Math 444 Kaggle Housing project

Jose
Nicolas Arenas River
Chance Vandergeugten
Prof Zambom

Problem:

In this report, we aim to predict the price of houses in an undisclosed housing market using predictor variables provided to us in a dataset as well as various methods of viewing and altering our data in R. To ensure that check as many bases as possible, as a group we decided to each make separate models, and see which model scored the highest (according to kaggles scoring system.)

Methods:

Initially, we noticed that within the csv we used to create our models, there were many cases of missing data for predictors. Because of this, we had the idea to impute the missing data using the MICE package within R. To do this, we changed all character values within our models to factors. Once this happened, we separated categorical and numerical values in order to impute differently for each type of value. With this imputed data set complete, we were able to use a more complex model that included every predictor available in our model. To start the model off, a model was created with predictors that are known for increasing the value of a house significantly. These predictors included lot area, pool area, overall condition, quality, and others. Initially, the model looked as such:

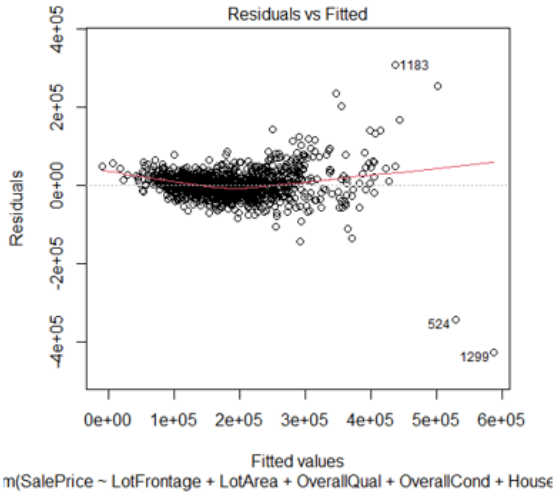
Full_model <- lm(SalePrice ~ MSZoning + LotFrontage + LotArea + OverallQual + OverallCond + Neighborhood + HouseStyle + ExterQual + ExterCond + X1stFlrSF + X2ndFlrSF + FullBath + HalfBath + BedroomAbvGr + KitchenAbvGr + Functional + GarageType + GarageCars + GarageArea + GarageQual + GarageCond + OpenPorchSF + EnclosedPorch + PoolArea + Fence + MiscFeature + MiscVal + SaleCondition).

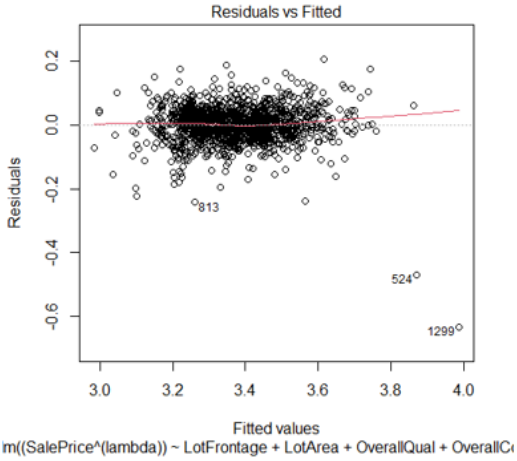
After starting with this model, we used the SignifReg Package to remove any predictors that were deemed unnecessary via adjusted r-squared as our method of comparison. After doing this our model looked like:

lm(formula = SalePrice ~ MSZoning + LotFrontage + LotArea + OverallQual + OverallCond + Neighborhood + HouseStyle + ExterQual + ExterCond + X1stFlrSF + X2ndFlrSF + FullBath + HalfBath + BedroomAbvGr + KitchenAbvGr + Functional + GarageType + GarageCars + EnclosedPorch + PoolArea + Fence + MiscFeature + SaleCondition).

From here, multicollinearity in the model was checked for, where we realized 2 predictors with high variance inflation factors: MSZoning and Neighborhood. After removing these, we looked at the residuals to see if the model was a good fit. When looking at the residuals we saw that they were heteroscedastic, which is bad, to fix this, we used a box-cox transformation to fit a better fitting model. To do this, we used the MASS package in R to use the boxcox command in order to find our optimal λ (lambda) to transform our model (in this case it was (0.10101). When doing

this, we found our residuals now transformed as such:





As we can see, our residual plot looks much better than before the transformation, as the residuals are now homoscedastic. With this, we can say that we had a good looking model. Thus, our final model looked as such:

final _modell <- lm((SalePrice ^ (lambda)) ~ LotFrontage + LotArea + OverallQual + OverallCond + HouseStyle + ExterQual + ExterCond + X1stFlrSF + X2ndFlrSF + FullBath + HalfBath + BedroomAbvGr + KitchenAbvGr + Functional+ GarageType + GarageCars + EnclosedPorch + PoolArea + Fence + MiscFeature + SaleCondition)

With this model chosen, we achieved a score of 0.8306, which ranked us at 4893. Seeing ourresult, we decided to try a different approach to improve our model.

