Hello, Century 21!

(Restatement of problem)

We hear you need some answers to some very important questions with respect to your business! We can provide the answers you are looking for concerning the housing market in Ames, Iowa.

We were able to get our hands on a dataset containing 1460 home sales, and 80 associated variables. While the dataset is quite big, you’ve narrowed things down to a few specific neighborhoods you’re interested in (NAmes, Edwards, and BrkSide), leaving us with 383 houses to study. This number according to statistical methodology will be sufficient to draw conclusions on for the neighborhoods. In short, you are interested in how specific neighborhoods and square footage are associated in price.

To begin, we need to look at the requested data and determine if it is suitable for analysis in its current form, and based on the non-normality of the histograms of sale price and square footage (Plot 1, Plot 2, respectively), these variables will need to be log-transformed to address the normality assumption (Plot 5, Plot 6 respectively).

(Build and fit the model)

At this point I believe we are ready to build a model and evaluate results.

Log(SalePrice)= B0 + B1 \* log(SquareFootage) + B2 \* Edwards + B3 \* NAmes + B4 \* log(SquareFootage) \* Edwards + B5  \* log(SquareFootage) \* NAmes

Log(SalePrice)= 6.33 + 7.480e^-1 \* log(SquareFootage) + 2.393e^-1\* Edwards + 4.315e^-1\* NAmes + -2.146e^-3\* log(SquareFootage) \* Edwards + .2.483^e-4 \* log(SquareFootage) \* NAmes

(Checking assumptions)

For linear regression we have some assumptions to meet before we can make inference on a model. Based on our evaluations from this model, the assumptions are met after applying a log transformation to sales price and square footage, and because of this we are going to make inference on medians now. Those assumptions are Normality of data, homogeneity of variance, and independence. We can examine the normality assumption from plot 5 and plot 6 and conclude that the data follows the trend of normality. The quantile-quantile (plot 9 and plot 11) looks roughly normal and is about as good as we can get with a quantile-quantile plot of real data. Homogeneity of variance appears to be met (plot 10, plot 10), as there is no obvious trend to the data, it seems to be spread evenly between the two sides of the line. Finally, we have the independence assumption, we will have to make this assumption for this dataset, although realistically we cannot see this as being true, houses will be related to each other.

Additionally, the residuals looked very good given the dataset, and since we have a large number of observations the central limit theorem will come into play. The data did show some outliers and leverage points (plots 7,plot 8, and plot 9). The outliers were all investigated and we concluded that there we no measurement errors and decided to keep the outliers in the data. The model was created again with outliers removed and we determined the results were very similar to the model with the outliers included. The plots of this model are 14,15,16,17,18,19,20.

(Comparing competing models)

The model we created has an r-squared value of 0.5178, meaning that 51.78% of the variance of the sales price variable is explained by the variation in neighborhood and square footage.

Internal CV press: 13.9, only one model created.

(Parameter Estimates)

Beta 0: Estimate =6.334, Standard Error =7.060e^-1, t-value=8.972, P-value: <0.0001, significant for model

Beta 1: Estimate =7.480^e-01, Standard Error =1.176e^-01, t-value=6.359, P-value: <0.0001, significant for model

Beta 2: Estimate =2.393^e-01, Standard Error =1.12e^-01, t-value=2.136, P-value: 0.03332, significant for model

Beta 3: Estimate =4.315^e-01, Standard Error =9.929e^-02, t-value=4.346, P-value: <0.0001, significant for model

Beta 4: Estimate =-2.146^e-04, Standard Error =8.655e^-05, t-value=-2.480, P-value:0.01359, significant for model

Beta 5: Estimate =-2.483^e-04, Standard Error =7.704e^-5, t-value=-3.223, P-value: <0.00138, significant for model

DF: 376

95% CI for Beta 0: (4.945,7.721)

95% CI for Beta 1: (0.516,9.793)

95% CI for Beta 2: (0.019,4.596)

95% CI for Beta 3: (0.236,6.268)

95% CI for Beta 4: (-0.0004,-4.444)

95% CI for Beta 5: (-0.0004,-9.685)

For each term in the model, the p-values were all below the standard significance level of 0.05, and thus for each variable we will reject the null hypothesis and conclude that there is a significant difference and that each parameter is useful in the model.

To more easily interpret the model, we can take a look at the equation for each neighborhood and how sale price is related to square footage.

Log(squarefootage)|Brkside = 6.334 + 0.748log(squarefootage)

Log(squarefootage)|Edwards = 6.573 + 0.74779log(squarefootage)

Log(squarefootage)|NAmes = 6.766 +0.74775log(squarefootage)

Interpretation of model:

Since this model is log transformed we will speak in terms of medians. When we increase the size of a house in a given neighborhood (since its log transformed we will say the size is doubled).

However, since this is an observational study, we cannot infer causation and say with certainty that the square footage or neighborhood can cause a change in sale price. It can be inferred that there are many more variables at play here, and we can conclude by saying that there is an association between the selling price of a house, it’s square footage, and its neighborhood.

Conclusion:

A big thanks to Century 21 in Ames for reaching out with these great questions! To quickly summarize, We were able to get our hands on a dataset containing 1460 home sales, and 80 associated variables. While the dataset is quite big, you’ve narrowed things down to a few specific neighborhoods you’re interested in (NAmes, Edwards, and BrkSide), leaving us with 383 houses to study. This number according to statistical methodology will be sufficient to draw conclusions on for the neighborhoods. In short, you are interested in how specific neighborhoods and square footage are associated in price. We have determined that there is an association between neighborhood and square footage, if you look at the interpretations above we can see how to interpret each of the neighborhoods of interest.

Analysis 2

Hello, Century 21!

(Restatement of problem)

We hear you need some answers to some very important questions with respect to your business! We can provide the answers you are looking for concerning the housing market in Ames, Iowa.

We were able to get our hands on a dataset containing 1460 home sales, and 81 associated variables. While the dataset is quite big, we can handle this amount of data! This number according to statistical methodology will be sufficient to draw conclusions on for the sale price!. In short, you are interested in predicting sales prices of home in all of Ames Iowa, including all neighborhoods.

To begin, we need to look at the requested data and determine if it is suitable for analysis in its current form, and based on the non-normality of the histogram of sale price (Plot 24, Plot 26, respectively), these variables will need to be log-transformed to address the normality assumption (Plot 5, Plot 6 respectively). Square footage will also be log transformed, the transformation is similar to plot 6.

Model Selection:

Stepwise

(Checking assumptions)

For linear regression we have some assumptions to meet before we can make inference on a model. Based on our evaluations from this model, the assumptions are met after applying a log transformation to sales price and square footage, and because of this we are going to make inference on medians now. Those assumptions are Normality of data, homogeneity of variance, and independence. We can examine the normality assumption from plot 5 and plot 6 and conclude that the data follows the trend of normality. Additionally, we can look at the Quantile-Quantile plot (Plot 28) and determine the data is roughly normally distributed. We will assume that the data is independent. additionally, we can look at the leverage plots and cook’s distance plot to determine that there are no significant outliers that would adversely affect the model. The residual vs fitted plot checks out as well, there is no obvious trend to the data and does not indicate evidence against linearity or non-constant variance.

Forward

(Checking assumptions)

For linear regression we have some assumptions to meet before we can make inference on a model. Based on our evaluations from this model, the assumptions are met after applying a log transformation to sales price and square footage, and because of this we are going to make inference on medians now. Those assumptions are Normality of data, homogeneity of variance, and independence. We can examine the normality assumption from plot 5 and plot 6 and conclude that the data follows the trend of normality. Additionally, we can look at the Quantile-Quantile plot (Plot 31) and determine the data is roughly normally distributed. We will assume that the data is independent. additionally, we can look at the leverage plots and cook’s distance plot to determine that there are no significant outliers that would adversely affect the model. The residual vs fitted plot checks out as well, there is no obvious trend to the data and does not indicate evidence against linearity or non-constant variance.

