrice ~ OverallQual + Neighborhood + GrLivArea +
##     GarageCars + OverallCond + BsmtFullBath + RoofMatl + TotalBsmtSF +
##     YearBuilt + BldgType + Condition2 + MSZoning + BsmtFinSF1 +
##     SaleCondition + Functional + LotArea + CentralAir + KitchenQual +
##     Condition1 + Fireplaces + Heating + ScreenPorch + SaleType +
##     Exterior1st + WoodDeckSF + YearRemodAdd + GarageArea + Foundation +
##     LandSlope + EnclosedPorch + HeatingQC + LotConfig + BsmtFinSF2 +
##     Street + X3SsnPorch + KitchenAbvGr + PoolArea + HalfBath +
```

```
##     FullBath + X1stFlrSF + LandContour, data = train_clean)
##
## Coefficients:
##          (Intercept)           OverallQual   NeighborhoodBlueste
##            1.961e+00             4.974e-02            -6.829e-02
##   NeighborhoodBrDale   NeighborhoodBrkSide   NeighborhoodClearCr
##            -6.628e-02             6.636e-03             3.222e-02
##  NeighborhoodCollgCr   NeighborhoodCrawfor   NeighborhoodEdwards
##            -2.703e-02             9.132e-02            -7.714e-02
##  NeighborhoodGilbert    NeighborhoodIDOTRR   NeighborhoodMeadowV
##            -2.667e-02            -3.520e-02            -1.647e-01
##  NeighborhoodMitchel     NeighborhoodNAmes   NeighborhoodNoRidge
##            -6.428e-02            -4.014e-02             1.924e-02
##  NeighborhoodNPkVill   NeighborhoodNridgHt    NeighborhoodNWAmes
##             1.043e-03             7.529e-02            -4.823e-02
##  NeighborhoodOldTown    NeighborhoodSawyer   NeighborhoodSawyerW
##            -5.715e-02            -3.720e-02            -2.268e-02
##  NeighborhoodSomerst   NeighborhoodStoneBr     NeighborhoodSWISU
##             1.865e-02             1.016e-01            -9.875e-03
##   NeighborhoodTimber   NeighborhoodVeenker             GrLivArea
##            -8.342e-03             2.735e-02             2.366e-04
##           GarageCars           OverallCond          BsmtFullBath
##             2.931e-02             3.815e-02             2.596e-02
##      RoofMatlCompShg       RoofMatlMembran         RoofMatlMetal
##             2.618e+00             2.971e+00             2.788e+00
##         RoofMatlRoll       RoofMatlTar&Grv       RoofMatlWdShake
##             2.688e+00             2.678e+00             2.645e+00
##      RoofMatlWdShngl           TotalBsmtSF             YearBuilt
##             2.711e+00             7.237e-05             2.106e-03
##       BldgType2fmCon        BldgTypeDuplex         BldgTypeTwnhs
##            -3.720e-03            -1.452e-02            -1.070e-01
##       BldgTypeTwnhsE       Condition2Feedr        Condition2Norm
##            -5.916e-02             5.222e-02             2.052e-02
##       Condition2PosA        Condition2PosN        Condition2RRAe
##             3.176e-01            -8.534e-01            -7.408e-02
##       Condition2RRAn        Condition2RRNn            MSZoningFV
##            -6.901e-02            -4.985e-02             4.162e-01
##           MSZoningRH            MSZoningRL            MSZoningRM
##             3.981e-01             4.021e-01             3.674e-01
##           BsmtFinSF1  SaleConditionAdjLand   SaleConditionAlloca
##             7.546e-05             9.985e-02             6.605e-02
##  SaleConditionFamily   SaleConditionNormal  SaleConditionPartial
##             1.937e-02             7.330e-02            -3.962e-02
##       FunctionalMaj2        FunctionalMin1        FunctionalMin2
##            -2.196e-01             4.221e-02             3.785e-02
##        FunctionalMod         FunctionalSev         FunctionalTyp
##            -6.446e-02            -3.512e-01             7.919e-02
##              LotArea           CentralAirY         KitchenQualFa
##             2.434e-06             5.879e-02            -6.694e-02
##         KitchenQualGd         KitchenQualTA       Condition1Feedr
```

```
##           -6.691e-02           -6.559e-02            3.068e-02
##       Condition1Norm       Condition1PosA       Condition1PosN
##            7.961e-02            5.086e-02            7.437e-02
##       Condition1RRAe       Condition1RRAn       Condition1RRNe
##           -4.347e-02            4.936e-02            1.130e-02
##       Condition1RRNn           Fireplaces          HeatingGasA
##            9.622e-02            2.462e-02            1.405e-01
##          HeatingGasW          HeatingGrav          HeatingOthW
##            2.096e-01           -6.718e-03            1.003e-01
##          HeatingWall          ScreenPorch          SaleTypeCon
##            2.365e-01            2.618e-04            8.764e-02
##        SaleTypeConLD        SaleTypeConLI        SaleTypeConLw
##            1.370e-01           -2.739e-02            2.283e-02
##          SaleTypeCWD          SaleTypeNew          SaleTypeOth
##            9.768e-02            1.535e-01            7.648e-02
##           SaleTypeWD   Exterior1stAsphShn   Exterior1stBrkComm
##           -1.116e-02            4.498e-03           -1.900e-01
##   Exterior1stBrkFace    Exterior1stCBlock   Exterior1stCemntBd
##            8.811e-02           -1.500e-02            4.699e-02
##   Exterior1stHdBoard   Exterior1stImStucc   Exterior1stMetalSd
##            1.555e-02           -8.017e-03            4.586e-02
##   Exterior1stPlywood     Exterior1stStone    Exterior1stStucco
##            2.000e-02           -1.983e-02            2.643e-02
##   Exterior1stVinylSd   Exterior1stWd Sdng   Exterior1stWdShing
##            3.764e-02            1.633e-02            1.690e-02
##          WoodDeckSF         YearRemodAdd           GarageArea
##            9.043e-05            6.150e-04            1.086e-04
##     FoundationCBlock      FoundationPConc       FoundationSlab
##            1.418e-02            3.663e-02           -2.902e-02
##      FoundationStone       FoundationWood         LandSlopeMod
##            1.171e-01           -1.221e-01            3.184e-02
##         LandSlopeSev        EnclosedPorch          HeatingQCFa
##           -1.115e-01            1.312e-04           -2.501e-02
##          HeatingQCGd          HeatingQCPo          HeatingQCTA
##           -2.146e-02           -6.681e-02           -3.134e-02
##     LotConfigCulDSac         LotConfigFR2         LotConfigFR3
##            2.358e-02           -2.409e-02           -8.676e-02
##      LotConfigInside           BsmtFinSF2           StreetPave
##           -1.142e-02            3.865e-05            1.045e-01
##           X3SsnPorch         KitchenAbvGr             PoolArea
##            1.670e-04           -4.438e-02            1.276e-04
##             HalfBath             FullBath           X1stFlrSF
##            2.148e-02            1.429e-02            3.197e-05
##       LandContourHLS       LandContourLow       LandContourLvl
##            4.572e-02            3.844e-03            2.877e-02
```

```
summary(log_forward)
```

```
##
## Call:
```

```
## lm(formula = Log_SalePrice ~ OverallQual + Neighborhood + GrLivArea +
##     GarageCars + OverallCond + BsmtFullBath + RoofMatl + TotalBsmtSF +
##     YearBuilt + BldgType + Condition2 + MSZoning + BsmtFinSF1 +
##     SaleCondition + Functional + LotArea + CentralAir + KitchenQual +
##     Condition1 + Fireplaces + Heating + ScreenPorch + SaleType +
##     Exterior1st + WoodDeckSF + YearRemodAdd + GarageArea + Foundation +
##     LandSlope + EnclosedPorch + HeatingQC + LotConfig + BsmtFinSF2 +
##     Street + X3SsnPorch + KitchenAbvGr + PoolArea + HalfBath +
##     FullBath + X1stFlrSF + LandContour, data = train_clean)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.69318 -0.04845  0.00053  0.05564  0.69318
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)           1.961e+00  7.446e-01   2.634 0.008539 **
## OverallQual           4.974e-02  4.254e-03  11.692  < 2e-16 ***
## NeighborhoodBlueste  -6.829e-02  8.437e-02  -0.809 0.418439
## NeighborhoodBrDale   -6.628e-02  4.756e-02  -1.394 0.163633
## NeighborhoodBrkSide   6.636e-03  4.006e-02   0.166 0.868455
## NeighborhoodClearCr   3.222e-02  3.979e-02   0.810 0.418240
## NeighborhoodCollgCr  -2.703e-02  3.132e-02  -0.863 0.388324
## NeighborhoodCrawfor   9.132e-02  3.690e-02   2.475 0.013453 *
## NeighborhoodEdwards  -7.714e-02  3.445e-02  -2.239 0.025324 *
## NeighborhoodGilbert  -2.667e-02  3.337e-02  -0.799 0.424346
## NeighborhoodIDOTRR   -3.520e-02  4.638e-02  -0.759 0.448074
## NeighborhoodMeadowV  -1.647e-01  4.813e-02  -3.423 0.000639 ***
## NeighborhoodMitchel  -6.428e-02  3.510e-02  -1.831 0.067253 .
## NeighborhoodNAmes    -4.014e-02  3.356e-02  -1.196 0.231781
## NeighborhoodNoRidge   1.924e-02  3.564e-02   0.540 0.589321
## NeighborhoodNPkVill   1.043e-03  4.825e-02   0.022 0.982762
## NeighborhoodNridgHt   7.529e-02  3.141e-02   2.397 0.016655 *
## NeighborhoodNWAmes   -4.823e-02  3.473e-02  -1.389 0.165158
## NeighborhoodOldTown  -5.715e-02  4.115e-02  -1.389 0.165103
## NeighborhoodSawyer   -3.720e-02  3.511e-02  -1.059 0.289573
## NeighborhoodSawyerW  -2.268e-02  3.378e-02  -0.671 0.502027
## NeighborhoodSomerst   1.865e-02  3.860e-02   0.483 0.629015
## NeighborhoodStoneBr   1.016e-01  3.587e-02   2.834 0.004667 **
## NeighborhoodSWISU    -9.875e-03  4.147e-02  -0.238 0.811801
## NeighborhoodTimber   -8.342e-03  3.564e-02  -0.234 0.814958
## NeighborhoodVeenker   2.735e-02  4.612e-02   0.593 0.553367
## GrLivArea             2.366e-04  1.340e-05  17.660  < 2e-16 ***
## GarageCars            2.931e-02  9.605e-03   3.052 0.002318 **
## OverallCond           3.815e-02  3.614e-03  10.556  < 2e-16 ***
## BsmtFullBath          2.596e-02  8.023e-03   3.236 0.001243 **
## RoofMatlCompShg       2.618e+00  1.370e-01  19.112  < 2e-16 ***
## RoofMatlMembran       2.971e+00  1.877e-01  15.825  < 2e-16 ***
## RoofMatlMetal         2.788e+00  1.852e-01  15.055  < 2e-16 ***
## RoofMatlRoll          2.688e+00  1.760e-01  15.267  < 2e-16 ***
```