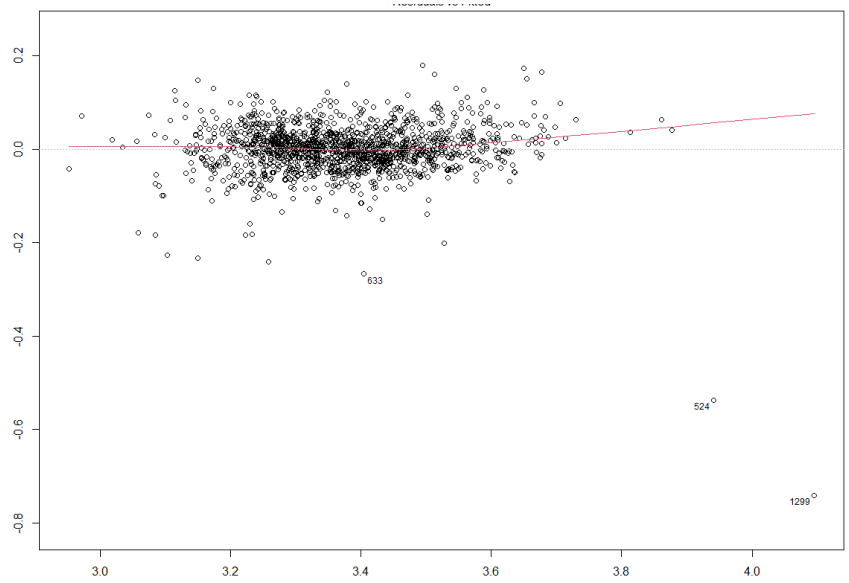
From here, we decided to approach the dataset differently. We did use the SignifReg package again, however we decided to build it from a null model, based on the criteria of finding the best AIC value, that is the model with the smallest AIC value. Doing so gave us a model that looked as such:

SalePrice ~ OverallQual + GrLivArea + BsmtFinSF1 + GarageCars +
MSSubClass + YearBuilt + BedroomAbvGr + OverallCond + LotArea +
MasVnrArea + BsmtFullBath + TotRmsAbvGrd + WoodDeckSF + ScreenPorch +
TotalBsmtSF + YearRemodAdd + KitchenAbvGr + Condition2 +
Fireplaces + FullBath + LandSlope + Neighborhood + Utilities +
LotShape + LandContour

From here, we reduced the model by removing predictors that were deemed insignificant by the summary of our model, up until we were met with a reduced model that looked like this:

SalePrice ~ OverallQual + GrLivArea + BsmtFinSF1 + GarageCars + MSSubClass
+ YearBuilt + BedroomAbvGr + OverallCond + LotArea + MasVnrArea
+ BsmtFullBath

Again, when we plotted the residuals of this model, we found that there seemed to be a parabolic relationship, which was not desired. To combat this, we decided to use a boxcox transformation again, where we found our lambda was 0.1010101. After doing this and looking at our residuals, we find that it looked as such:



We see that this is much more desirable, and that it increased the adjusted r-squared of our model by approximately 5%. (.8018 -> .8533) From this point, we were able to make predictions based on our model, and found that some values were marked as NA. As a result, we found the mean of all available predictions of sale prices, and filled in the NA values with said mean. By doing this, we were able to submit a model that received a score of .15727, which was good for a ranking of 3285. However, we still felt we could produce a better model than this, and we attempted to do as such.

In an attempt to create a more optimal model, we again use the SignifReg package to create a base model from all the non-categorical, non-NA variables from the given train data set. Instead of an AIC based model, like we made previously, we used a model that was based off an adjusted r-squared criterion, and we built the model from the ground up using a null model as well as the "forward" command in SignifReg. After this model was built, we had to determine which dummy variables would prove to be useful in our model. To do this, we used the "unique" function in R to examine non-categorical variables. To keep a simple model, we excluded categorical variables with more than 4 possible choices, and we did not consider variables with NA either. After we sorted and chose our categorical variables, we ran the SignifReg command again, except in "both" directions with the same criterion of adjusted r-squared. However, this time the scope was expanded to include our newly chosen categorical variables. This would result in a fully built model that included both numerical and categorical predictors. Once again, a boxcox transformation was applied to our response variable in order to correct some diagnostics and improve our r-adj value. This would result in the final model used to make our

predictions. Once we made our predictions , there were 4 predicted v values that came out as NA in the csv file. Again, we estimated these values using the mean of all the predictions, and we finally had a complete prediction file ready for submission.