Backward

(Checking assumptions)

For linear regression we have some assumptions to meet before we can make inference on a model. Based on our evaluations from this model, the assumptions are met after applying a log transformation to sales price and square footage, and because of this we are going to make inference on medians now. Those assumptions are Normality of data, homogeneity of variance, and independence. We can examine the normality assumption from plot 5 and plot 6 and conclude that the data follows the trend of normality. Additionally, we can look at the Quantile-Quantile plot (Plot 36) and determine the data is roughly normally distributed. We will assume that the data is independent. additionally, we can look at the leverage plots and cook’s distance plot to determine that there are no significant outliers that would adversely affect the model (Plot 37 and Plot 38 respectively). The residual vs fitted plot checks out as well, there is no obvious trend to the data and does not indicate evidence against linearity or non-constant variance.

Custom

(Checking assumptions)

For linear regression we have some assumptions to meet before we can make inference on a model. Based on our evaluations from this model, the assumptions are met after applying a log transformation to sales price and square footage, and because of this we are going to make inference on medians now. Those assumptions are Normality of data, homogeneity of variance, and independence. We can examine the normality assumption from plot 5 and plot 6 and conclude that the data follows the trend of normality. Additionally, we can look at the Quantile-Quantile plot (Plot 40) and determine the data is roughly normally distributed. We will assume that the data is independent. additionally, we can look at the leverage plots and cook’s distance plot to determine that there are no significant outliers that would adversely affect the model (Plot 42 and Plot 41 respectively). The residual vs fitted plot checks out as well, there is no obvious trend to the data and does not indicate evidence against linearity or non-constant variance.

Comparing Competing Models:

|  |  |  |  |
| --- | --- | --- | --- |
| **Predictive Models** | **Adjusted R2** | **CV PRESS** | **Kaggle Score** |
| Forward | .931 | 17.6 | 0.16557 |
| Backward | .721 | 22.7 | 0.21270 |
| Stepwise | .808 | 68.1 | 0.17457 |
| CUSTOM | .87 | 35.2 | .15615 |

https://www.kaggle.com/bensharn

Conclusions:

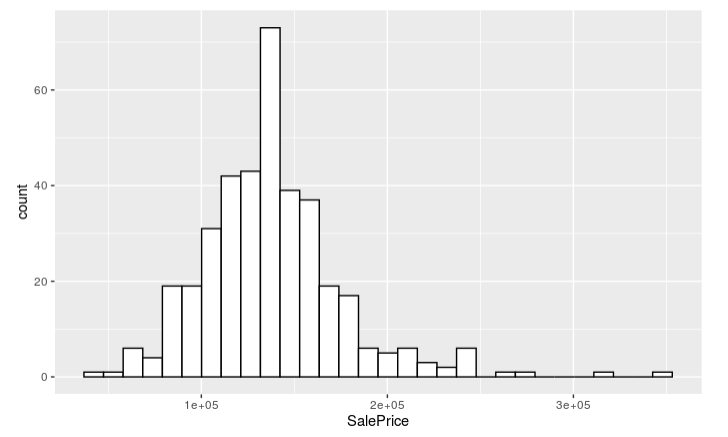
We hear you need some answers to some very important questions with respect to your business, and if you read the above you can see that we’ve got em! The Ames, Iowa housing market is an interesting one!

We were able to get our hands on a dataset containing 1460 home sales, and 81 associated variables. While the dataset is quite big, we can handle this amount of data! This number according to statistical methodology will be sufficient to draw conclusions on for the sale price!. In short, you are interested in predicting sales prices of home in all of Ames Iowa, including all neighborhoods.

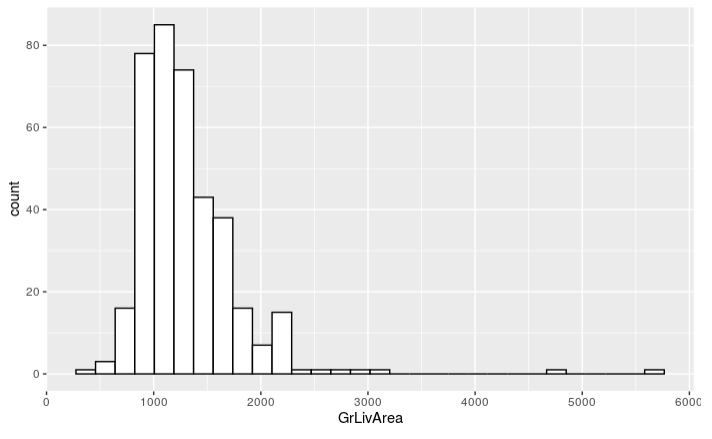
We created 4 different models using linear regression to predict sale prices of houses in Ames, and the custom model performed the best, with the lowest overall Kaggle score, meaning that this model is the closest to the truth! We took nine predictor variables (OverallQual,log(GrLivArea),Neighborhood,GarageCars,ExterQual,TotalBsmtSF,GarageArea,KitchenQual and yearBuilt) and developed those into a regression model with surprisingly good results. We ended up with an R^2 value of .87, meaning that 87% of the variance of the sales price variable is explained by the variation in our nine predictors variables.

Appendix:

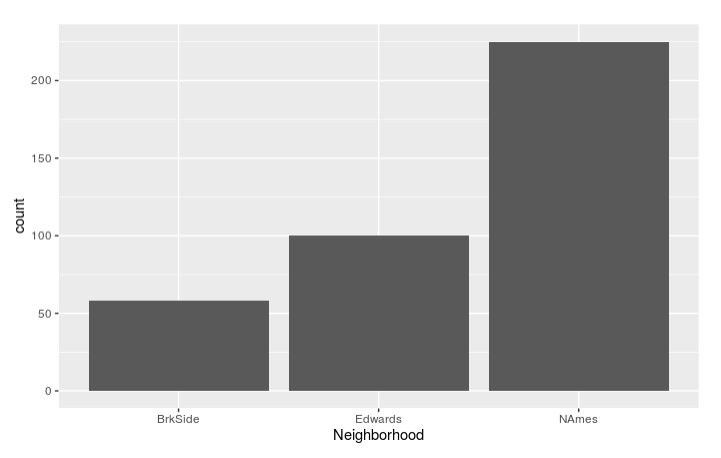
Plot 1, non-log transformed histogram of sale price



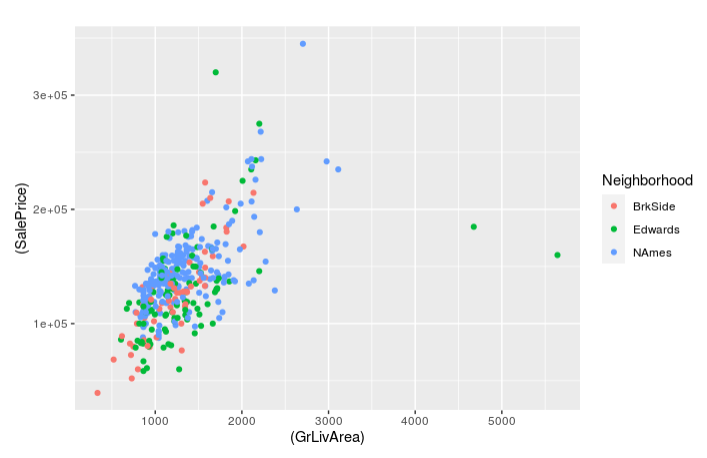
Plot 2, non-log transformed histogram of square footage