```
## RoofMatlTar&Grv       2.678e+00  1.423e-01  18.818  < 2e-16 ***
## RoofMatlWdShake       2.645e+00  1.483e-01  17.831  < 2e-16 ***
## RoofMatlWdShngl       2.711e+00  1.418e-01  19.119  < 2e-16 ***
## TotalBsmtSF           7.237e-05  1.683e-05   4.301 1.82e-05 ***
## YearBuilt             2.106e-03  3.014e-04   6.988 4.40e-12 ***
## BldgType2fmCon       -3.720e-03  2.505e-02  -0.149 0.881960
## BldgTypeDuplex       -1.452e-02  2.668e-02  -0.544 0.586553
## BldgTypeTwnhs        -1.070e-01  2.346e-02  -4.560 5.60e-06 ***
## BldgTypeTwnhsE       -5.916e-02  1.570e-02  -3.767 0.000172 ***
## Condition2Feedr       5.222e-02  9.820e-02   0.532 0.594965
## Condition2Norm        2.052e-02  8.370e-02   0.245 0.806355
## Condition2PosA        3.176e-01  1.388e-01   2.289 0.022241 *
## Condition2PosN       -8.534e-01  1.190e-01  -7.169 1.25e-12 ***
## Condition2RRAe       -7.408e-02  1.387e-01  -0.534 0.593341
## Condition2RRAn       -6.901e-02  1.388e-01  -0.497 0.619062
## Condition2RRNn       -4.985e-02  1.170e-01  -0.426 0.670123
## MSZoningFV            4.162e-01  5.292e-02   7.864 7.66e-15 ***
## MSZoningRH            3.981e-01  5.289e-02   7.528 9.50e-14 ***
## MSZoningRL            4.021e-01  4.505e-02   8.924  < 2e-16 ***
## MSZoningRM            3.674e-01  4.206e-02   8.736  < 2e-16 ***
## BsmtFinSF1            7.546e-05  1.055e-05   7.154 1.39e-12 ***
## SaleConditionAdjLand  9.985e-02  5.996e-02   1.665 0.096078 .
## SaleConditionAlloca   6.605e-02  3.789e-02   1.743 0.081510 .
## SaleConditionFamily   1.937e-02  2.743e-02   0.706 0.480189
## SaleConditionNormal   7.330e-02  1.278e-02   5.737 1.20e-08 ***
## SaleConditionPartial -3.962e-02  6.737e-02  -0.588 0.556531
## FunctionalMaj2       -2.196e-01  5.836e-02  -3.763 0.000176 ***
## FunctionalMin1        4.221e-02  3.669e-02   1.151 0.250075
## FunctionalMin2        3.785e-02  3.606e-02   1.050 0.294103
## FunctionalMod        -6.446e-02  4.331e-02  -1.488 0.136920
## FunctionalSev        -3.512e-01  1.193e-01  -2.943 0.003304 **
## FunctionalTyp         7.919e-02  3.128e-02   2.532 0.011454 *
## LotArea               2.434e-06  4.341e-07   5.607 2.50e-08 ***
## CentralAirY           5.879e-02  1.622e-02   3.624 0.000301 ***
## KitchenQualFa        -6.694e-02  2.615e-02  -2.559 0.010593 *
## KitchenQualGd        -6.691e-02  1.408e-02  -4.753 2.23e-06 ***
## KitchenQualTA        -6.559e-02  1.638e-02  -4.004 6.57e-05 ***
## Condition1Feedr       3.068e-02  2.174e-02   1.411 0.158401
## Condition1Norm        7.961e-02  1.787e-02   4.456 9.05e-06 ***
## Condition1PosA        5.086e-02  4.395e-02   1.157 0.247387
## Condition1PosN        7.437e-02  3.238e-02   2.297 0.021777 *
## Condition1RRAe       -4.347e-02  4.070e-02  -1.068 0.285650
## Condition1RRAn        4.936e-02  2.996e-02   1.648 0.099607 .
## Condition1RRNe        1.130e-02  8.036e-02   0.141 0.888183
## Condition1RRNn        9.622e-02  5.573e-02   1.726 0.084495 .
## Fireplaces            2.462e-02  5.910e-03   4.165 3.32e-05 ***
## HeatingGasA           1.405e-01  1.110e-01   1.267 0.205542
## HeatingGasW           2.096e-01  1.139e-01   1.840 0.066038 .
## HeatingGrav          -6.718e-03  1.194e-01  -0.056 0.955155
## HeatingOthW           1.003e-01  1.373e-01   0.730 0.465306
```

```
## HeatingWall           2.365e-01  1.267e-01   1.867 0.062145 .
## ScreenPorch           2.618e-04  5.428e-05   4.823 1.58e-06 ***
## SaleTypeCon           8.764e-02  8.021e-02   1.093 0.274729
## SaleTypeConLD         1.370e-01  4.290e-02   3.193 0.001441 **
## SaleTypeConLI        -2.739e-02  5.194e-02  -0.527 0.598124
## SaleTypeConLw         2.283e-02  5.327e-02   0.429 0.668270
## SaleTypeCWD           9.768e-02  5.818e-02   1.679 0.093418 .
## SaleTypeNew           1.535e-01  6.965e-02   2.203 0.027743 *
## SaleTypeOth           7.648e-02  6.562e-02   1.166 0.244018
## SaleTypeWD           -1.116e-02  1.861e-02  -0.600 0.548589
## Exterior1stAsphShn    4.498e-03  1.143e-01   0.039 0.968622
## Exterior1stBrkComm   -1.900e-01  8.745e-02  -2.173 0.029966 *
## Exterior1stBrkFace    8.811e-02  3.168e-02   2.782 0.005484 **
## Exterior1stCBlock    -1.500e-02  1.131e-01  -0.133 0.894491
## Exterior1stCemntBd    4.699e-02  3.296e-02   1.426 0.154131
## Exterior1stHdBoard    1.555e-02  2.880e-02   0.540 0.589161
## Exterior1stImStucc   -8.017e-03  1.119e-01  -0.072 0.942892
## Exterior1stMetalSd    4.586e-02  2.805e-02   1.635 0.102322
## Exterior1stPlywood    2.000e-02  3.042e-02   0.657 0.511057
## Exterior1stStone     -1.983e-02  8.808e-02  -0.225 0.821929
## Exterior1stStucco     2.643e-02  3.508e-02   0.754 0.451231
## Exterior1stVinylSd    3.764e-02  2.818e-02   1.336 0.181871
## Exterior1stWd Sdng    1.633e-02  2.790e-02   0.585 0.558547
## Exterior1stWdShing    1.690e-02  3.494e-02   0.484 0.628727
## WoodDeckSF            9.043e-05  2.580e-05   3.504 0.000473 ***
## YearRemodAdd          6.150e-04  2.352e-04   2.615 0.009034 **
## GarageArea            1.086e-04  3.267e-05   3.323 0.000916 ***
## FoundationCBlock      1.418e-02  1.374e-02   1.032 0.302259
## FoundationPConc       3.663e-02  1.517e-02   2.415 0.015861 *
## FoundationSlab       -2.902e-02  3.322e-02  -0.873 0.382568
## FoundationStone       1.171e-01  4.633e-02   2.529 0.011570 *
## FoundationWood       -1.221e-01  6.579e-02  -1.855 0.063757 .
## LandSlopeMod          3.184e-02  1.726e-02   1.845 0.065272 .
## LandSlopeSev         -1.115e-01  4.569e-02  -2.442 0.014755 *
## EnclosedPorch         1.312e-04  5.454e-05   2.407 0.016236 *
## HeatingQCFa          -2.501e-02  2.045e-02  -1.223 0.221497
## HeatingQCGd          -2.146e-02  9.221e-03  -2.327 0.020119 *
## HeatingQCPo          -6.681e-02  1.163e-01  -0.574 0.565731
## HeatingQCTA          -3.134e-02  9.122e-03  -3.435 0.000610 ***
## LotConfigCulDSac      2.358e-02  1.394e-02   1.692 0.090899 .
## LotConfigFR2         -2.409e-02  1.789e-02  -1.347 0.178224
## LotConfigFR3         -8.676e-02  5.692e-02  -1.524 0.127680
## LotConfigInside      -1.142e-02  7.773e-03  -1.469 0.142096
## BsmtFinSF2            3.865e-05  2.047e-05   1.888 0.059220 .
## StreetPave            1.045e-01  5.051e-02   2.069 0.038732 *
## X3SsnPorch            1.670e-04  1.009e-04   1.655 0.098080 .
## KitchenAbvGr         -4.438e-02  2.359e-02  -1.881 0.060150 .
## PoolArea              1.276e-04  7.825e-05   1.631 0.103160
## HalfBath              2.148e-02  8.871e-03   2.421 0.015613 *
## FullBath              1.429e-02  9.549e-03   1.496 0.134874
```

```
## X1stFlrSF             3.197e-05  1.978e-05   1.616 0.106256
## LandContourHLS        4.572e-02  2.264e-02   2.020 0.043601 *
## LandContourLow        3.844e-03  2.746e-02   0.140 0.888696
## LandContourLvl        2.877e-02  1.615e-02   1.782 0.075012 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1058 on 1322 degrees of freedom
## Multiple R-squared:  0.9364, Adjusted R-squared:  0.9298
## F-statistic: 142.1 on 137 and 1322 DF,  p-value: < 2.2e-16
```

Backward Elimination

```
#Do a backwards elimination
log_backward = step(log_all, direction = 'backward', scope =
formula(log_all), trace = 0)
log_backward
```