Results:

Ultimately, our final model looked as such:

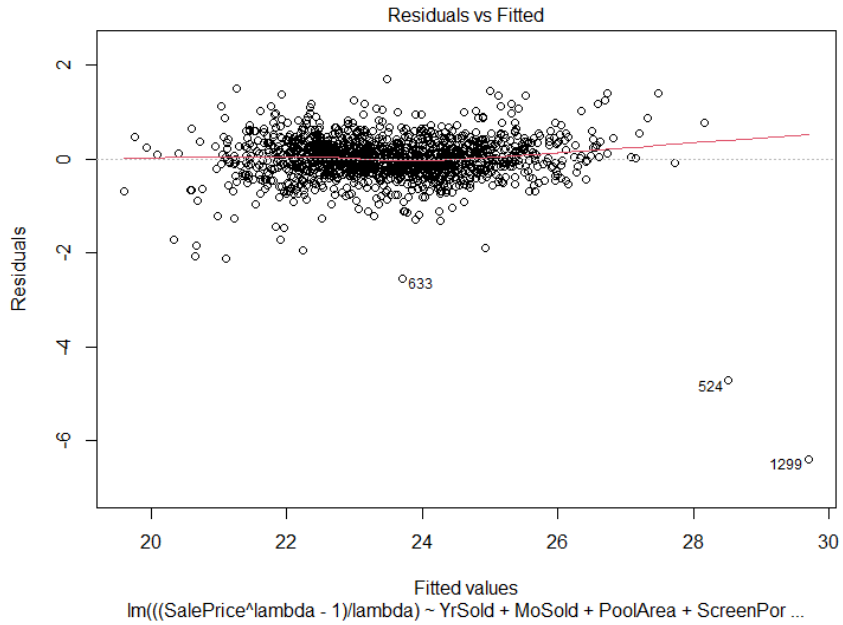
((Saleprice^Lambda-1) / Lambda)YrSold + MoSold + PoolArea + ScreenPorch + X3SsnPorch + EnclosedPorch + OpenPorchSF + WoodDeckSF + PavedDrive + GarageArea + GarageCars + Fireplaces + TotRmsAbvGrd + KitchenAbvGr + BedroomAbvGr + HalfBath + FullBath + BsmtHalfBath + BsmtFullBath + GrLivArea + LowQualFinSF + X2ndFlrSF + X1stFlrSF + TotalBsmtSF + BsmtUnfSF + BsmtFinSF1 + BsmtFinSF2 + YearRemodAdd + YearBuilt + OverallCond + OverallQual + LotArea + MSSubClass + Street + LandContour + LandSlope + ExterQual + CentralAir + KitchenQual + PavedDrive)

Where we found our lambda value was equal to 0.1010101. Additionally, we have our summary, which revealed that this model had an r-adj value of 0.8734. This was higher than the previous model by 2%!