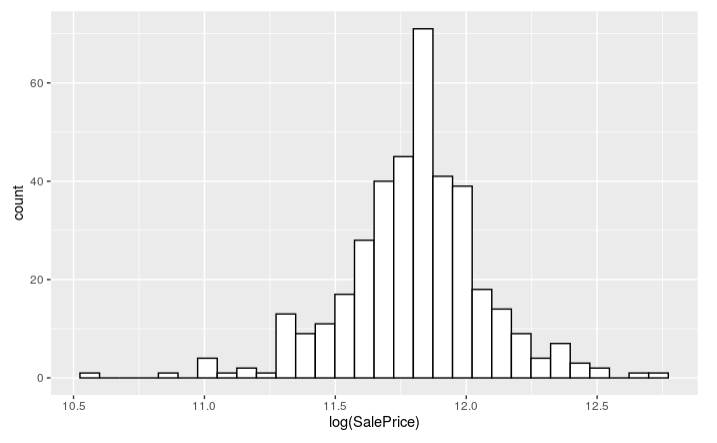
Plot 3, bar plot of house counts in each neighborhood



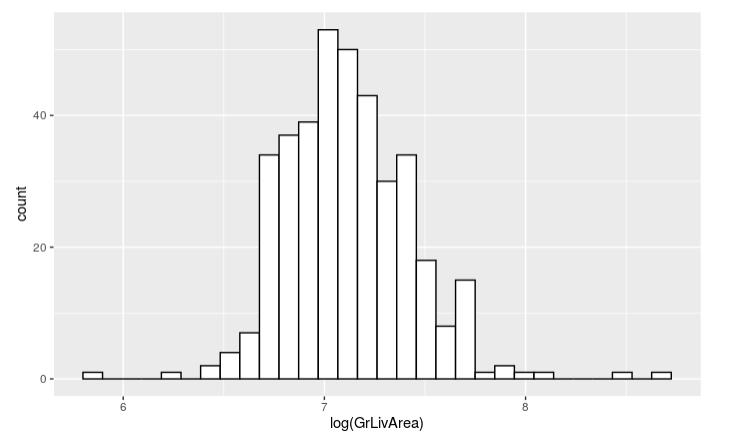
Plot 4, scatter plot of non-transformed data of sale price with neighborhood and square footage



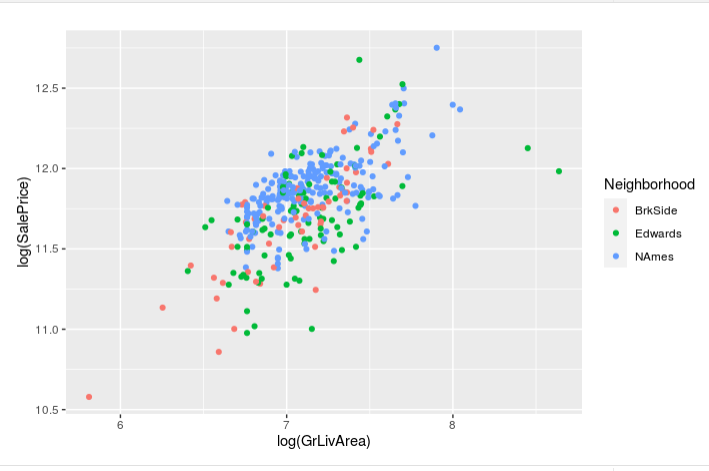
Plot 5, histogram of log-transformed data of house prices



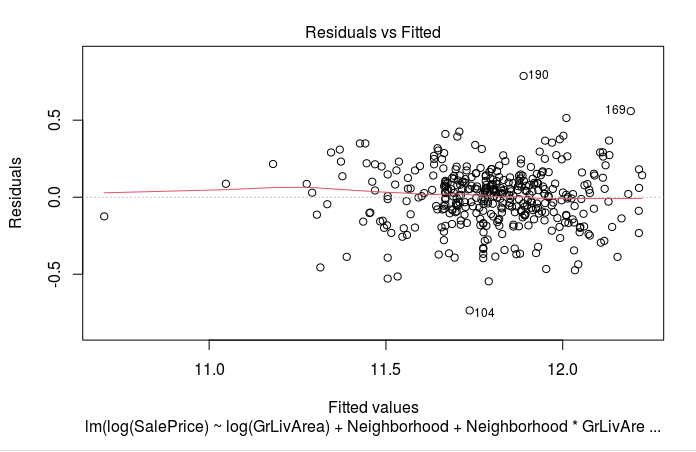
Plot 6, histogram of log-transformed square footage



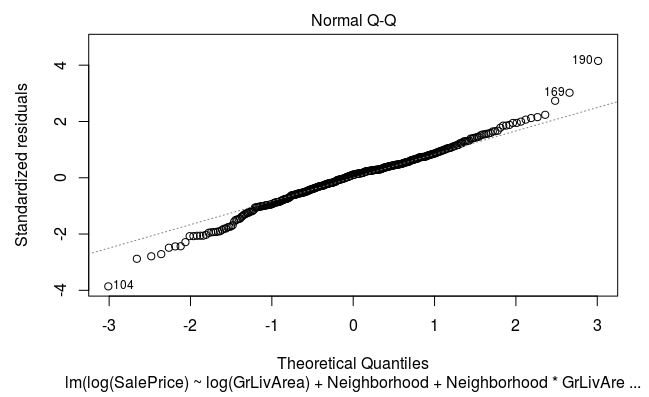
Plot 7, scatter plot of log-transformed data of sale price, neighborhood, and square footage



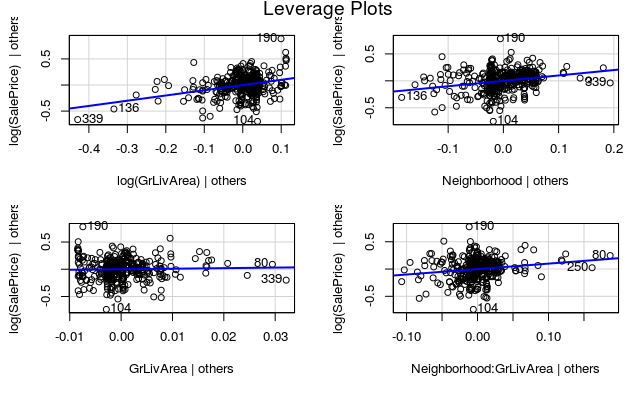
Plot 8, residual plot for LM model with outliers included



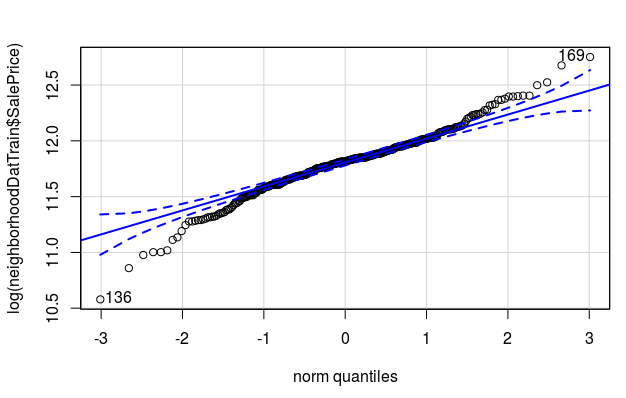
Plot 9, Quantile-Quantile plot with outliers included



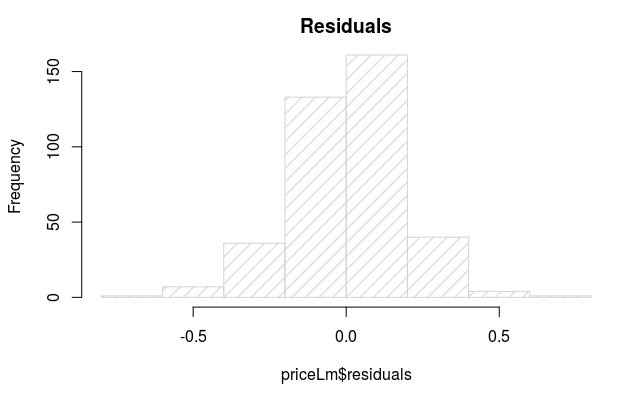
Plot 10, Leverage plots for model with outliers