```
##
## Call:
## lm(formula = Log_SalePrice ~ MSZoning + LotArea + Street + LandContour +
##     Utilities + LotConfig + LandSlope + Neighborhood + Condition1 +
##     Condition2 + BldgType + OverallQual + OverallCond + YearBuilt +
##     YearRemodAdd + RoofStyle + RoofMatl + Exterior1st + Foundation +
##     BsmtFinSF1 + BsmtFinSF2 + BsmtUnfSF + Heating + HeatingQC +
##     CentralAir + X1stFlrSF + X2ndFlrSF + LowQualFinSF + BsmtFullBath +
##     FullBath + HalfBath + KitchenAbvGr + KitchenQual + TotRmsAbvGrd +
##     Functional + Fireplaces + GarageCars + GarageArea + WoodDeckSF +
##     OpenPorchSF + EnclosedPorch + X3SsnPorch + ScreenPorch +
##     PoolArea + SaleType + SaleCondition, data = train_clean)
##
## Coefficients:
##          (Intercept)            MSZoningFV            MSZoningRH
##            1.970e+00             4.195e-01             3.978e-01
##           MSZoningRL            MSZoningRM               LotArea
##            4.043e-01             3.706e-01             2.612e-06
##           StreetPave        LandContourHLS        LandContourLow
##            1.115e-01             4.555e-02            -3.820e-03
##       LandContourLvl       UtilitiesNoSeWa      LotConfigCulDSac
##            2.919e-02            -1.583e-01             2.527e-02
##         LotConfigFR2          LotConfigFR3       LotConfigInside
##           -2.617e-02            -8.993e-02            -1.216e-02
##         LandSlopeMod          LandSlopeSev   NeighborhoodBlueste
##            3.592e-02            -1.396e-01            -6.886e-02
##   NeighborhoodBrDale   NeighborhoodBrkSide   NeighborhoodClearCr
##           -7.204e-02             6.798e-03             1.985e-02
##  NeighborhoodCollgCr   NeighborhoodCrawfor   NeighborhoodEdwards
##           -2.409e-02             9.053e-02            -7.966e-02
##  NeighborhoodGilbert    NeighborhoodIDOTRR   NeighborhoodMeadowV
##           -2.317e-02            -3.746e-02            -1.666e-01
```

```
##  NeighborhoodMitchel     NeighborhoodNAmes   NeighborhoodNoRidge
##            -6.541e-02            -4.210e-02            2.451e-02
##  NeighborhoodNPkVill   NeighborhoodNridgHt    NeighborhoodNWAmes
##            -5.221e-03             7.565e-02            -5.309e-02
##  NeighborhoodOldTown    NeighborhoodSawyer   NeighborhoodSawyerW
##            -5.918e-02            -3.922e-02            -2.164e-02
##  NeighborhoodSomerst   NeighborhoodStoneBr     NeighborhoodSWISU
##             1.882e-02             1.051e-01            -4.200e-03
##   NeighborhoodTimber   NeighborhoodVeenker       Condition1Feedr
##             1.474e-03             2.807e-02             2.931e-02
##       Condition1Norm       Condition1PosA        Condition1PosN
##             7.775e-02             5.641e-02             7.106e-02
##       Condition1RRAe       Condition1RRAn        Condition1RRNe
##            -4.570e-02             4.124e-02             8.165e-03
##       Condition1RRNn       Condition2Feedr        Condition2Norm
##             9.151e-02             6.195e-02             1.825e-02
##       Condition2PosA       Condition2PosN        Condition2RRAe
##             2.888e-01            -8.480e-01            -4.896e-01
##       Condition2RRAn       Condition2RRNn        BldgType2fmCon
##            -6.496e-02            -5.222e-02            -1.714e-03
##       BldgTypeDuplex        BldgTypeTwnhs        BldgTypeTwnhsE
##            -1.159e-02            -9.902e-02            -5.216e-02
##          OverallQual          OverallCond             YearBuilt
##             4.890e-02             3.792e-02             2.127e-03
##         YearRemodAdd       RoofStyleGable      RoofStyleGambrel
##             6.145e-04            -4.127e-02            -3.870e-02
##         RoofStyleHip      RoofStyleMansard         RoofStyleShed
##            -3.778e-02             1.390e-02             3.639e-01
##      RoofMatlCompShg       RoofMatlMembran         RoofMatlMetal
##             2.614e+00             2.982e+00             2.785e+00
##         RoofMatlRoll       RoofMatlTar&Grv       RoofMatlWdShake
##             2.674e+00             2.643e+00             2.536e+00
##      RoofMatlWdShngl    Exterior1stAsphShn    Exterior1stBrkComm
##             2.721e+00             1.124e-03            -1.888e-01
##   Exterior1stBrkFace     Exterior1stCBlock    Exterior1stCemntBd
##             8.665e-02            -1.332e-02             4.247e-02
##   Exterior1stHdBoard    Exterior1stImStucc    Exterior1stMetalSd
##             1.355e-02            -7.294e-03             4.455e-02
##   Exterior1stPlywood      Exterior1stStone     Exterior1stStucco
##             1.805e-02             2.344e-02             2.594e-02
##   Exterior1stVinylSd    Exterior1stWd Sdng    Exterior1stWdShing
##             3.459e-02             1.615e-02             1.675e-02
##     FoundationCBlock       FoundationPConc        FoundationSlab
##             1.501e-02             3.398e-02            -2.946e-02
##      FoundationStone        FoundationWood            BsmtFinSF1
##             1.042e-01            -1.245e-01             1.476e-04
##           BsmtFinSF2             BsmtUnfSF           HeatingGasA
##             1.105e-04             7.089e-05             1.403e-01
##          HeatingGasW           HeatingGrav           HeatingOthW
##             2.057e-01            -8.994e-03             1.145e-01
```

```
##          HeatingWall           HeatingQCFa           HeatingQCGd
##            2.384e-01            -2.591e-02            -2.052e-02
##          HeatingQCPo           HeatingQCTA           CentralAirY
##           -6.339e-02            -3.158e-02             6.079e-02
##            X1stFlrSF             X2ndFlrSF          LowQualFinSF
##            2.543e-04             2.224e-04             1.556e-04
##         BsmtFullBath              FullBath              HalfBath
##            2.402e-02             1.455e-02             1.905e-02
##         KitchenAbvGr         KitchenQualFa         KitchenQualGd
##           -5.192e-02            -5.996e-02            -6.577e-02
##        KitchenQualTA          TotRmsAbvGrd        FunctionalMaj2
##           -6.474e-02             5.611e-03            -2.158e-01
##       FunctionalMin1        FunctionalMin2         FunctionalMod
##            3.549e-02             3.598e-02            -7.274e-02
##        FunctionalSev         FunctionalTyp            Fireplaces
##           -3.732e-01             7.502e-02             2.457e-02
##           GarageCars            GarageArea            WoodDeckSF
##            2.742e-02             1.140e-04             9.159e-05
##          OpenPorchSF         EnclosedPorch            X3SsnPorch
##            7.015e-05             1.373e-04             1.779e-04
##          ScreenPorch              PoolArea           SaleTypeCon
##            2.752e-04             1.330e-04             8.705e-02
##        SaleTypeConLD         SaleTypeConLI         SaleTypeConLw
##            1.352e-01            -3.642e-02             2.030e-02
##          SaleTypeCWD           SaleTypeNew           SaleTypeOth
##            9.584e-02             1.324e-01             7.144e-02
##           SaleTypeWD  SaleConditionAdjLand   SaleConditionAlloca
##           -1.502e-02             8.919e-02             6.272e-02
##  SaleConditionFamily   SaleConditionNormal  SaleConditionPartial
##            1.688e-02             7.027e-02            -2.679e-02
```

```
summary(log_backward)
```

```
##
## Call:
## lm(formula = Log_SalePrice ~ MSZoning + LotArea + Street + LandContour +
##     Utilities + LotConfig + LandSlope + Neighborhood + Condition1 +
##     Condition2 + BldgType + OverallQual + OverallCond + YearBuilt +
##     YearRemodAdd + RoofStyle + RoofMatl + Exterior1st + Foundation +
##     BsmtFinSF1 + BsmtFinSF2 + BsmtUnfSF + Heating + HeatingQC +
##     CentralAir + X1stFlrSF + X2ndFlrSF + LowQualFinSF + BsmtFullBath +
##     FullBath + HalfBath + KitchenAbvGr + KitchenQual + TotRmsAbvGrd +
##     Functional + Fireplaces + GarageCars + GarageArea + WoodDeckSF +
##     OpenPorchSF + EnclosedPorch + X3SsnPorch + ScreenPorch +
##     PoolArea + SaleType + SaleCondition, data = train_clean)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.69689 -0.04738  0.00043  0.05437  0.69689
##
```

```
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)            1.970e+00  7.499e-01   2.627 0.008719 **
## MSZoningFV             4.195e-01  5.299e-02   7.916 5.20e-15 ***
## MSZoningRH             3.978e-01  5.294e-02   7.513 1.06e-13 ***
## MSZoningRL             4.043e-01  4.521e-02   8.942  < 2e-16 ***
## MSZoningRM             3.706e-01  4.227e-02   8.767  < 2e-16 ***
## LotArea                2.612e-06  4.414e-07   5.917 4.18e-09 ***
## StreetPave             1.115e-01  5.047e-02   2.208 0.027395 *
## LandContourHLS         4.555e-02  2.262e-02   2.013 0.044289 *
## LandContourLow        -3.820e-03  2.759e-02  -0.138 0.889919
## LandContourLvl         2.919e-02  1.613e-02   1.809 0.070626 .
## UtilitiesNoSeWa       -1.583e-01  1.121e-01  -1.413 0.157986
## LotConfigCulDSac       2.527e-02  1.401e-02   1.804 0.071466 .
## LotConfigFR2          -2.617e-02  1.795e-02  -1.457 0.145266
## LotConfigFR3          -8.993e-02  5.688e-02  -1.581 0.114100
## LotConfigInside       -1.216e-02  7.789e-03  -1.561 0.118747
## LandSlopeMod           3.592e-02  1.731e-02   2.076 0.038116 *
## LandSlopeSev          -1.396e-01  4.782e-02  -2.918 0.003583 **
## NeighborhoodBlueste   -6.886e-02  8.434e-02  -0.816 0.414375
## NeighborhoodBrDale    -7.204e-02  4.765e-02  -1.512 0.130780
## NeighborhoodBrkSide    6.798e-03  4.066e-02   0.167 0.867240
## NeighborhoodClearCr    1.985e-02  4.052e-02   0.490 0.624323
## NeighborhoodCollgCr   -2.409e-02  3.166e-02  -0.761 0.446831
## NeighborhoodCrawfor    9.053e-02  3.728e-02   2.428 0.015311 *
## NeighborhoodEdwards   -7.966e-02  3.484e-02  -2.287 0.022373 *
## NeighborhoodGilbert   -2.317e-02  3.363e-02  -0.689 0.490985
## NeighborhoodIDOTRR    -3.746e-02  4.666e-02  -0.803 0.422163
## NeighborhoodMeadowV   -1.666e-01  4.833e-02  -3.447 0.000584 ***
## NeighborhoodMitchel   -6.541e-02  3.550e-02  -1.843 0.065591 .
## NeighborhoodNAmes     -4.210e-02  3.389e-02  -1.242 0.214281
## NeighborhoodNoRidge    2.451e-02  3.626e-02   0.676 0.499099
## NeighborhoodNPkVill   -5.221e-03  4.824e-02  -0.108 0.913835
## NeighborhoodNridgHt    7.565e-02  3.151e-02   2.401 0.016494 *
## NeighborhoodNWAmes    -5.309e-02  3.509e-02  -1.513 0.130544
## NeighborhoodOldTown   -5.918e-02  4.175e-02  -1.418 0.156561
## NeighborhoodSawyer    -3.922e-02  3.542e-02  -1.107 0.268398
## NeighborhoodSawyerW   -2.164e-02  3.413e-02  -0.634 0.526271
## NeighborhoodSomerst    1.882e-02  3.890e-02   0.484 0.628706
## NeighborhoodStoneBr    1.051e-01  3.607e-02   2.915 0.003619 **
## NeighborhoodSWISU     -4.200e-03  4.219e-02  -0.100 0.920711
## NeighborhoodTimber     1.474e-03  3.598e-02   0.041 0.967334
## NeighborhoodVeenker    2.807e-02  4.633e-02   0.606 0.544724
## Condition1Feedr        2.931e-02  2.175e-02   1.347 0.178064
## Condition1Norm         7.775e-02  1.787e-02   4.350 1.46e-05 ***
## Condition1PosA         5.641e-02  4.409e-02   1.279 0.201047
## Condition1PosN         7.106e-02  3.262e-02   2.179 0.029540 *
## Condition1RRAe        -4.570e-02  4.064e-02  -1.124 0.261062
## Condition1RRAn         4.124e-02  3.013e-02   1.369 0.171315
## Condition1RRNe         8.165e-03  8.018e-02   0.102 0.918903
```