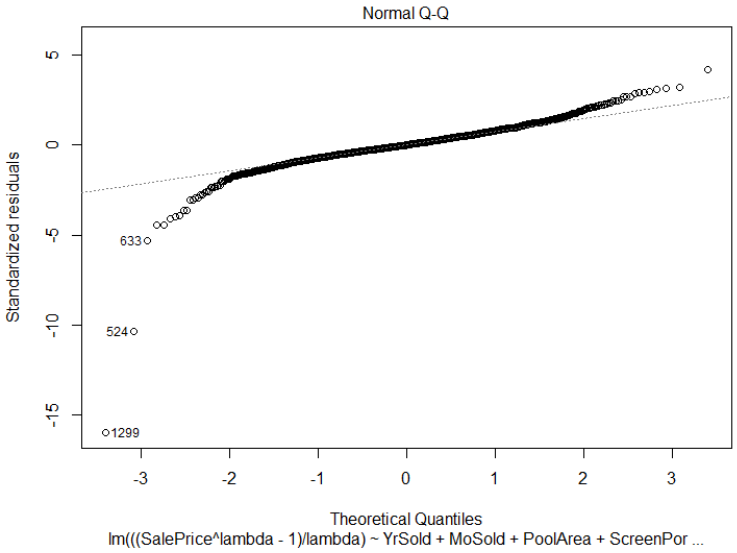
```
Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)     4.443e+01  1.972e+01   2.253 0.024412 *
YrSold         -2.274e-02  9.782e-03  -2.325 0.020236 *
MoSold         -8.174e-04  4.790e-03  -0.171 0.864528
PoolArea       -1.227e-03  3.264e-04  -3.758 0.000178 ***
ScreenPorch     1.126e-03  2.388e-04   4.714 2.66e-06 ***
X3SsnPorch      7.160e-04  4.351e-04   1.645 0.100096
EnclosedPorch   4.874e-04  2.344e-04   2.079 0.037783 *
OpenPorchSF    -1.491e-05  2.120e-04  -0.070 0.943945
WoodDeckSF      4.005e-04  1.106e-04   3.622 0.000303 ***
PavedDriveP     1.022e-01  1.050e-01   0.973 0.330691
PavedDriveY     1.431e-01  6.217e-02   2.302 0.021496 *
GarageArea      3.622e-05  1.354e-04   0.267 0.789145
GarageCars      2.229e-01  3.969e-02   5.617 2.34e-08 ***
Fireplaces      1.380e-01  2.469e-02   5.590 2.72e-08 ***
TotRmsAbvGrd    4.950e-02  1.749e-02   2.831 0.004712 **
KitchenAbvGr   -1.392e-01  7.283e-02  -1.912 0.056092 .
BedroomAbvGr    4.987e-03  2.417e-02   0.206 0.836597
HalfBath        6.976e-02  3.697e-02   1.887 0.059402 .
FullBath        1.328e-01  3.943e-02   3.367 0.000779 ***
BsmtHalfBath    3.971e-02  5.694e-02   0.697 0.485647
BsmtFullBath    1.991e-01  3.645e-02   5.462 5.56e-08 ***
GrLivArea       6.469e-04  8.038e-05   8.048 1.77e-15 ***
LowQualFinSF   -2.088e-04  2.833e-04  -0.737 0.461352
X2ndFlrSF      -8.708e-05  7.446e-05  -1.170 0.242388
TotalBsmtSF     2.016e-04  9.811e-05   2.055 0.040054 *
BsmtUnfSF      -5.537e-05  8.636e-05  -0.641 0.521554
BsmtFinSF1      3.304e-05  8.407e-05   0.393 0.694347
YearRemodAdd    2.416e-03  9.779e-04   2.471 0.013609 *
YearBuilt       7.491e-03  9.318e-04   8.039 1.89e-15 ***
OverallCond     1.429e-01  1.483e-02   9.635  < 2e-16 ***
OverallQual     2.529e-01  1.767e-02  14.312  < 2e-16 ***
LotArea         7.501e-06  1.739e-06   4.314 1.72e-05 ***
MSSubClass     -1.977e-03  3.651e-04  -5.416 7.15e-08 ***
StreetPave      5.491e-01  2.197e-01   2.499 0.012565 *
LandContourHLS  3.375e-01  9.526e-02   3.543 0.000409 ***
LandContourLow  2.392e-01  1.152e-01   2.076 0.038101 *
LandContourLvl  1.972e-01  6.765e-02   2.915 0.003616 **
LandSlopeMod    1.737e-01  7.446e-02   2.333 0.019806 *
LandSlopeSev   -6.345e-02  1.762e-01  -0.360 0.718884
ExterQualFa    -1.861e-01  1.837e-01  -1.013 0.311161
ExterQualGd    -3.861e-02  8.638e-02  -0.447 0.654936
ExterQualTA    -1.724e-01  9.638e-02  -1.789 0.073816 .
CentralAirY     1.743e-01  6.272e-02   2.779 0.005518 **
KitchenQualFa  -2.500e-01  1.148e-01  -2.177 0.029634 *
KitchenQualGd  -2.840e-01  6.472e-02  -4.388 1.23e-05 ***
KitchenQualTA  -3.603e-01  7.348e-02  -4.903 1.05e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.4808 on 1414 degrees of freedom
Multiple R-squared:  0.8773,    Adjusted R-squared:  0.8734
F-statistic: 224.7 on 45 and 1414 DF,  p-value: < 2.2e-16
```

Looking at the residuals of this model, we can see that after the boxcox transformation the residuals appear to be mostly homoscedastic, with the exception of a couple influential points (as

shown below.) Based on the residual plot, we can see that the points 524 and 1299 could be seen as influential points.



The Normal Q-Q plot was not as normal as we had hoped for as well, but since our goal was focused on the predictions, we can get away with the Q-Q plot that we have. With this model, we received a submission score of **0.14399**! This was good for a rank of 2505. Although we acknowledge that we could have improved this model, we were satisfied with it.



Code Used:

1st model:

```
library(mice)
library(missForest)


train <- read.csv("train.csv", header = TRUE)
train[c(1:5),]
str(train)

train2 <- train
train2 <- as.data.frame(unclass(train2), stringsAsFactors = TRUE)
str(train2)

#Generate 10% missing values at Random
train2.mis <- prodNA(train2, noNA = 0.1)
str(train2.mis)

#Check missing values introduced in the data
summary(train2.mis)

#remove categorical variables
train2.mis1 <- subset(train2.mis, select = -c(MSZoning, Street, Alley, LotShape,
LandContour, Utilities, LotConfig, LandSlope, Neighborhood, Condition1, Condition2,
BldgType, HouseStyle, RoofStyle, RoofMatl, Exterior1st, Exterior2nd, MasVnrType,
ExterQual, ExterCond, Foundation, BsmtQual, BsmtCond, BsmtExposure, BsmtFinType1,
BsmtFinType2, Heating, HeatingQC, CentralAir, Electrical, KitchenQual, Functional,
FireplaceQu, GarageType, GarageFinish, GarageQual, GarageCond, PavedDrive, PoolQC,
Fence, MiscFeature, SaleType, SaleCondition))
summary(train2.mis1)

#impute numercial values
imputed_Data1<- mice(train2.mis1, m=1, maxit = 50, method = 'pmm', seed = 500)
summary(imputed_Data1)
complete(imputed_Data1)
train2_imputed_numerical <- complete(imputed_Data1)
train2_imputed_numerical[c(1:5),]

#remove numerical values from train2.mis
train2.mis2 <- subset(train2.mis, select = c(MSZoning, Street, Alley, LotShape,
LandContour, Utilities, LotConfig, LandSlope, Neighborhood))
train2.mis3 <- subset(train2.mis, select = c(Condition1, Condition2, BldgType,
HouseStyle, RoofStyle, RoofMatl, Exterior1st))
train2.mis4 <- subset(train2.mis, select = c(Exterior2nd, MasVnrType, ExterQual,
ExterCond, Foundation, BsmtQual, BsmtCond, BsmtExposure))
train2.mis5 <- subset(train2.mis, select = c(BsmtFinType1, BsmtFinType2, Heating,
HeatingQC, CentralAir, Electrical, KitchenQual, Functional))
train2.mis6 <- subset(train2.mis, select = c(FireplaceQu, GarageType, GarageFinish,
GarageQual, GarageCond, PavedDrive, PoolQC, Fence, MiscFeature, SaleType,
SaleCondition))
```

```
summary(train2.mis2)
summary(train2.mis3)
summary(train2.mis4)
summary(train2.mis5)
summary(train2.mis6)


#impute categorical values pt1
imputed_Data2 <- mice(train2.mis2, m=1, maxit = 50, method = 'polyreg', seed = 500)
summary(imputed_Data2)
complete(imputed_Data2)
train2_imputed_categorical1 <- complete(imputed_Data2)
train2_imputed_categorical1[c(1:5),]

#impute categorical values pt2
imputed_Data3 <- mice(train2.mis3, m=1, maxit = 50, method = 'polyreg', seed = 500)
summary(imputed_Data3)
complete(imputed_Data3)
train2_imputed_categorical2 <- complete(imputed_Data3)
train2_imputed_categorical2[c(1:5),]

#impute categorical values pt3
imputed_Data4 <- mice(train2.mis4, m=1, maxit = 50, method = 'polyreg', seed = 500)
summary(imputed_Data4)
complete(imputed_Data4)
train2_imputed_categorical3 <- complete(imputed_Data4)
train2_imputed_categorical3[c(1:5),]

#impute categorical values pt4
imputed_Data5 <- mice(train2.mis5, m=1, maxit = 50, method = 'polyreg', seed = 500)
summary(imputed_Data5)
complete(imputed_Data5)
train2_imputed_categorical4 <- complete(imputed_Data5)
train2_imputed_categorical4[c(1:5),]

#impute categorical values pt5
imputed_Data6 <- mice(train2.mis6, m=1, maxit = 50, method = 'polyreg', seed = 500)
summary(imputed_Data6)
complete(imputed_Data6)
train2_imputed_categorical5<- complete(imputed_Data6)
train2_imputed_categorical5[(1:5),]


#combine all datasets
##train2_imputed_numerical1
##train2_imputed_categorical1
##train2_imputed_categorical2
##train2_imputed_categorical3
##train2_imputed_categorical4
##train2_imputed_categorical5

Comb1 <- cbind(train2_imputed_numerical, train2_imputed_categorical1)
Comb2 <- cbind(Comb1, train2_imputed_categorical2)
Comb3 <- cbind(Comb2, train2_imputed_categorical3)
Comb4 <- cbind(Comb3, train2_imputed_categorical4)
```

```
Complete_df <- cbind(Comb4, train2_imputed_categorical5)


#Turn dataset to CSV File
write.csv(Complete_df,"C:\\Users\\arena\\OneDrive\\Documents\\R\\MATH 444\\Project
1\\MICE Imputed Dataset.csv",row.names = TRUE)

library(car)
library(MASS)
library(SignifReg)

train <- read.csv("MICE Imputed Dataset.csv", header = TRUE)
train[c(1:5),]

attach(train)

str(train)
intercept_model <-lm(SalePrice ~ 1)
full_model <- lm(SalePrice ~ MSZoning + LotFrontage + LotArea + OverallQual +
OverallCond + Neighborhood + HouseStyle + ExterQual + ExterCond + X1stFlrSF +
X2ndFlrSF + FullBath + HalfBath + BedroomAbvGr + KitchenAbvGr + Functional +
GarageType + GarageCars + GarageArea + GarageQual + GarageCond + OpenPorchSF +
EnclosedPorch + PoolArea + Fence + MiscFeature + MiscVal + SaleCondition)
summary(full_model)

###stepAIC Model Building
forwardstep_model <- stepAIC(full_model, scope = ~ MSZoning + LotFrontage + LotArea +
OverallQual + OverallCond + Neighborhood + HouseStyle + ExterQual + ExterCond +
X1stFlrSF + X2ndFlrSF + FullBath + HalfBath + BedroomAbvGr + KitchenAbvGr + Functional
+ GarageType + GarageCars + GarageArea + GarageQual + GarageCond + OpenPorchSF +
EnclosedPorch + PoolArea + Fence + MiscFeature + MiscVal + SaleCondition, direction =
"backward")
backwardstep_model <- stepAIC(intercept_model,  scope = ~ MSZoning + LotFrontage +
LotArea + OverallQual + OverallCond + Neighborhood + HouseStyle + ExterQual +
ExterCond + X1stFlrSF + X2ndFlrSF + FullBath + HalfBath + BedroomAbvGr + KitchenAbvGr
+ Functional + GarageType + GarageCars + GarageArea + GarageQual + GarageCond +