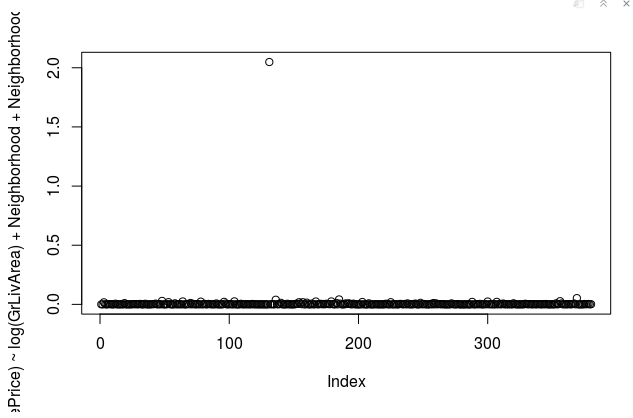
Plot 11, More detailed Quantile-Quantile plot with outliers included



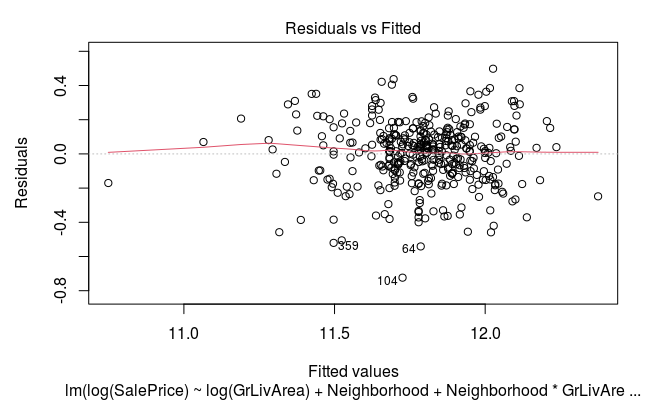
Plot 12, Residual Plot



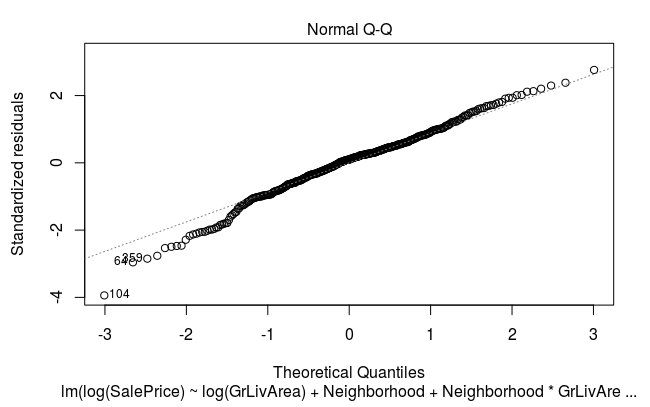
Plot 13, Cook’s Distance of non outlier removed data



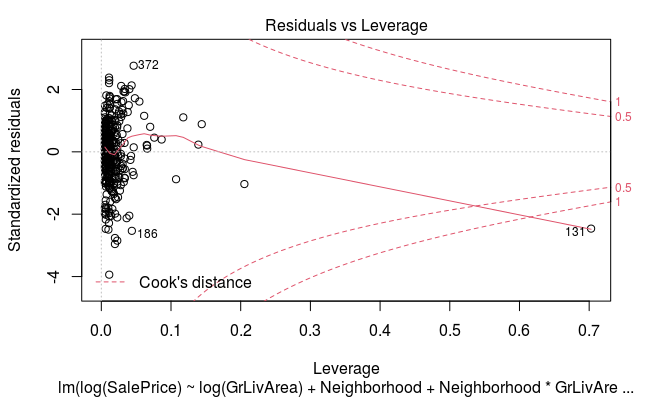
Plot 14, Residuals vs fitted on data with outliers removed



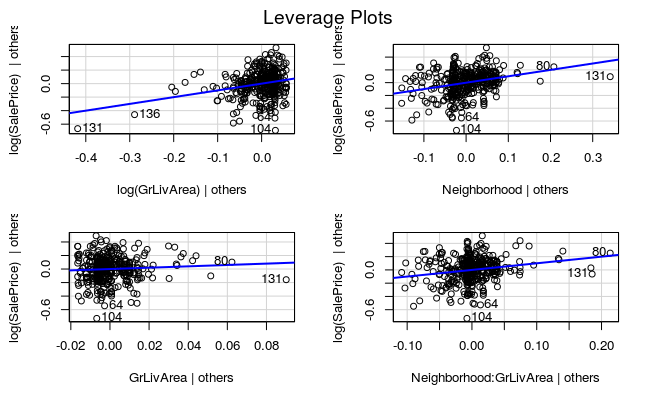
Plot 15, Quantile-Quantile plots of data with outliers removed



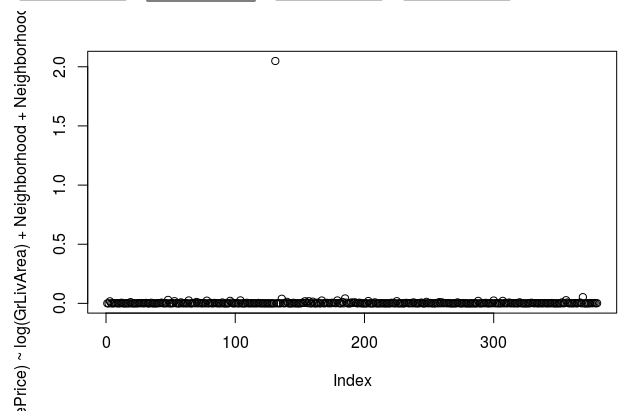
Plot 16, Cook’s distance plotted with data without outliers



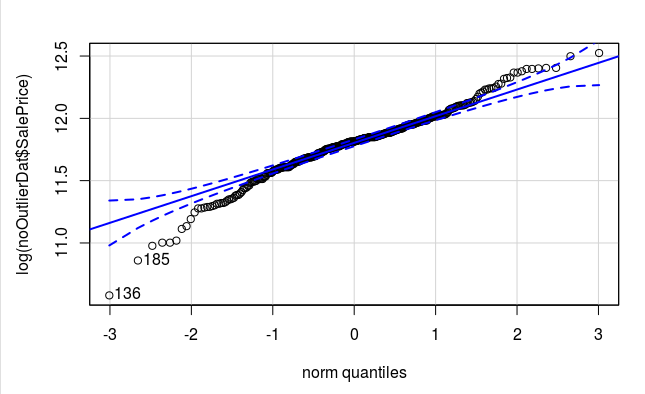
Plot 17, Leverage plots with outliers removed



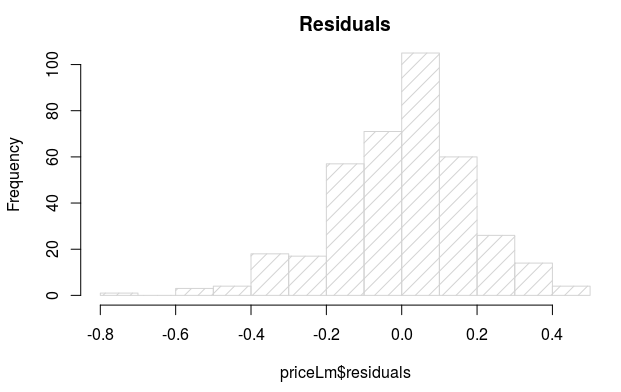
Plot 18, Cook’s distance (again) with outliers removed