```
## Condition1RRNn        9.151e-02  5.573e-02   1.642 0.100848
## Condition2Feedr       6.195e-02  9.842e-02   0.629 0.529160
## Condition2Norm        1.825e-02  8.368e-02   0.218 0.827378
## Condition2PosA        2.888e-01  1.397e-01   2.068 0.038837 *
## Condition2PosN       -8.480e-01  1.190e-01  -7.125 1.71e-12 ***
## Condition2RRAe       -4.896e-01  1.945e-01  -2.517 0.011948 *
## Condition2RRAn       -6.496e-02  1.386e-01  -0.469 0.639418
## Condition2RRNn       -5.222e-02  1.168e-01  -0.447 0.654990
## BldgType2fmCon       -1.714e-03  2.509e-02  -0.068 0.945544
## BldgTypeDuplex       -1.159e-02  2.685e-02  -0.432 0.666149
## BldgTypeTwnhs        -9.902e-02  2.374e-02  -4.171 3.24e-05 ***
## BldgTypeTwnhsE       -5.216e-02  1.628e-02  -3.204 0.001387 **
## OverallQual           4.890e-02  4.266e-03  11.463  < 2e-16 ***
## OverallCond           3.792e-02  3.617e-03  10.485  < 2e-16 ***
## YearBuilt             2.127e-03  3.024e-04   7.032 3.27e-12 ***
## YearRemodAdd          6.145e-04  2.353e-04   2.612 0.009101 **
## RoofStyleGable       -4.127e-02  8.092e-02  -0.510 0.610102
## RoofStyleGambrel     -3.870e-02  8.757e-02  -0.442 0.658642
## RoofStyleHip         -3.778e-02  8.117e-02  -0.465 0.641705
## RoofStyleMansard      1.390e-02  9.331e-02   0.149 0.881619
## RoofStyleShed         3.639e-01  1.546e-01   2.353 0.018747 *
## RoofMatlCompShg       2.614e+00  1.376e-01  18.996  < 2e-16 ***
## RoofMatlMembran       2.982e+00  2.032e-01  14.674  < 2e-16 ***
## RoofMatlMetal         2.785e+00  2.019e-01  13.795  < 2e-16 ***
## RoofMatlRoll          2.674e+00  1.765e-01  15.147  < 2e-16 ***
## RoofMatlTar&Grv       2.643e+00  1.594e-01  16.577  < 2e-16 ***
## RoofMatlWdShake       2.536e+00  1.532e-01  16.553  < 2e-16 ***
## RoofMatlWdShngl       2.721e+00  1.425e-01  19.089  < 2e-16 ***
## Exterior1stAsphShn    1.124e-03  1.143e-01   0.010 0.992150
## Exterior1stBrkComm   -1.888e-01  8.759e-02  -2.156 0.031291 *
## Exterior1stBrkFace    8.665e-02  3.172e-02   2.732 0.006388 **
## Exterior1stCBlock    -1.332e-02  1.128e-01  -0.118 0.906051
## Exterior1stCemntBd    4.247e-02  3.307e-02   1.284 0.199269
## Exterior1stHdBoard    1.355e-02  2.888e-02   0.469 0.639026
## Exterior1stImStucc   -7.294e-03  1.119e-01  -0.065 0.948044
## Exterior1stMetalSd    4.455e-02  2.810e-02   1.585 0.113203
## Exterior1stPlywood    1.805e-02  3.055e-02   0.591 0.554818
## Exterior1stStone      2.344e-02  8.967e-02   0.261 0.793789
## Exterior1stStucco     2.594e-02  3.508e-02   0.739 0.459854
## Exterior1stVinylSd    3.459e-02  2.826e-02   1.224 0.221213
## Exterior1stWd Sdng    1.615e-02  2.797e-02   0.577 0.563718
## Exterior1stWdShing    1.675e-02  3.501e-02   0.479 0.632356
## FoundationCBlock      1.501e-02  1.381e-02   1.087 0.277070
## FoundationPConc       3.398e-02  1.520e-02   2.236 0.025536 *
## FoundationSlab       -2.946e-02  3.334e-02  -0.884 0.377025
## FoundationStone       1.042e-01  4.686e-02   2.223 0.026398 *
## FoundationWood       -1.245e-01  6.572e-02  -1.894 0.058379 .
## BsmtFinSF1            1.476e-04  1.820e-05   8.110 1.15e-15 ***
## BsmtFinSF2            1.105e-04  2.459e-05   4.491 7.70e-06 ***
## BsmtUnfSF             7.089e-05  1.690e-05   4.195 2.91e-05 ***
```

```
## HeatingGasA           1.403e-01  1.111e-01   1.263 0.206671
## HeatingGasW           2.057e-01  1.141e-01   1.803 0.071684 .
## HeatingGrav          -8.994e-03  1.196e-01  -0.075 0.940071
## HeatingOthW           1.145e-01  1.375e-01   0.833 0.405225
## HeatingWall           2.384e-01  1.269e-01   1.879 0.060494 .
## HeatingQCFa          -2.591e-02  2.044e-02  -1.267 0.205203
## HeatingQCGd          -2.052e-02  9.226e-03  -2.225 0.026278 *
## HeatingQCPo          -6.339e-02  1.160e-01  -0.546 0.584938
## HeatingQCTA          -3.158e-02  9.116e-03  -3.464 0.000549 ***
## CentralAirY           6.079e-02  1.633e-02   3.723 0.000205 ***
## X1stFlrSF             2.543e-04  2.220e-05  11.455  < 2e-16 ***
## X2ndFlrSF             2.224e-04  1.699e-05  13.091  < 2e-16 ***
## LowQualFinSF          1.556e-04  6.533e-05   2.382 0.017368 *
## BsmtFullBath          2.402e-02  8.051e-03   2.983 0.002906 **
## FullBath              1.455e-02  9.621e-03   1.512 0.130711
## HalfBath              1.905e-02  8.993e-03   2.119 0.034307 *
## KitchenAbvGr         -5.192e-02  2.400e-02  -2.164 0.030667 *
## KitchenQualFa        -5.996e-02  2.621e-02  -2.287 0.022338 *
## KitchenQualGd        -6.577e-02  1.412e-02  -4.657 3.53e-06 ***
## KitchenQualTA        -6.474e-02  1.637e-02  -3.954 8.10e-05 ***
## TotRmsAbvGrd          5.611e-03  3.802e-03   1.476 0.140297
## FunctionalMaj2       -2.158e-01  5.848e-02  -3.689 0.000234 ***
## FunctionalMin1        3.549e-02  3.675e-02   0.966 0.334395
## FunctionalMin2        3.598e-02  3.620e-02   0.994 0.320527
## FunctionalMod        -7.274e-02  4.348e-02  -1.673 0.094580 .
## FunctionalSev        -3.732e-01  1.203e-01  -3.101 0.001970 **
## FunctionalTyp         7.502e-02  3.135e-02   2.393 0.016857 *
## Fireplaces            2.457e-02  5.926e-03   4.146 3.60e-05 ***
## GarageCars            2.742e-02  9.630e-03   2.847 0.004480 **
## GarageArea            1.140e-04  3.272e-05   3.483 0.000512 ***
## WoodDeckSF            9.159e-05  2.585e-05   3.544 0.000408 ***
## OpenPorchSF           7.015e-05  5.060e-05   1.386 0.165894
## EnclosedPorch         1.373e-04  5.472e-05   2.509 0.012239 *
## X3SsnPorch            1.779e-04  1.008e-04   1.765 0.077835 .
## ScreenPorch           2.752e-04  5.460e-05   5.041 5.29e-07 ***
## PoolArea              1.330e-04  7.846e-05   1.696 0.090177 .
## SaleTypeCon           8.705e-02  8.014e-02   1.086 0.277587
## SaleTypeConLD         1.352e-01  4.299e-02   3.145 0.001700 **
## SaleTypeConLI        -3.642e-02  5.201e-02  -0.700 0.483935
## SaleTypeConLw         2.030e-02  5.324e-02   0.381 0.703111
## SaleTypeCWD           9.584e-02  5.817e-02   1.648 0.099664 .
## SaleTypeNew           1.324e-01  6.985e-02   1.896 0.058146 .
## SaleTypeOth           7.144e-02  6.553e-02   1.090 0.275787
## SaleTypeWD           -1.502e-02  1.873e-02  -0.802 0.422746
## SaleConditionAdjLand  8.919e-02  6.068e-02   1.470 0.141848
## SaleConditionAlloca   6.272e-02  3.785e-02   1.657 0.097722 .
## SaleConditionFamily   1.688e-02  2.748e-02   0.614 0.539317
## SaleConditionNormal   7.027e-02  1.281e-02   5.484 4.99e-08 ***
## SaleConditionPartial -2.679e-02  6.746e-02  -0.397 0.691372
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1056 on 1313 degrees of freedom
## Multiple R-squared:  0.9372, Adjusted R-squared:  0.9302
## F-statistic: 134.1 on 146 and 1313 DF,  p-value: < 2.2e-16
```

Stepwise Selection

```
# Perform stepwise selection using BIC
log_stepwise <- stepAIC(log_all, direction = "both", k =
log(nrow(train_clean)), trace = 0)
log_stepwise