OpenPorchSF + EnclosedPorch + PoolArea + Fence + MiscFeature + MiscVal +
SaleCondition, direction = "forward")
bothstep_model <- stepAIC(full_model,  scope = ~ MSZoning + LotFrontage + LotArea +
OverallQual + OverallCond + Neighborhood + HouseStyle + ExterQual + ExterCond +
X1stFlrSF + X2ndFlrSF + FullBath + HalfBath + BedroomAbvGr + KitchenAbvGr + Functional
+ GarageType + GarageCars + GarageArea + GarageQual + GarageCond + OpenPorchSF +
EnclosedPorch + PoolArea + Fence + MiscFeature + MiscVal + SaleCondition, direction =
"both")
summary(forwardstep_model)
summary(backwardstep_model)
summary(bothstep_model)


#Backward:
AIC=30539.57
SalePrice ~ MSZoning + LotFrontage + LotArea + OverallQual +
    OverallCond + Neighborhood + HouseStyle + ExterQual + ExterCond +
    X1stFlrSF + X2ndFlrSF + FullBath + HalfBath + BedroomAbvGr +
    KitchenAbvGr + Functional + GarageType + GarageCars + EnclosedPorch +
```

```
    PoolArea + Fence + SaleCondition

#Forward:
AIC=30539.57
SalePrice ~ OverallQual + Neighborhood + X1stFlrSF + X2ndFlrSF +
    ExterQual + GarageCars + KitchenAbvGr + LotArea + SaleCondition +
    OverallCond + PoolArea + GarageType + EnclosedPorch + LotFrontage +
    ExterCond + Fence + HouseStyle + HalfBath + Functional +
    MSZoning + BedroomAbvGr + FullBath

#Both:
AIC=30539.57
SalePrice ~ MSZoning + LotFrontage + LotArea + OverallQual +
    OverallCond + Neighborhood + HouseStyle + ExterQual + ExterCond +
    X1stFlrSF + X2ndFlrSF + FullBath + HalfBath + BedroomAbvGr +
    KitchenAbvGr + Functional + GarageType + GarageCars + EnclosedPorch +
    PoolArea + Fence + SaleCondition

###SignifReg Model Building
ARSQ_Backward_Model <- SignifReg(full_model, scope = ~ MSZoning + LotFrontage +
LotArea + OverallQual + OverallCond + Neighborhood + HouseStyle + ExterQual +
ExterCond + X1stFlrSF + X2ndFlrSF + FullBath + HalfBath + BedroomAbvGr + KitchenAbvGr
+ Functional + GarageType + GarageCars + GarageArea + GarageQual + GarageCond +
OpenPorchSF + EnclosedPorch + PoolArea + Fence + MiscFeature + MiscVal +
SaleCondition, direction = "backward", criterion = "r-adj", trace = TRUE)
ARSQ_Forward_Model <- SignifReg(intercept_model, scope = ~ MSZoning + LotFrontage +
LotArea + OverallQual + OverallCond + Neighborhood + HouseStyle + ExterQual +
ExterCond + X1stFlrSF + X2ndFlrSF + FullBath + HalfBath + BedroomAbvGr + KitchenAbvGr
+ Functional + GarageType + GarageCars + GarageArea + GarageQual + GarageCond +
OpenPorchSF + EnclosedPorch + PoolArea + Fence + MiscFeature + MiscVal +
SaleCondition, direction = "forward", criterion = "r-adj", trace = TRUE)
ARSQ_Both_Model <- SignifReg(full_model, scope = ~ MSZoning + LotFrontage + LotArea +
OverallQual + OverallCond + Neighborhood + HouseStyle + ExterQual + ExterCond +
X1stFlrSF + X2ndFlrSF + FullBath + HalfBath + BedroomAbvGr + KitchenAbvGr + Functional
+ GarageType + GarageCars + GarageArea + GarageQual + GarageCond + OpenPorchSF +
EnclosedPorch + PoolArea + Fence + MiscFeature + MiscVal + SaleCondition, direction =
"both", criterion = "r-adj", trace = TRUE)

summary(ARSQ_Backward_Model)
summary(ARSQ_Forward_Model)
summary(ARSQ_Both_Model)


ARSQ_Backward_Model
ARSQ_Forward_Model
ARSQ_Both_Model

#Based on the results of adj-R squared, the ARSQ_Both_Model is the best model!
library(car)
vif(ARSQ_Both_Model)

#the vifs show that Neighborhood, and MSZoning are show high
#signs of multicollinearty, so we want to remove these predictors
```

```
ARSQ_Both_Model_edited1 <- lm(SalePrice ~ LotFrontage + LotArea + OverallQual +
OverallCond + HouseStyle + ExterQual + ExterCond + X1stFlrSF + X2ndFlrSF + FullBath +
HalfBath + BedroomAbvGr + KitchenAbvGr + Functional + GarageType + GarageCars +
EnclosedPorch + PoolArea + Fence + MiscFeature + SaleCondition)
vif(ARSQ_Both_Model_edited1)

plot(ARSQ_Both_Model_edited1)

#find optimal lambda for Box-Cox transformation
bc <- boxcox(SalePrice ~ LotFrontage + LotArea + OverallQual + OverallCond +
HouseStyle + ExterQual + ExterCond + X1stFlrSF + X2ndFlrSF + FullBath + HalfBath +
BedroomAbvGr + KitchenAbvGr + Functional + GarageType + GarageCars + EnclosedPorch +
PoolArea + Fence + MiscFeature + SaleCondition)
lambda <- bc$x[which.max(bc$y)]
ARSQ_Both_Model_edited2 <- lm((SalePrice^(lambda)) ~ LotFrontage + LotArea +
OverallQual + OverallCond + HouseStyle + ExterQual + ExterCond + X1stFlrSF + X2ndFlrSF
+ FullBath + HalfBath + BedroomAbvGr + KitchenAbvGr + Functional + GarageType +
GarageCars + EnclosedPorch + PoolArea + Fence + MiscFeature + SaleCondition)
ARSQ_Both_Model_edited3 <- lm(((SalePrice^lambda - 1) /1)  ~ LotFrontage + LotArea +
OverallQual + OverallCond + HouseStyle + ExterQual + ExterCond + X1stFlrSF + X2ndFlrSF
+ FullBath + HalfBath + BedroomAbvGr + KitchenAbvGr + Functional + GarageType +
GarageCars + EnclosedPorch + PoolArea + Fence + MiscFeature + SaleCondition)

summary(ARSQ_Both_Model_edited2)
summary(ARSQ_Both_Model_edited3)

plot(ARSQ_Both_Model_edited2)
plot(ARSQ_Both_Model_edited3)


final_model1 <- ARSQ_Both_Model_edited2
summary(final_model1)
vif(final_model1)
plot(final_model1)
data_test = read.csv("test.csv", header = TRUE)

y_hat = as.numeric(predict(final_model1, data_test))

y_hat_converted = (y_hat)^(1/lambda)

output = data.frame(Id = data_test$Id, SalePrice = y_hat_converted)
output[c(1:5),]
write.csv(output, file = "Submission2.csv", row.names = FALSE)
```