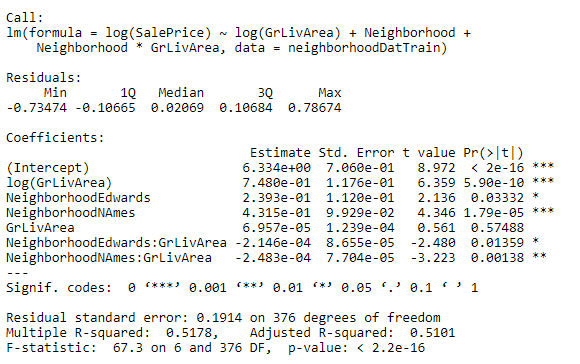
Plot 19, A more detailed quantile-quantile plot with outliers removed



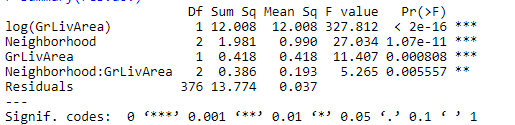
Plot 20, Residual plot of data with removed outliers



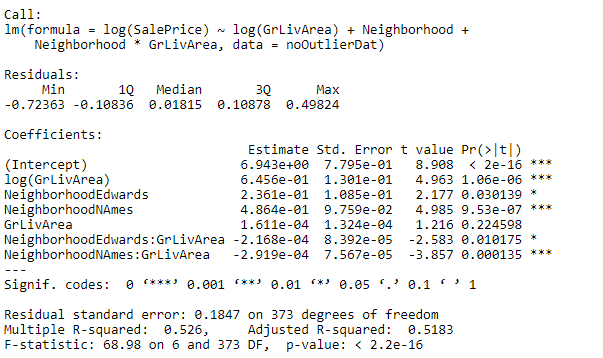
Plot 21, Model without outliers removed:



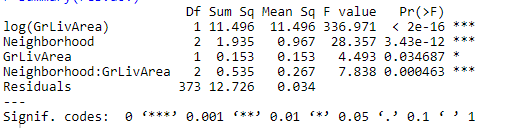
Plot 22, ANOVA table with outliers



Plot 23, Model with outliers removed

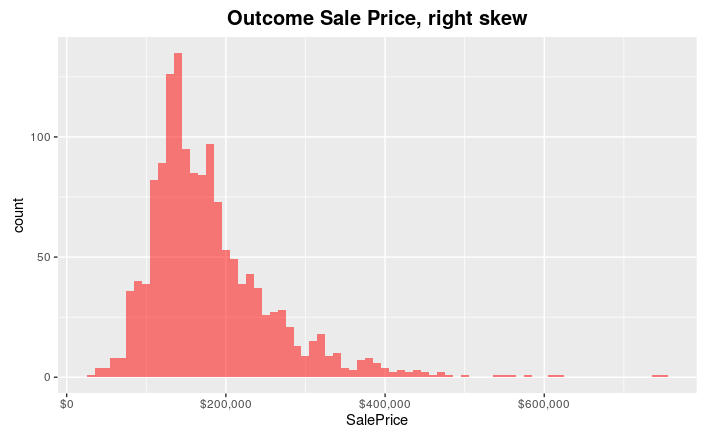


Plot 23, ANOVA table with outliers removed

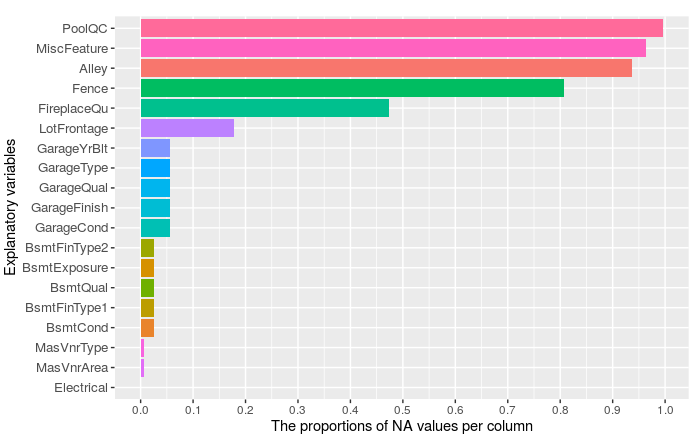


Analysis 2 Plots

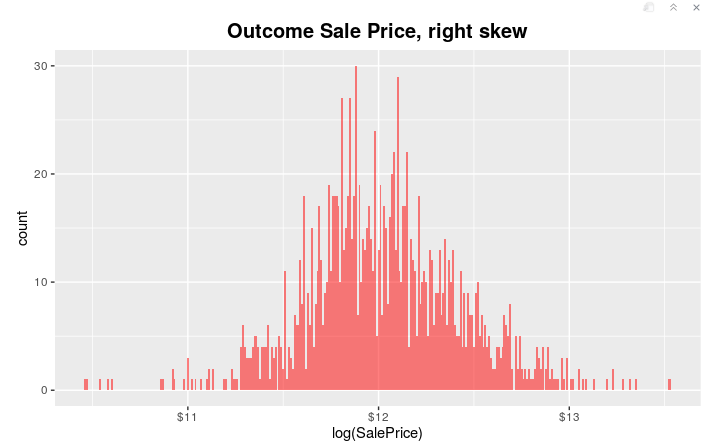
Plot 24, outcome Sale price



Plot 25, Missing data plot



Plot 26, log transformed Sale Price



Plot 27, Residuals Vs Fitted Step model

Chart, scatter chart

Description automatically generated

Plot 28, Quantile-Quantile Step model

Chart, line chart

Description automatically generated

Plot 29, Leverage Plots for step model

Diagram, engineering drawing

Description automatically generated

Plot 30, Cook’s Distance Step model

Chart, scatter chart

Description automatically generated

Plot 31, Residuals vs fitted, Forward Model

Chart, scatter chart

Description automatically generated

Plot 32, Quantile-Quantile forward model

Chart

Description automatically generated

Plot 33, Cook’s Distance forward model

Chart, scatter chart

Description automatically generated

Plot 34, Leverage plot forward model

Chart, scatter chart

Description automatically generated

Plot 35, Residuals vs Fitted backward model

Chart, scatter chart

Description automatically generated

Plot 36, Quantile-Quantile backward model

Chart, line chart

Description automatically generated

Plot 37, Leverage Plots backward model

Graphical user interface, diagram

Description automatically generated

Plot 38, Cook’s Distance backward model

Chart, scatter chart

Description automatically generated

Plot 39, Residuals vs fitted custom model

Chart, scatter chart

Description automatically generated

Plot 40, Quantile-Quantile plot custom model

Chart

Description automatically generated

Plot 41, Leverage Plots custom model

Diagram, arrow

Description automatically generated

Plot 42, Cook’s distance custom model

Chart, histogram

Description automatically generated

Code Appendix:

---

title: "Project 1 DS6371"

output:

html\_document:

df\_print: paged

---

```{r setup, include=FALSE}

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

```

```{r}

#Libaries

library(dplyr)

library(ggplot2)

library(car)

library(caret)

library(scales)

library(tidyr)

library(readr)

library(purrr)

library(forcats)

library(imputeMissings)

library(tidyverse)

library(leaps)

library(MASS)

library(olsrr)

library(asbio)

library(DAAG)

```

```{r}

#read in data

train <- read.csv("train.csv")

test <- read.csv("test.csv")

```

```{r}

#Select data

neighborhoodDatTrain <- dplyr::filter(train,Neighborhood =="Edwards" | Neighborhood =="NAmes" | Neighborhood == "BrkSide")

neighborhoodDatTrain <- neighborhoodDatTrain[,c(13,47,81)]

neighborhoodDatTest <- dplyr::filter(test,Neighborhood =="Edwards" | Neighborhood =="NAmes" | Neighborhood == "BrkSide")

neighborhoodDatTest <- neighborhoodDatTest[,c(13,47)]

neighborhoodDatTrain$Neighborhood <- as.factor(neighborhoodDatTrain$Neighborhood)