##
## Call:
## lm(formula = Log_SalePrice ~ MSZoning + LotArea + LandSlope +
##     Condition2 + OverallQual + OverallCond + YearBuilt + YearRemodAdd +
##     RoofMatl + Foundation + BsmtFinSF1 + BsmtFinSF2 + BsmtUnfSF +
##     CentralAir + X1stFlrSF + X2ndFlrSF + LowQualFinSF + KitchenAbvGr +
##     KitchenQual + Functional + Fireplaces + GarageCars + GarageArea +
##     ScreenPorch + SaleCondition, data = train_clean)
##
## Coefficients:
##          (Intercept)            MSZoningFV            MSZoningRH
##            1.947e+00             4.069e-01             3.736e-01
##           MSZoningRL            MSZoningRM               LotArea
##            3.819e-01             2.899e-01             2.974e-06
##         LandSlopeMod          LandSlopeSev       Condition2Feedr
##            4.190e-02            -1.446e-01             1.240e-01
##       Condition2Norm        Condition2PosA        Condition2PosN
##            9.190e-02             1.898e-01            -9.177e-01
##       Condition2RRAe        Condition2RRAn        Condition2RRNn
##           -2.565e-02            -4.990e-02             7.682e-02
##          OverallQual           OverallCond             YearBuilt
##            5.993e-02             4.085e-02             1.757e-03
##         YearRemodAdd       RoofMatlCompShg       RoofMatlMembran
##            8.433e-04             3.057e+00             3.441e+00
##        RoofMatlMetal         RoofMatlRoll       RoofMatlTar&Grv
##            3.354e+00             3.042e+00             3.086e+00
##      RoofMatlWdShake       RoofMatlWdShngl      FoundationCBlock
##            3.067e+00             3.085e+00            -2.043e-02
##      FoundationPConc        FoundationSlab       FoundationStone
##            3.504e-02            -1.770e-03             1.098e-01
##       FoundationWood            BsmtFinSF1            BsmtFinSF2
##           -1.326e-01             2.013e-04             1.660e-04
##            BsmtUnfSF           CentralAirY             X1stFlrSF
##            1.040e-04             5.913e-02             2.776e-04
##            X2ndFlrSF          LowQualFinSF          KitchenAbvGr
##            2.686e-04             1.782e-04            -5.835e-02
##        KitchenQualFa         KitchenQualGd         KitchenQualTA
```

```
##           -9.382e-02            -7.497e-02            -9.049e-02
##       FunctionalMaj2        FunctionalMin1        FunctionalMin2
##           -1.300e-01             6.317e-02             6.916e-02
##        FunctionalMod         FunctionalSev         FunctionalTyp
##           -2.410e-02            -3.565e-01             1.026e-01
##           Fireplaces            GarageCars            GarageArea
##            3.373e-02             3.223e-02             1.125e-04
##          ScreenPorch  SaleConditionAdjLand   SaleConditionAlloca
##            2.092e-04             4.002e-02             6.439e-02
##  SaleConditionFamily   SaleConditionNormal  SaleConditionPartial
##            1.669e-02             7.445e-02             1.411e-01
```

```
summary(log_stepwise)
```

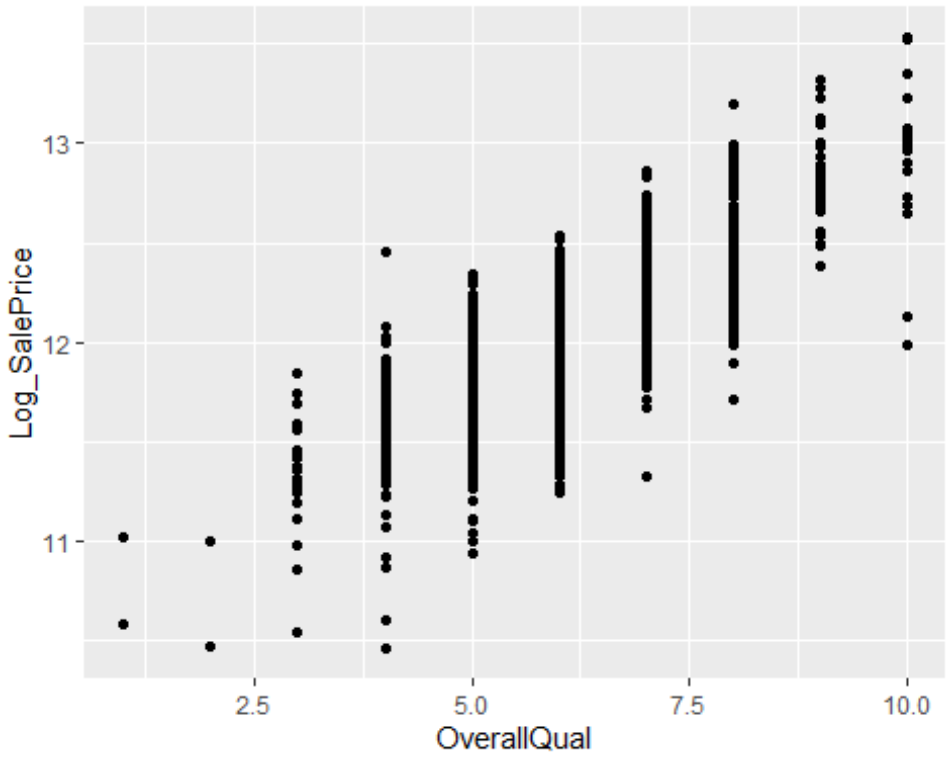
```
##
## Call:
## lm(formula = Log_SalePrice ~ MSZoning + LotArea + LandSlope +
##     Condition2 + OverallQual + OverallCond + YearBuilt + YearRemodAdd +
##     RoofMatl + Foundation + BsmtFinSF1 + BsmtFinSF2 + BsmtUnfSF +
##     CentralAir + X1stFlrSF + X2ndFlrSF + LowQualFinSF + KitchenAbvGr +
##     KitchenQual + Functional + Fireplaces + GarageCars + GarageArea +
##     ScreenPorch + SaleCondition, data = train_clean)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.82415 -0.05900  0.00302  0.06501  0.82415
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)           1.947e+00  5.492e-01   3.545 0.000406 ***
## MSZoningFV            4.069e-01  4.289e-02   9.488  < 2e-16 ***
## MSZoningRH            3.736e-01  4.888e-02   7.642 3.94e-14 ***
## MSZoningRL            3.819e-01  3.993e-02   9.563  < 2e-16 ***
## MSZoningRM            2.899e-01  4.020e-02   7.211 9.03e-13 ***
## LotArea               2.974e-06  4.222e-07   7.043 2.93e-12 ***
## LandSlopeMod          4.190e-02  1.552e-02   2.700 0.007025 **
## LandSlopeSev         -1.446e-01  4.621e-02  -3.129 0.001787 **
## Condition2Feedr       1.240e-01  9.859e-02   1.258 0.208643
## Condition2Norm        9.190e-02  8.530e-02   1.077 0.281529
## Condition2PosA        1.898e-01  1.470e-01   1.291 0.196929
## Condition2PosN       -9.177e-01  1.213e-01  -7.567 6.89e-14 ***
## Condition2RRAe       -2.565e-02  1.441e-01  -0.178 0.858752
## Condition2RRAn       -4.990e-02  1.442e-01  -0.346 0.729344
## Condition2RRNn        7.682e-02  1.202e-01   0.639 0.522748
## OverallQual           5.993e-02  4.293e-03  13.960  < 2e-16 ***
## OverallCond           4.085e-02  3.734e-03  10.940  < 2e-16 ***
## YearBuilt             1.757e-03  2.293e-04   7.662 3.40e-14 ***
## YearRemodAdd          8.433e-04  2.413e-04   3.494 0.000490 ***
## RoofMatlCompShg       3.057e+00  1.333e-01  22.944  < 2e-16 ***
## RoofMatlMembran       3.441e+00  1.901e-01  18.096  < 2e-16 ***
```

```
## RoofMatlMetal         3.354e+00  1.856e-01  18.069  < 2e-16 ***
## RoofMatlRoll          3.042e+00  1.787e-01  17.023  < 2e-16 ***
## RoofMatlTar&Grv       3.086e+00  1.390e-01  22.201  < 2e-16 ***
## RoofMatlWdShake       3.067e+00  1.444e-01  21.241  < 2e-16 ***
## RoofMatlWdShngl       3.085e+00  1.396e-01  22.097  < 2e-16 ***
## FoundationCBlock     -2.043e-02  1.360e-02  -1.502 0.133349
## FoundationPConc       3.504e-02  1.589e-02   2.205 0.027606 *
## FoundationSlab       -1.770e-03  3.288e-02  -0.054 0.957084
## FoundationStone       1.098e-01  4.968e-02   2.209 0.027333 *
## FoundationWood       -1.326e-01  6.996e-02  -1.896 0.058174 .
## BsmtFinSF1            2.013e-04  1.737e-05  11.592  < 2e-16 ***
## BsmtFinSF2            1.660e-04  2.505e-05   6.627 4.87e-11 ***
## BsmtUnfSF             1.040e-04  1.718e-05   6.054 1.81e-09 ***
## CentralAirY           5.913e-02  1.531e-02   3.861 0.000118 ***
## X1stFlrSF             2.776e-04  1.891e-05  14.684  < 2e-16 ***
## X2ndFlrSF             2.686e-04  9.516e-06  28.222  < 2e-16 ***
## LowQualFinSF          1.782e-04  6.644e-05   2.683 0.007385 **
## KitchenAbvGr         -5.835e-02  1.692e-02  -3.448 0.000582 ***
## KitchenQualFa        -9.382e-02  2.696e-02  -3.480 0.000516 ***
## KitchenQualGd        -7.497e-02  1.421e-02  -5.274 1.54e-07 ***
## KitchenQualTA        -9.049e-02  1.672e-02  -5.413 7.30e-08 ***
## FunctionalMaj2       -1.300e-01  6.193e-02  -2.099 0.036012 *
## FunctionalMin1        6.317e-02  3.850e-02   1.641 0.101081
## FunctionalMin2        6.916e-02  3.797e-02   1.821 0.068760 .
## FunctionalMod        -2.410e-02  4.538e-02  -0.531 0.595390
## FunctionalSev        -3.565e-01  1.288e-01  -2.768 0.005710 **
## FunctionalTyp         1.026e-01  3.266e-02   3.143 0.001709 **
## Fireplaces            3.373e-02  6.046e-03   5.579 2.89e-08 ***
## GarageCars            3.223e-02  9.868e-03   3.266 0.001115 **
## GarageArea            1.125e-04  3.326e-05   3.383 0.000736 ***
## ScreenPorch           2.092e-04  5.716e-05   3.659 0.000263 ***
## SaleConditionAdjLand  4.002e-02  6.135e-02   0.652 0.514277
## SaleConditionAlloca   6.439e-02  3.806e-02   1.692 0.090878 .
## SaleConditionFamily   1.669e-02  2.897e-02   0.576 0.564480
## SaleConditionNormal   7.445e-02  1.268e-02   5.870 5.42e-09 ***
## SaleConditionPartial  1.411e-01  1.742e-02   8.097 1.22e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1166 on 1403 degrees of freedom
## Multiple R-squared:  0.9181, Adjusted R-squared:  0.9148
## F-statistic: 280.8 on 56 and 1403 DF,  p-value: < 2.2e-16
```

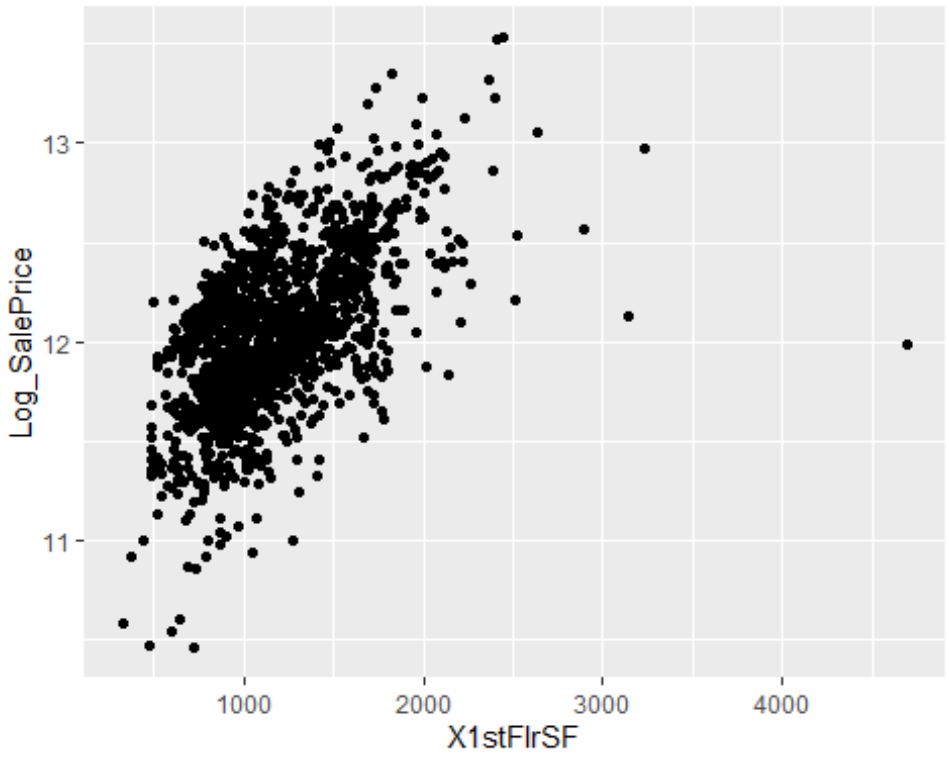
Checking Assumptions!