2nd model:

```
library(SignifReg)
setwd("C:/Users/josie/OneDrive/Documents/444")
View(train3_numeric_impute_mean)
```

```
newtrainmodel = read.csv("train imputed mean.csv", header = TRUE)
View(newtrainmodel)
officialtrain = newtrainmodel[,2:50]
nullmodel = lm(SalePrice~1, data = officialtrain)
fullmodel = lm(SalePrice ~ MSSubClass + LotFrontage + LotArea + OverallQual +
OverallCond + YearBuilt
               + YearRemodAdd + MasVnrArea + BsmtFinSF1 + BsmtFinSF2 + BsmtUnfSF +
TotalBsmtSF
               + X1stFlrSF + X2ndFlrSF + LowQualFinSF + GrLivArea + BsmtFullBath +
BsmtHalfBath
               + FullBath + HalfBath + BedroomAbvGr + KitchenAbvGr + TotRmsAbvGrd +
Fireplaces + GarageYrBlt
               + GarageCars + GarageArea + WoodDeckSF + OpenPorchSF + EnclosedPorch +
X3SsnPorch + ScreenPorch
               + PoolArea + MiscVal + MoSold + YrSold + MSZoning + Street + Alley +
LotShape + LandContour
               + Utilities + LotConfig + LandSlope + Neighborhood + Condition1 +
Condition2, data = officialtrain)
scope = list(lower=formula(nullmodel), upper = formula(fullmodel))
forwardSelect = SignifReg(nullmodel, scope = scope, direction = "forward", trace =
FALSE, criterion = "AIC", alpha = 1)
forwardSelect
AIC(forwardSelect)
summary(forwardSelect)
reducedmodel = lm(SalePrice ~ OverallQual + GrLivArea + BsmtFinSF1  + GarageCars +
MSSubClass
                  + YearBuilt + BedroomAbvGr + OverallCond + LotArea + MasVnrArea
                  + BsmtFullBath, data = officialtrain)
bc <- boxcox(SalePrice ~ OverallQual + GrLivArea + BsmtFinSF1  + GarageCars +
MSSubClass
             + YearBuilt + BedroomAbvGr + OverallCond + LotArea + MasVnrArea
             + BsmtFullBath, data = officialtrain)
(lambda <- bc$x[which.max(bc$y)])
boxcoxreducedmodel <- lm(SalePrice^0.1010101 ~ OverallQual + GrLivArea + BsmtFinSF1  +
GarageCars + MSSubClass
                         +              + YearBuilt + BedroomAbvGr + OverallCond +
LotArea + MasVnrArea
                         +              + BsmtFullBath, data = officialtrain)
AIC(boxcoxreducedmodel)
bc <- boxcox(SalePrice ~ OverallQual + GrLivArea + BsmtFinSF1 + GarageCars +
               +                  MSSubClass + YearBuilt + BedroomAbvGr + OverallCond
+ LotArea +
               +                  MasVnrArea + BsmtFullBath + TotRmsAbvGrd +
WoodDeckSF + ScreenPorch +
               +                  TotalBsmtSF + YearRemodAdd + KitchenAbvGr +
Condition2 +
               +                  Fireplaces + FullBath + LandSlope + Neighborhood +
Utilities +
               +                  LotShape + LandContour, data = officialtrain)
(lambda <- bc$x[which.max(bc$y)])
boxcoxforwardselect <- lm(SalePrice^lambda ~ OverallQual + GrLivArea + BsmtFinSF1 +
GarageCars +
                            +                                 +
MSSubClass + YearBuilt + BedroomAbvGr + OverallCond + LotArea +
```

```
                              +                                  +
MasVnrArea + BsmtFullBath + TotRmsAbvGrd + WoodDeckSF + ScreenPorch +
                              +                                  +
TotalBsmtSF + YearRemodAdd + KitchenAbvGr + Condition2 +
                              +                                  +
Fireplaces + FullBath + LandSlope + Neighborhood + Utilities +
                              +                                  +
LotShape + LandContour, data = officialtrain)
newy_hat = as.numeric(predict(boxcoxforwardselect, officialtrain))
newy_hat = newy_hat^(1/lambda)
output = data.frame(Id = data_test$Id, SalePrice = y_hat_converted)
write.csv(output, file = "kagglesubmission3.csv", row.names = FALSE)
```