```

```{r}

#Some summary statistics on training data set

summary(train)

#Some summary statistics on training data set

summary(test)

#Some EDA on the training

# Basic histogram of prices

p<-ggplot(neighborhoodDatTrain, aes(x=SalePrice)) + geom\_histogram(color="black", fill="white")

p

# Basic histogram of sq footage

q<-ggplot(neighborhoodDatTrain, aes(x=GrLivArea)) + geom\_histogram(color="black", fill="white")

q

# Bar plot of neighborhoods

ggplot(neighborhoodDatTrain) + geom\_bar(aes(x = Neighborhood))

#Scatter plots

r <-ggplot(neighborhoodDatTrain, aes(x = (GrLivArea), y = (SalePrice),color=Neighborhood)) +

geom\_point()

r

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

#Log transform

# Basic histogram of prices

p<-ggplot(neighborhoodDatTrain, aes(x=log(SalePrice))) + geom\_histogram(color="black", fill="white")

p

# Basic histogram of sq footage

q<-ggplot(neighborhoodDatTrain, aes(x=log(GrLivArea))) + geom\_histogram(color="black", fill="white")

q

#Scatter plots

r <-ggplot(neighborhoodDatTrain, aes(x = log(GrLivArea), y = log(SalePrice),color=Neighborhood)) +

geom\_point()

r

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

#Create lm with outliers

priceLm <- lm(log(SalePrice)~log(GrLivArea)+Neighborhood+Neighborhood\*GrLivArea,data=neighborhoodDatTrain)

summary(priceLm)

confint(priceLm)

#Look at ANOVA

res.aov <- aov(log(SalePrice)~log(GrLivArea)+Neighborhood+Neighborhood\*GrLivArea, data = neighborhoodDatTrain)

summary(res.aov)

plot(priceLm)

#Model diagnostics, leverage plots

#leveragePlots(lm(log(SalePrice)~log(GrLivArea)+Neighborhood+Neighborhood\*GrLivArea,data=neighborhoodDatTrain))

#Model diagnostics, Cook's Distance

#plot(cooks.distance(lm(log(SalePrice)~log(GrLivArea)+Neighborhood+Neighborhood\*GrLivArea,data=neighborhoodDatTrain)))

#sort(cooks.distance(lm(log(SalePrice)~log(GrLivArea)+Neighborhood+Neighborhood\*GrLivArea,data=neighborhoodDatTrain)),decreasing = TRUE)

#Model diagnostics, Quantile-Quantile plot

qqPlot(log(neighborhoodDatTrain$SalePrice))

#Hist of residuals

h <- hist(priceLm$residuals, breaks = 10, density = 10,col = "lightgray",main = "Residuals")

xfit <- seq(min(priceLm$residuals), max(priceLm$residuals), length = 40)

yfit <- dnorm(xfit, mean = mean(priceLm$residuals), sd = sd(priceLm$residuals))

yfit <- yfit \* diff(h$mids[1:2]) \* length(priceLm$residuals)

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

#Create lm without outliers

noOutlierDat <- neighborhoodDatTrain[-c(169,190,339),]

priceLm <- lm(log(SalePrice)~log(GrLivArea)+Neighborhood+Neighborhood\*GrLivArea,data=noOutlierDat)

summary(priceLm)

confint(priceLm)

#Internal CV

trainIndex <- createDataPartition(neighborhoodDatTrain$SalePrice, p = .8, list = FALSE, times = 1)

head(trainIndex)

saleTrain <- neighborhoodDatTrain[ trainIndex,]

saleTest <- neighborhoodDatTrain[-trainIndex,]

pred.w.plim <- predict(lm(log(SalePrice)~log(GrLivArea)+Neighborhood+Neighborhood\*GrLivArea,data=saleTrain), saleTest, interval = "prediction")

pred.w.clim <- predict(lm(log(SalePrice)~log(GrLivArea)+Neighborhood+Neighborhood\*GrLivArea,data=saleTrain), saleTest, interval = "confidence")

#View(exp(pred.w.plim))

matplot(saleTest$SalePrice, cbind(pred.w.clim, pred.w.plim[,-1]),

lty = c(1,2,2,3,3), type = "l", ylab = "predicted y")

#predict test data set

pred.w.plim <- predict(priceLm, neighborhoodDatTest, interval = "prediction")

#View(exp(pred.w.plim))

#View(neighborhoodDatTrain$SalePrice)

#Look at ANOVA

res.aov <- aov(log(SalePrice)~log(GrLivArea)+Neighborhood+Neighborhood\*GrLivArea, data = noOutlierDat)

summary(res.aov)

plot(priceLm)

#Model diagnostics, leverage plots

#leveragePlots(lm(log(SalePrice)~log(GrLivArea)+Neighborhood+Neighborhood\*GrLivArea,data=noOutlierDat))

#Model diagnostics, Cook's Distance

#plot(cooks.distance(lm(log(SalePrice)~log(GrLivArea)+Neighborhood+Neighborhood\*GrLivArea,data=noOutlierDat)))

#sort(cooks.distance(lm(log(SalePrice)~log(GrLivArea)+Neighborhood+Neighborhood\*GrLivArea,data=noOutlierDat)),decreasing = TRUE)

#Model diagnostics, Quantile-Quantile plot

qqPlot(log(noOutlierDat$SalePrice))

#Hist of residuals

h <- hist(priceLm$residuals, breaks = 10, density = 10,col = "lightgray",main = "Residuals")

xfit <- seq(min(priceLm$residuals), max(priceLm$residuals), length = 40)

yfit <- dnorm(xfit, mean = mean(priceLm$residuals), sd = sd(priceLm$residuals))

yfit <- yfit \* diff(h$mids[1:2]) \* length(priceLm$residuals)

CVdat <- CVlm(data = neighborhoodDatTrain, form.lm = formula(log(SalePrice)~log(GrLivArea)+Neighborhood+Neighborhood\*GrLivArea),

m = 3, dots = FALSE, seed = 29, plotit = c("Observed","Residual"),

main="Small symbols show cross-validation predicted values",

legend.pos="topleft", printit = TRUE)

CVdat

(press(priceLm))

```

```{r}

####Analysis 2####

#read in data

analysis2Train <- read.csv("train.csv")

test <- read.csv("test.csv")