```
##CHECKING ASSUMPTIONS OF EACH MODEL:
#We can see that a good amount of the variables we've chosen are linearly
related
# to log_SalePrice
#Distribution of Overall Qual vs. Log_SalePrice the means increase linearly:
```

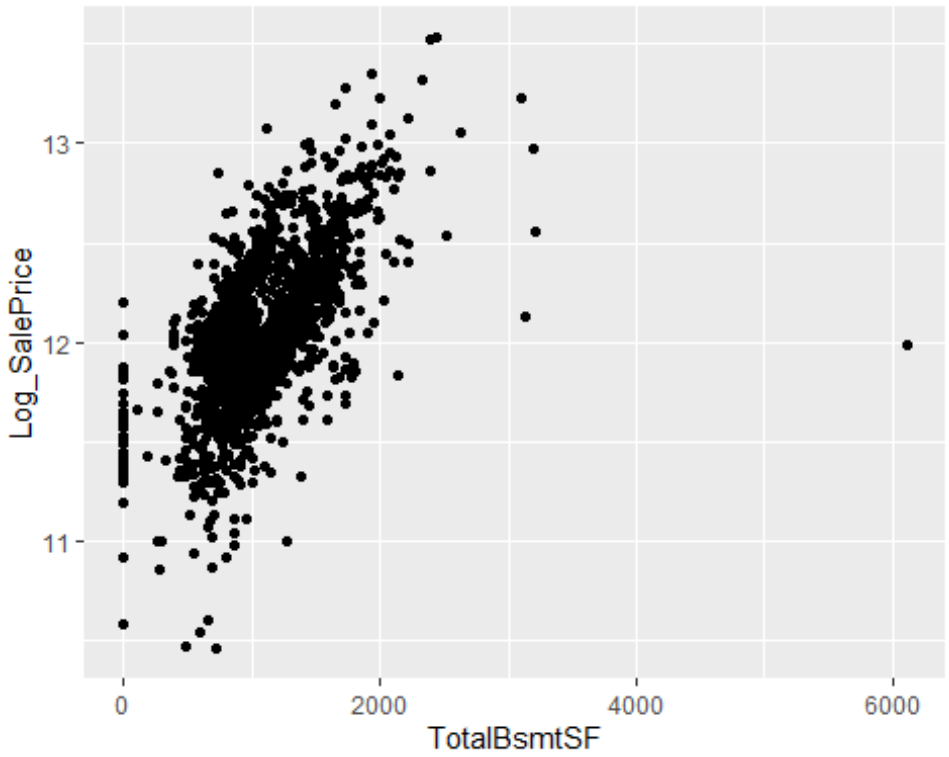
```r
ggplot(data = train_clean, aes(x = OverallQual, y = Log_SalePrice)) +
  geom_point()
```




```r
#Distribution of X1stFlrSF vs. Log_SalePrice shows the linear correlation:
ggplot(data = train_clean, aes(x = X1stFlrSF, y = Log_SalePrice)) +
  geom_point()
```




```
#Distribution of TotalBsmtSF vs. Log_SalePrice:
ggplot(data = train_clean, aes(x = TotalBsmtSF, y = Log_SalePrice)) +
  geom_point()
```



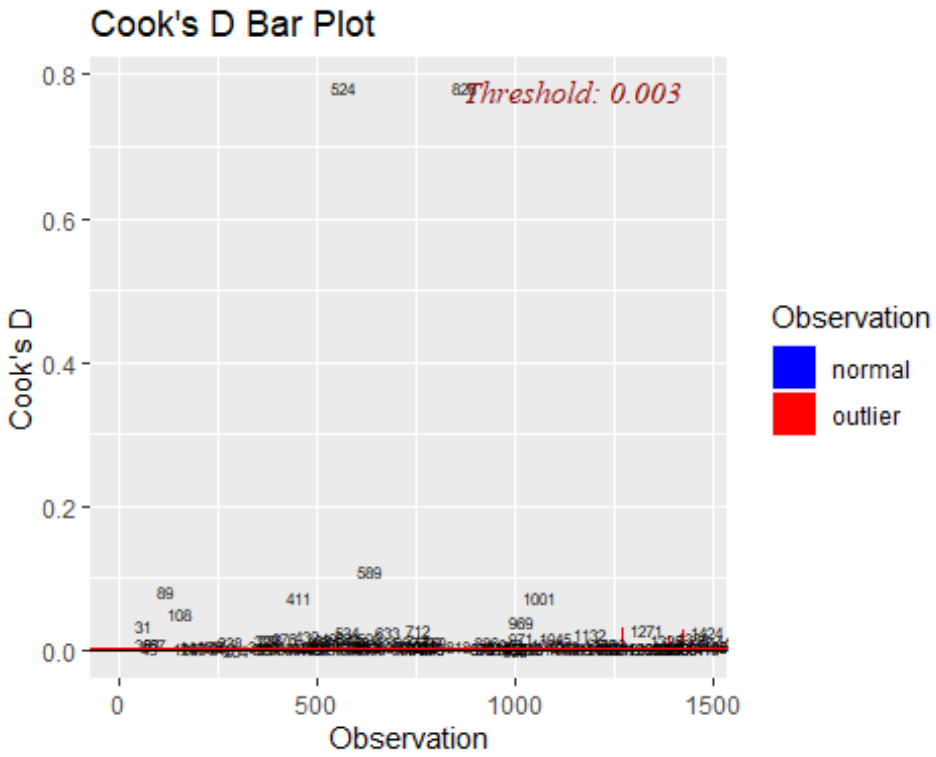
Here we see how the variables are related linearly.

```
##CHECKING VIF FOR MULTICOLLINEARITY:
# Calculate VIF using our first model (forward)
vif_values <- vif(log_forward)
# Print VIF values
print(vif_values)
```

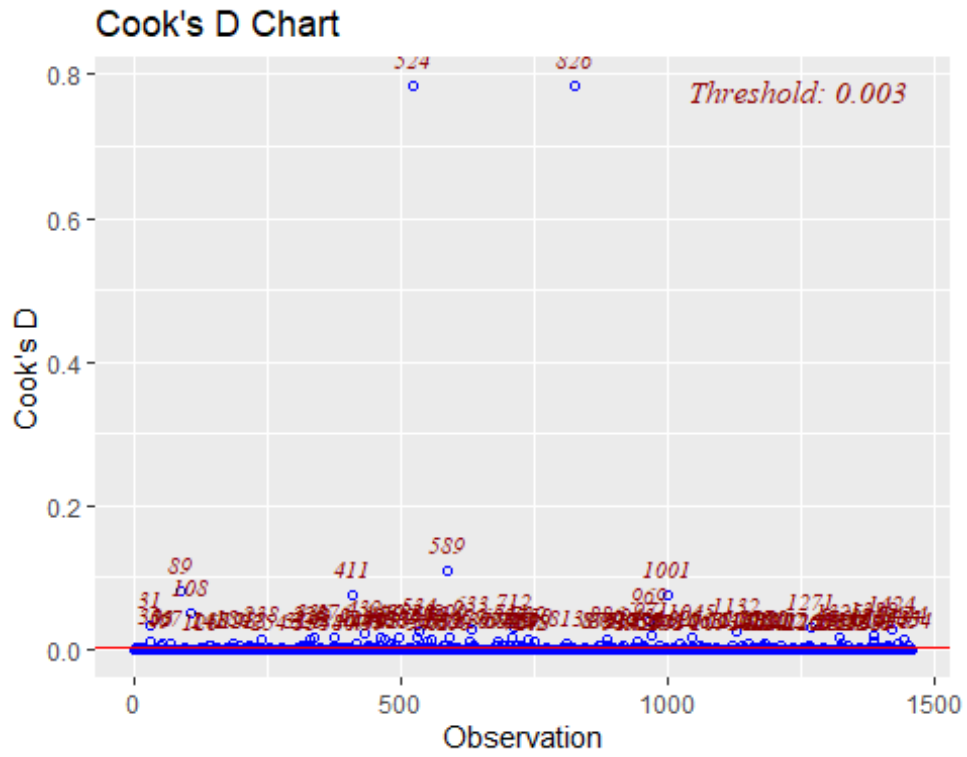
```
##                       GVIF Df GVIF^(1/(2*Df))
## OverallQual       4.509738  1        2.123614
## Neighborhood  29583.098697 24        1.239215
## GrLivArea         6.456258  1        2.540917
## GarageCars        6.713012  1        2.590948
## OverallCond       2.107146  1        1.451601
## BsmtFullBath      2.258345  1        1.502779
## RoofMatl          6.744836  7        1.146073
## TotalBsmtSF       7.100094  1        2.664600
## YearBuilt        10.800238  1        3.286372
## BldgType         18.242098  4        1.437588
## Condition2        3.312698  7        1.089321
## MSZoning         41.212701  4        1.591765
## BsmtFinSF1        3.015461  1        1.736508
## SaleCondition   111.117153  5        1.601689
## Functional        2.722810  6        1.087055
## LotArea           2.446549  1        1.564145
## CentralAir        2.087894  1        1.444955
## KitchenQual       5.438793  3        1.326123
## Condition1        5.785656  8        1.115956
## Fireplaces        1.891535  1        1.375331
## Heating           3.345264  5        1.128348
## ScreenPorch       1.193337  1        1.092400
## SaleType        104.774797  8        1.337415
## Exterior1st      47.637912 14        1.147960
## WoodDeckSF        1.363009  1        1.167480
## YearRemodAdd      3.072281  1        1.752792
## GarageArea        6.358661  1        2.521639
## Foundation       16.701179  5        1.325180
## LandSlope         3.804207  2        1.396581
## EnclosedPorch     1.447600  1        1.203163
## HeatingQC         4.533722  4        1.207972
## LotConfig         1.813242  4        1.077226
## BsmtFinSF2        1.421127  1        1.192110
## Street            1.361650  1        1.166897
## X3SsnPorch        1.140413  1        1.067901
## KitchenAbvGr      3.521002  1        1.876433
## PoolArea          1.287709  1        1.134773
## HalfBath          2.593240  1        1.610354
## FullBath          3.606283  1        1.899021
## X1stFlrSF         7.618886  1        2.760233
## LandContour       3.441721  3        1.228747
```

There are a few variables with high multicollinearity (Neighborhood, MSZoning and Sale Condition) we will leave these for our forward model. While brainstorming for our custom model these will be the specific variables we will leave out.

```
# Plot Cook's distance
ols_plot_cooksd_bar(log_forward) #We can see that there are only two
observations
```




```
# with a Cook's D greater than 0.2. The rest fall below it. Since there are
1400+
# observations we can leave them in.
ols_plot_cooksd_chart(log_forward)
```



We can see that there are only two observations with a Cook's D greater than 0.2. The rest fall below it. Since there are 1400+ observations we can leave them in.

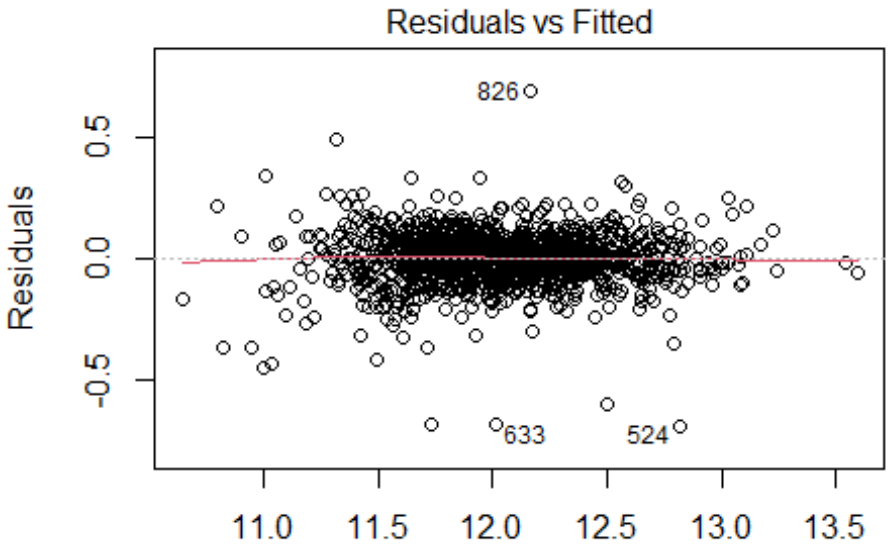
```
#CHECKING FOR HETEROSCEDASTICITY:
bptest(log_forward)

##
##  studentized Breusch-Pagan test
##
## data:  log_forward
## BP = 591.75, df = 137, p-value < 2.2e-16

bptest(log_backward)

##
##  studentized Breusch-Pagan test
##
## data:  log_backward
## BP = 602.88, df = 146, p-value < 2.2e-16

bptest(log_stepwise)

##
##  studentized Breusch-Pagan test
##
## data:  log_stepwise
## BP = 688.74, df = 56, p-value < 2.2e-16
```

```
#Each p-value < 2.2e-16
```

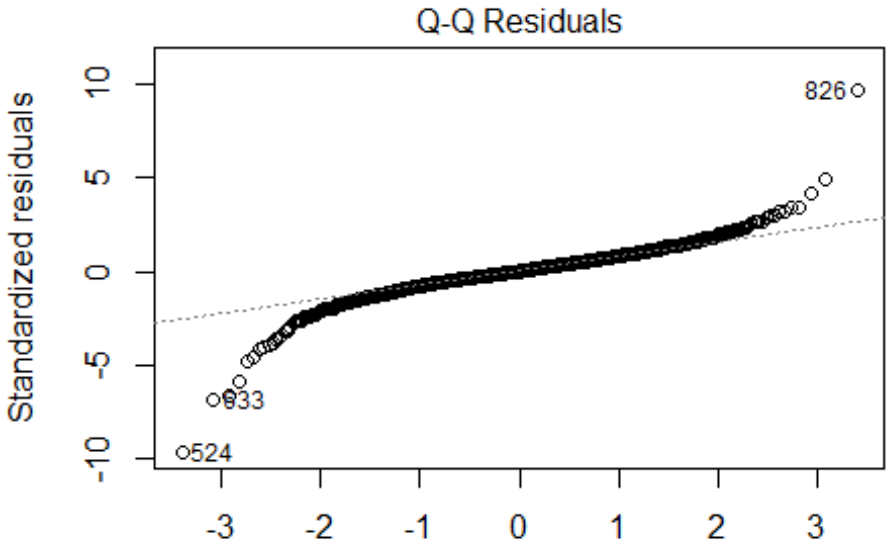
When running the Studentized Breusch-Pagan test, our respective p-value for each of the models is < 2.2e-16. This extremely small p-value provides evidence against Heteroscedasticity, meaning, the variance across variables is constant.

```
plot(log_forward)
```

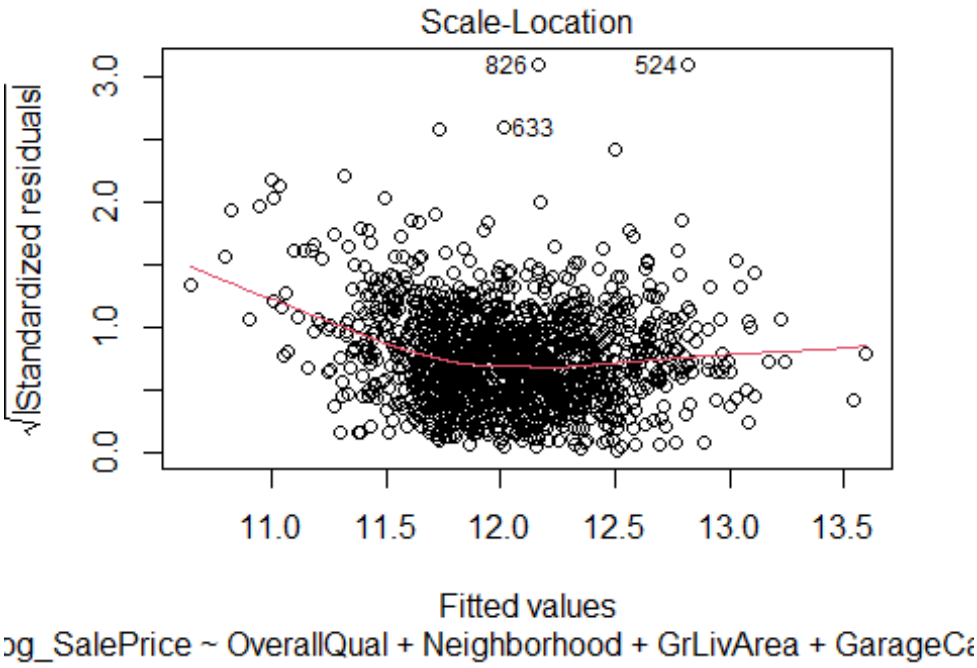
```
## Warning: not plotting observations with leverage one:
##   121, 272, 326, 584, 667, 1004, 1012, 1188, 1231, 1276, 1299, 1322, 1371
```

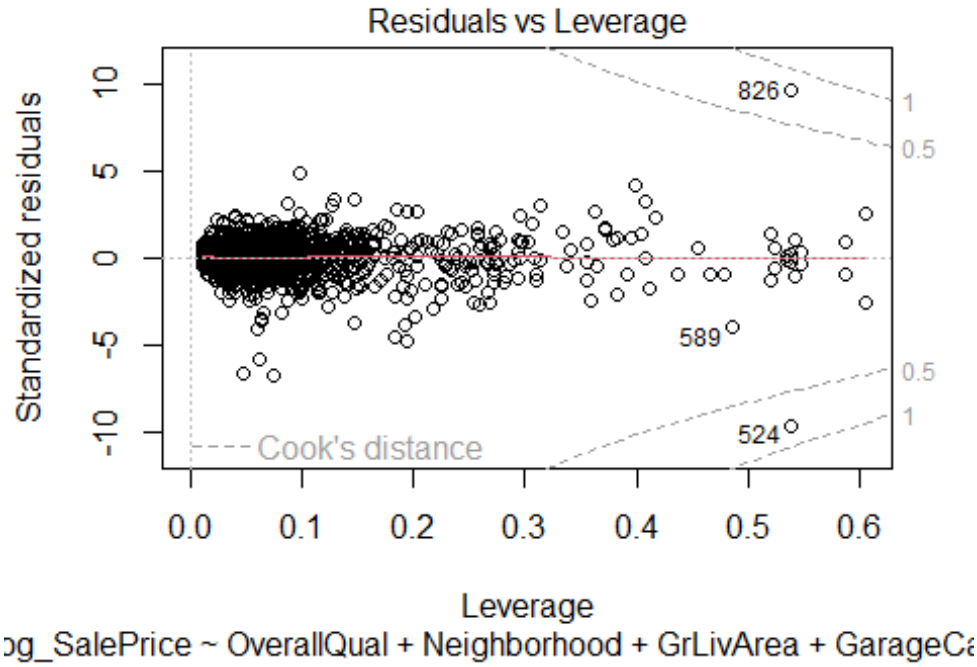


ɔg_SalePrice ~ OverallQual + Neighborhood + GrLivArea + GarageCa



ɔg_SalePrice ~ OverallQual + Neighborhood + GrLivArea + GarageCa





Our residual plot and our QQ plots both look to fulfill our assumptions.

NOW… Let's build our predictions. We will deal with simply the log_forward, log_backward, and log_stepwise first.