```

```{r}

#Look at outcome variable

ggplot(data = analysis2Train %>% filter(!is.na(log(SalePrice)))) +

geom\_histogram(aes(x = log(SalePrice)), fill = "red", alpha = 1/2, binwidth = 0.01) +

scale\_x\_continuous(labels = dollar\_format()) +

labs(

title = "Outcome Sale Price, right skew"

) +

theme(

plot.title = element\_text(hjust = 0.5, size = 15, face = "bold"),

)

#Look at missing data

na\_prop <- analysis2Train %>%

dplyr::select(-SalePrice) %>%

map(is.na) %>%

map\_dfr(mean) %>%

pivot\_longer(cols = everything(), names\_to = "variables", values\_to = "prop") %>%

filter(prop > 0) %>%

arrange(desc(prop))

na\_prop %>%

ggplot(aes(x = fct\_reorder(variables, prop), y = prop, fill = variables)) +

geom\_bar(stat = "identity") +

coord\_flip() +

theme(legend.position = "none") +

labs(

x = "Explanatory variables",

y = "The proportions of NA values per column"

) +

scale\_y\_continuous(breaks = seq(0, 1, by = 0.1)) +

theme(axis.text.y = element\_text(size = 10))

analysis2Train$PoolQC[is.na(analysis2Train$PoolQC)] <- "None"

analysis2Train$MiscFeature[is.na(analysis2Train$MiscFeature)] <- "None"

analysis2Train$Alley[is.na(analysis2Train$Alley)] <- "No"

analysis2Train$Fence[is.na(analysis2Train$Fence)] <- "No"

analysis2Train$FireplaceQu[is.na(analysis2Train$FireplaceQu)] <- "No"

analysis2Train$GarageType[is.na(analysis2Train$GarageType)] <- "No"

analysis2Train$GarageFinish[is.na(analysis2Train$GarageFinish)] <- "No"

analysis2Train$GarageQual[is.na(analysis2Train$GarageQual)] <- "No"

analysis2Train$GarageCond[is.na(analysis2Train$GarageCond)] <- "No"

analysis2Train$BsmtExposure[is.na(analysis2Train$BsmtExposure)] <- "NoBs"

analysis2Train$BsmtCond[is.na(analysis2Train$BsmtCond)] <- "NoBs"

analysis2Train$BsmtQual[is.na(analysis2Train$BsmtQual)] <- "NoBs"

analysis2Train$BsmtFinType1[is.na(analysis2Train$BsmtFinType1)] <- "NoBs"

analysis2Train$BsmtFinType2[is.na(analysis2Train$BsmtFinType2)] <- "NoBs"

# To specify the levels of ordered factors

PoolQC\_lev <- c("None", "Fa", "TA", "Gd", "Ex")

Fence\_lev <- c("No", "MnWw", "GdWo", "MnPrv", "GdPrv")

FireplaceQu\_lev <- c("No", "Po", "Fa", "TA", "Gd", "Ex")

GarageFinish\_lev <- c("No", "Unf", "RFn", "Fin")

GarageQual\_lev <- c("No", "Po", "Fa", "TA", "Gd", "Ex")

GarageCond\_lev <- c("No", "Po", "Fa", "TA", "Gd", "Ex")

BsmtExposure\_lev <- c("NoBs", "No", "Mn", "Av", "Gd")

BsmtCond\_lev <- c("NoBs", "Po", "Fa", "TA", "Gd", "Ex")

BsmtQual\_lev <- c("NoBs", "Po", "Fa", "TA", "Gd", "Ex")

BsmtFinType1\_lev <- c("NoBs", "Unf", "LwQ", "Rec", "BLQ", "ALQ", "GLQ")

BsmtFinType2\_lev <- c("NoBs", "Unf", "LwQ", "Rec", "BLQ", "ALQ", "GLQ")

analysis2Train2 <- analysis2Train %>%

mutate(PoolQC = parse\_factor(PoolQC, levels = PoolQC\_lev, ordered = TRUE),

MiscFeature = parse\_factor(MiscFeature),

Alley = parse\_factor(Alley),

Fence = parse\_factor(Fence, levels = Fence\_lev, ordered = TRUE),

FireplaceQu = parse\_factor(FireplaceQu, levels = FireplaceQu\_lev, ordered = TRUE),

GarageType = parse\_factor(GarageType),

GarageFinish = parse\_factor(GarageFinish, levels = GarageFinish\_lev, ordered = TRUE),

GarageQual = parse\_factor(GarageQual, levels = GarageQual\_lev, ordered = TRUE),

GarageCond = parse\_factor(GarageCond, levels = GarageCond\_lev, ordered = TRUE),

BsmtExposure = parse\_factor(BsmtExposure, levels = BsmtExposure\_lev, ordered = TRUE),

BsmtCond = parse\_factor(BsmtCond, levels = BsmtCond\_lev, ordered = TRUE),

BsmtQual = parse\_factor(BsmtQual, levels = BsmtQual\_lev, ordered = TRUE),

BsmtFinType1 = parse\_factor(BsmtFinType1, levels = BsmtFinType1\_lev, ordered = TRUE),

BsmtFinType2 = parse\_factor(BsmtFinType2, levels = BsmtFinType2\_lev, ordered = TRUE))

#lets impute some data

#Col 4 lot frontage

analysis2Train2[,4][is.na(analysis2Train2[,4])] <- round(mean(analysis2Train2[,4], na.rm = TRUE))

#Col 27, massvnr

analysis2Train2[,27][is.na(analysis2Train2[,27])] <- round(mean(analysis2Train2[,27], na.rm = TRUE))

#Col 60, Garage year built

analysis2Train2[,60][is.na(analysis2Train2[,60])] <- round(mean(analysis2Train2[,60], na.rm = TRUE))

#Col 26, MasVnrType

analysis2Train2$MasVnrType <- analysis2Train2$MasVnrType %>% tidyr::replace\_na("Stone")

#Col 43, electrical

analysis2Train2$Electrical <- analysis2Train2$Electrical %>% tidyr::replace\_na("SBrkr ")

analysis2Train2$GrLivArea <- log(analysis2Train2$GrLivArea)

#sort(is.na(analysis2Train2),decreasing = TRUE)

#analysis2Train2[!complete.cases(analysis2Train2),]

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

#test dat

#Look at missing data

na\_prop <- test %>%

map(is.na) %>%

map\_dfr(mean) %>%

pivot\_longer(cols = everything(), names\_to = "variables", values\_to = "prop") %>%

filter(prop > 0) %>%

arrange(desc(prop))

na\_prop %>%

ggplot(aes(x = fct\_reorder(variables, prop), y = prop, fill = variables)) +

geom\_bar(stat = "identity") +

coord\_flip() +

theme(legend.position = "none") +

labs(

x = "Explanatory variables",

y = "The proportions of NA values per column"

) +

scale\_y\_continuous(breaks = seq(0, 1, by = 0.1)) +

theme(axis.text.y = element\_text(size = 10))

test$PoolQC[is.na(test$PoolQC)] <- "None"

test$MiscFeature[is.na(test$MiscFeature)] <- "None"

test$Alley[is.na(test$Alley)] <- "No"

test$Fence[is.na(test$Fence)] <- "No"

test$FireplaceQu[is.na(test$FireplaceQu)] <- "No"

test$GarageType[is.na(test$GarageType)] <- "No"

test$GarageFinish[is.na(test$GarageFinish)] <- "No"

test$GarageQual[is.na(test$GarageQual)] <- "No"

test$GarageCond[is.na(test$GarageCond)] <- "No"

test$BsmtExposure[is.na(test$BsmtExposure)] <- "NoBs"

test$BsmtCond[is.na(test$BsmtCond)] <- "NoBs"

test$BsmtQual[is.na(test$BsmtQual)] <- "NoBs"

test$BsmtFinType1[is.na(test$BsmtFinType1)] <- "NoBs"

test$BsmtFinType2[is.na(test$BsmtFinType2)] <- "NoBs"

# To specify the levels of ordered factors

PoolQC\_lev <- c("None", "Fa", "TA", "Gd", "Ex")

Fence\_lev <- c("No", "MnWw", "GdWo", "MnPrv", "GdPrv")

FireplaceQu\_lev <- c("No", "Po", "Fa", "TA", "Gd", "Ex")

GarageFinish\_lev <- c("No", "Unf", "RFn", "Fin")

GarageQual\_lev <- c("No", "Po", "Fa", "TA", "Gd", "Ex")

GarageCond\_lev <- c("No", "Po", "Fa", "TA", "Gd", "Ex")

BsmtExposure\_lev <- c("NoBs", "No", "Mn", "Av", "Gd")

BsmtCond\_lev <- c("NoBs", "Po", "Fa", "TA", "Gd", "Ex")

BsmtQual\_lev <- c("NoBs", "Po", "Fa", "TA", "Gd", "Ex")

BsmtFinType1\_lev <- c("NoBs", "Unf", "LwQ", "Rec", "BLQ", "ALQ", "GLQ")

BsmtFinType2\_lev <- c("NoBs", "Unf", "LwQ", "Rec", "BLQ", "ALQ", "GLQ")

test2 <- test %>%

mutate(PoolQC = parse\_factor(PoolQC, levels = PoolQC\_lev, ordered = TRUE),

MiscFeature = parse\_factor(MiscFeature),

Alley = parse\_factor(Alley),

Fence = parse\_factor(Fence, levels = Fence\_lev, ordered = TRUE),

FireplaceQu = parse\_factor(FireplaceQu, levels = FireplaceQu\_lev, ordered = TRUE),

GarageType = parse\_factor(GarageType),

GarageFinish = parse\_factor(GarageFinish, levels = GarageFinish\_lev, ordered = TRUE),

GarageQual = parse\_factor(GarageQual, levels = GarageQual\_lev, ordered = TRUE),

GarageCond = parse\_factor(GarageCond, levels = GarageCond\_lev, ordered = TRUE),

BsmtExposure = parse\_factor(BsmtExposure, levels = BsmtExposure\_lev, ordered = TRUE),

BsmtCond = parse\_factor(BsmtCond, levels = BsmtCond\_lev, ordered = TRUE),

BsmtQual = parse\_factor(BsmtQual, levels = BsmtQual\_lev, ordered = TRUE),

BsmtFinType1 = parse\_factor(BsmtFinType1, levels = BsmtFinType1\_lev, ordered = TRUE),

BsmtFinType2 = parse\_factor(BsmtFinType2, levels = BsmtFinType2\_lev, ordered = TRUE))

#lets impute some data

#Col 4 lot frontage

test2[,4][is.na(test2[,4])] <- round(mean(test2[,4], na.rm = TRUE))

#Col 27, massvnr

test2[,27][is.na(test2[,27])] <- round(mean(test2[,27], na.rm = TRUE))

#Col 60, Garage year built

test2[,60][is.na(test2[,60])] <- round(mean(test2[,60], na.rm = TRUE))

#Col 26, MasVnrType

test2$MasVnrType <- test2$MasVnrType %>% tidyr::replace\_na("Stone")

#Col 43, electrical

test2$Electrical <- test2$Electrical %>% tidyr::replace\_na("SBrkr ")

test2$GrLivArea <- log(test2$GrLivArea)