```
#MAKING PREDICTIONS ON THE TEST DATASET
# Predict Log_SalePrice using the forward-selected model
forward_predictions <- predict(log_forward, newdata = test_clean)

# Predict Log_SalePrice using the backward-selected model
backward_predictions <- predict(log_backward, newdata = test_clean)

# Predict Log_SalePrice using the stepwise-selected model
stepwise_predictions <- predict(log_stepwise, newdata = test_clean)

# Create a dataframe with predictions from each model
predictions_df <- data.frame(
  Forward_Predictions = forward_predictions,
  Backward_Predictions = backward_predictions,
  Stepwise_Predictions = stepwise_predictions
)
# Take the exponential of each variable to back-transform
predictions_df <- exp(predictions_df)
# Rename the columns by adding a string to indicate they represent SalePrice
colnames(predictions_df) <- paste0(colnames(predictions_df), "_SalePrice")
View(predictions_df)
```

We can see when viewing the dataframe we have all our predicted values for the Test_clean dataframe.

Analysing our Performance (Cross Validation)

Forward

```
##DEFINING FORWARD MODEL
formula_forward <- Log_SalePrice ~ OverallQual + Neighborhood + GrLivArea +
GarageCars + OverallCond + BsmtFullBath + RoofMatl + TotalBsmtSF + YearBuilt
+ BldgType + Condition2 + MSZoning + BsmtFinSF1 + SaleCondition + Functional
+ LotArea + CentralAir + KitchenQual + Condition1 + Fireplaces + Heating +
ScreenPorch + SaleType + Exterior1st + WoodDeckSF + YearRemodAdd + GarageArea
+ Foundation + LandSlope + EnclosedPorch + HeatingQC + LotConfig + BsmtFinSF2
+ Street + X3SsnPorch + KitchenAbvGr + PoolArea + HalfBath + FullBath +
X1stFlrSF + LandContour
# Train your model with 10-fold cross-validation
model_forward <- train(
  formula_forward,
  data = train_clean,
  method = "lm",
  trControl = trainControl(method = "cv", number = 10)
)

## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient
fit;
```

```
## attr(*, "non-estim") has doubtful cases

## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient
fit;
## attr(*, "non-estim") has doubtful cases

## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient
fit;
## attr(*, "non-estim") has doubtful cases

## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient
fit;
## attr(*, "non-estim") has doubtful cases

## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient
fit;
## attr(*, "non-estim") has doubtful cases

## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient
fit;
## attr(*, "non-estim") has doubtful cases

## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient
fit;
## attr(*, "non-estim") has doubtful cases

## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient
fit;
## attr(*, "non-estim") has doubtful cases

## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient
fit;
## attr(*, "non-estim") has doubtful cases

# Get cross-validated prediction errors
cv_press_forward <- model_forward$results$RMSE
# [1] 0.1804603
```

Backward

```
##DO THE SAME FOR BACKWARD:
# Define your model formula
formula_backward <- Log_SalePrice ~ MSZoning + LotArea + Street + LandContour
+
          Utilities + LotConfig + LandSlope + Neighborhood + Condition1 +
          Condition2 + BldgType + OverallQual + OverallCond + YearBuilt +
          YearRemodAdd + RoofStyle + RoofMatl + Exterior1st + Foundation +
          BsmtFinSF1 + BsmtFinSF2 + BsmtUnfSF + Heating + HeatingQC +
          CentralAir + X1stFlrSF + X2ndFlrSF + LowQualFinSF + BsmtFullBath +
          FullBath + HalfBath + KitchenAbvGr + KitchenQual + TotRmsAbvGrd +
```

```
          Functional + Fireplaces + GarageCars + GarageArea + WoodDeckSF +
          OpenPorchSF + EnclosedPorch + X3SsnPorch + ScreenPorch +
          PoolArea + SaleType + SaleCondition
# Train your model with 10-fold cross-validation
model_backward <- train(
  formula_backward,
  data = train_clean,
  method = "lm",
  trControl = trainControl(method = "cv", number = 10)
)
```

```
## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient
fit;
## attr(*, "non-estim") has doubtful cases

## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient
fit;
## attr(*, "non-estim") has doubtful cases

## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient
fit;
## attr(*, "non-estim") has doubtful cases

## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient
fit;
## attr(*, "non-estim") has doubtful cases

## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient
fit;
## attr(*, "non-estim") has doubtful cases

## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient
fit;
## attr(*, "non-estim") has doubtful cases

## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient
fit;
## attr(*, "non-estim") has doubtful cases
```

```
# Get cross-validated prediction errors
cv_press_backward <- model_backward$results$RMSE
# [1] 0.1798864
```

<u>Stepwise</u>

```
##DO THE SAME FOR STEPWISE:
# Define your model formula
formula_stepwise <- Log_SalePrice ~ MSZoning + LotArea + LandSlope +
          Condition2 + OverallQual + OverallCond + YearBuilt + YearRemodAdd +
          RoofMatl + Foundation + BsmtFinSF1 + BsmtFinSF2 + BsmtUnfSF +
```

```
          CentralAir + X1stFlrSF + X2ndFlrSF + LowQualFinSF + KitchenAbvGr +
          KitchenQual + Functional + Fireplaces + GarageCars + GarageArea +
          ScreenPorch + SaleCondition
# Train your model with 10-fold cross-validation
model_stepwise <- train(
  formula_stepwise,
  data = train_clean,
  method = "lm",
  trControl = trainControl(method = "cv", number = 10)
)

## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient
fit;
## attr(*, "non-estim") has doubtful cases

## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient
fit;
## attr(*, "non-estim") has doubtful cases

## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient
fit;
## attr(*, "non-estim") has doubtful cases

## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient
fit;
## attr(*, "non-estim") has doubtful cases

# Get cross-validated prediction errors
cv_press_stepwise <- model_stepwise$results$RMSE
# [1] 0.2003931
```

After all of our analysis for CV PRESS we see that our best performing model is the first one (Forward_Selection).

Creating/Submitting Dataframes

Here we will have to create dataframes with the ID's from test_clean with the new predicted values added in.

```
#BUILDING THE SUBMISSION DATAFRAMES:
submission_for = data.frame(
  ID = test_df$Id,
  SalePrice = predictions_df$Forward_Predictions_SalePrice)

submission_back = data.frame(
  ID = test_df$Id,
  SalePrice = predictions_df$Backward_Predictions_SalePrice)

submission_step = data.frame(
```

```
  ID = test_df$Id,
  SalePrice = predictions_df$Stepwise_Predictions_SalePrice)
```

Next and final step is to create the CSV for each submission so we can obtain a Kaggle score.

```
#SAVING THE DATAFRAMES TO CSV FOR UPLOADING TO KAGGLE
# Exporting submission_for dataframe
write.csv(submission_for, file = "submission_for.csv", row.names = FALSE)

# Exporting submission_back dataframe
write.csv(submission_back, file = "submission_back.csv", row.names = FALSE)

# Exporting submission_step dataframe
write.csv(submission_step, file = "submission_step.csv", row.names = FALSE)
```

Creating and checking the Custom Model

```
###CREATING A CUSTOM LINEAR MODEL:
# Define your custom model formula (Removing High VIF variables from our
forward model)
formula_custom <- Log_SalePrice ~ OverallQual + GrLivArea + GarageCars +
OverallCond + BsmtFullBath + RoofMatl + TotalBsmtSF + YearBuilt + BldgType +
Condition2 + BsmtFinSF1 + Functional + LotArea + CentralAir + KitchenQual +
Condition1 + Fireplaces + Heating + ScreenPorch + SaleType + Exterior1st +
WoodDeckSF + YearRemodAdd + GarageArea + Foundation + LandSlope +
EnclosedPorch + HeatingQC + LotConfig + BsmtFinSF2 + Street + X3SsnPorch +
KitchenAbvGr + PoolArea + HalfBath + FullBath + X1stFlrSF + LandContour
# Train your model with 10-fold cross-validation
model_custom <- train(
  formula_custom,
  data = train_clean,
  method = "lm",
  trControl = trainControl(method = "cv", number = 10)
)

## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient
fit;
## attr(*, "non-estim") has doubtful cases

## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient
fit;
## attr(*, "non-estim") has doubtful cases

## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient
fit;
## attr(*, "non-estim") has doubtful cases

## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient
fit;
## attr(*, "non-estim") has doubtful cases
```

```
## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient
fit;
## attr(*, "non-estim") has doubtful cases

## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient
fit;
## attr(*, "non-estim") has doubtful cases

## Warning in predict.lm(modelFit, newdata): prediction from rank-deficient
fit;
## attr(*, "non-estim") has doubtful cases
```

```
# Get cross-validated prediction errors
cv_press_custom <- model_custom$results$RMSE


# Predict Log_SalePrice using the stepwise-selected model
custom_predictions <- predict(model_custom, newdata = test_clean)

# Create a dataframe with predictions from each model
predictions_df <- data.frame(
  Forward_Predictions = forward_predictions,
  Backward_Predictions = backward_predictions,
  Stepwise_Predictions = stepwise_predictions,
  Custom_Predictions = custom_predictions
)

# Take the exponential of each variable to back-transform again
predictions_df <- exp(predictions_df)
# Rename the columns by adding a string to indicate they represent SalePrice
colnames(predictions_df) <- paste0(colnames(predictions_df), "_SalePrice")

#BUILDING THE CUSTOM SUBMISSION DATAFRAMES:
submission_custom = data.frame(
  ID = test_df$Id,
  SalePrice = predictions_df$Custom_Predictions_SalePrice)
# Exporting submission_custom dataframe
write.csv(submission_custom, file = "submission_custom.csv", row.names =
FALSE)
```

Lastly, this model did not perform better than our Forward model. So our Custom Model that was uploaded to Kaggle will be the same as our Forward model for performance stats.

GitHub Pages

Victoria Hernandez

torih1541/House-Prices-Advanced-Regression-Techniques (github.com)

Kosi Okeke
https://github.com/KOkeke94/House-Prices-Advanced-Regression-Techniques