```

```{r}

####Build Models####

#Full model

full.model <- lm(log(SalePrice)~.,data = analysis2Train2)

#Stepwise model

step.model <- stepAIC(full.model,direction = "both",trace = FALSE)

#Get model summary

step.model$pred

models <- regsubsets(log(SalePrice)~., data = analysis2Train2, nvmax = 1,

method = "seqrep")

summary(models)

#Set seed for reproducibility

set.seed(123)

# Set up repeated k-fold cross-validation

train.control <- trainControl(method = "cv", number = 10)

# Train the model

step.model <- train(log(SalePrice) ~., data = analysis2Train2,

method = "leapBackward",

tuneGrid = data.frame(nvmax = 1:75),

trControl = train.control

)

step.model$results

step.model$bestTune

stepLm <- lm(log(SalePrice)~OverallQual+OverallCond+YearBuilt+BsmtFinType2+KitchenAbvGr+GarageCond,data = analysis2Train2)

summary(stepLm)

summary(step.model$finalModel)

coef(step.model$finalModel, 6)

press(stepLm)

plot(stepLm)

#Model diagnostics, leverage plots

#leveragePlots(stepLm,data=analysis2Train2)

#Model diagnostics, Cook's Distance

#plot(cooks.distance(stepLm,data=analysis2Train2))

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

min.model = lm(log(SalePrice) ~ 1, data=analysis2Train2)

biggest <- formula(lm(log(SalePrice)~.,analysis2Train2))

biggest

fwd.model = step(min.model, direction='forward', scope=biggest)

summary(fwd.model)

forwardlm <- lm(log(SalePrice)~GrLivArea+Neighborhood+GarageCars+OverallCond+HouseStyle+YearBuilt+RoofMatl+BsmtFinSF1+MSZoning+Functional+Condition1+SaleCondition+KitchenQual+LotArea+Condition1+Exterior1st+ScreenPorch+Heating+LandSlope+WoodDeckSF+TotalBsmtSF+LotConfig+CentralAir+GarageQual+BsmtFullBath+Fireplaces+X2ndFlrSF+YearRemodAdd+GarageArea+Foundation+LotFrontage+KitchenAbvGr+GarageCond+SaleType+ExterCond+Street+HalfBath,data = analysis2Train2)

summary(forwardlm)

plot(forwardlm)

#Model diagnostics, leverage plots

#leveragePlots(forwardlm,data=analysis2Train2)

#Model diagnostics, Cook's Distance

#plot(cooks.distance(forwardlm,data=analysis2Train2))

#predict test data set

forward.lm.plim <- predict(forwardlm, test2, interval = "prediction")

####Write forward model

forward.lm.plim <- forward.lm.plim[,1]

forward.lm.plim <- as.data.frame(forward.lm.plim)

forward.lm.plim <- exp(forward.lm.plim[,1])

forward.lm.plim <- forward.lm.plim %>% rename(SalePrice = forward.lm.plim,)

forward.lm.plim[,1][is.na(forward.lm.plim[,1])] <- round(mean(forward.lm.plim[,1], na.rm = TRUE))

out <- write.csv(forward.lm.plim,"forwardModel.csv")

####Write Step model

stepWise.lm.plim <- predict(step.model, test2, interval = "prediction")

stepWise.lm.plim <- exp(stepWise.lm.plim)

stepWise.lm.plim <- as.data.frame(stepWise.lm.plim)

stepWise.lm.plim <- stepWise.lm.plim %>% rename(SalePrice = stepWise.lm.plim,)

stepWise.lm.plim$ID <- seq.int(nrow(stepWise.lm.plim))

outStepwise <- write.csv(stepWise.lm.plim,"stepwiseModel.csv")

#View(stepWise.lm.plim)

# Set seed for reproducibility

set.seed(123)

# Set up repeated k-fold cross-validation

train.control <- trainControl(method = "cv", number = 10)

# Train the model

step.model <- train(log(SalePrice) ~., data = analysis2Train2,

method = "leapBackward",

tuneGrid = data.frame(nvmax = 1:10),

trControl = train.control

)

step.model$results

summary(step.model$finalModel)

coef(step.model$finalModel, 7)

backlm <- lm(log(SalePrice)~OverallQual+OverallCond+YearBuilt+RoofMatl+BsmtFinType2+KitchenAbvGr+GarageCond,data = analysis2Train2)

summary(backlm)

#predict test data set

back.lm.plim <- predict(backlm, test2, interval = "prediction")

back.lm.plim <- back.lm.plim[,1]

back.lm.plim <- exp(back.lm.plim)

back.lm.plim <- as.data.frame(back.lm.plim)

back.lm.plim <- back.lm.plim %>% rename(SalePrice = back.lm.plim,)

back.lm.plim$ID <- seq.int(nrow(back.lm.plim))

outBack <- write.csv(backModel,"backModel1.csv")

#View(back.lm.plim)

#Model diagnostics, leverage plots

#leveragePlots(backlm,data=analysis2Train2)

#Model diagnostics, Cook's Distance

#plot(cooks.distance(backlm,data=analysis2Train2))

#plot(backlm)

#####Custom model####

customlm <- lm(log(SalePrice)~OverallQual+log(GrLivArea)+Neighborhood+GarageCars+ExterQual+TotalBsmtSF+GarageArea+KitchenQual+YearBuilt,data=analysis2Train2)

summary(customlm)

custom.lm.plim <- predict(customlm, test2, interval = "prediction")

custom.lm.plim <- custom.lm.plim[,1]

custom.lm.plim <- exp(custom.lm.plim)

custom.lm.plim <- as.data.frame(custom.lm.plim)

custom.lm.plim <- custom.lm.plim %>% rename(SalePrice = custom.lm.plim,)

custom.lm.plim$ID <- seq.int(nrow(custom.lm.plim))

outCustom <- write.csv(custom.lm.plim,"customModel1.csv")

#Model diagnostics, leverage plots

#leveragePlots(customlm,data=analysis2Train2)

#Model diagnostics, Cook's Distance

#plot(cooks.distance(customlm,data=analysis2Train2))

#plot(customlm)

#View(custom.lm.plim)

```