3rd model:

```
library(SignifReg)
library(MASS)
library(car)
house = read.csv("train.csv", header=TRUE)
house2 = read.csv("test.csv", header=TRUE)
attach(house)
attach(house2)

## Scope for 1st step ##

scope1 = ~ YrSold + MoSold + PoolArea + ScreenPorch + X3SsnPorch + EnclosedPorch +
OpenPorchSF +
WoodDeckSF + PavedDrive + GarageArea + GarageCars + Fireplaces + TotRmsAbvGrd +
KitchenAbvGr + BedroomAbvGr +
HalfBath + FullBath + BsmtHalfBath + BsmtFullBath + GrLivArea + LowQualFinSF +
X2ndFlrSF + X1stFlrSF + +TotalBsmtSF +
BsmtUnfSF + BsmtFinSF2 + BsmtFinSF1 + YearRemodAdd + YearBuilt + OverallCond +
OverallQual + LotArea + MSSubClass


## Scope for 2nd step ##

scope2 = ~ YrSold + MoSold + PoolArea + ScreenPorch + X3SsnPorch + EnclosedPorch +
OpenPorchSF + WoodDeckSF +
PavedDrive + GarageArea + GarageCars + Fireplaces + TotRmsAbvGrd + KitchenAbvGr +
BedroomAbvGr + HalfBath + FullBath +
BsmtHalfBath + BsmtFullBath + GrLivArea + LowQualFinSF + X2ndFlrSF + X1stFlrSF +
+TotalBsmtSF + BsmtUnfSF + BsmtFinSF2 +
 BsmtFinSF1 + YearRemodAdd + YearBuilt + OverallCond + OverallQual + LotArea +
MSSubClass +
KitchenQual + CentralAir + ExterQual + LandSlope + LandContour + Street


## Model Setup ##

null_model = lm(SalePrice ~ 1)
full_model = lm(SalePrice ~ YrSold + MoSold + PoolArea + ScreenPorch + X3SsnPorch +
EnclosedPorch + OpenPorchSF +
WoodDeckSF + PavedDrive + GarageArea + GarageCars + Fireplaces + TotRmsAbvGrd +
KitchenAbvGr + BedroomAbvGr +
```

```
HalfBath + FullBath + BsmtHalfBath + BsmtFullBath + GrLivArea + LowQualFinSF +
X2ndFlrSF + X1stFlrSF + +TotalBsmtSF +
BsmtUnfSF + BsmtFinSF2 + BsmtFinSF1 + YearRemodAdd + YearBuilt + OverallCond +
OverallQual + LotArea + MSSubClass)


## Construct 1st Iteration Model ##

m1 = SignifReg(null_model, direction= "forward", criterion= "r-adj", scope= scope1)


## Construct 2nd Iteration Model ##

m2 = SignifReg(m1, direction= "both", criterion= "r-adj", scope= scope2)


## Boxcox Transformation ##

bc = boxcox(m2)
lambda = bc$x[which.max(bc$y)]

bc_m2 = lm(((SalePrice^lambda - 1) / lambda)~ YrSold + MoSold + PoolArea + ScreenPorch
+ X3SsnPorch + EnclosedPorch +
               OpenPorchSF + WoodDeckSF + PavedDrive + GarageArea + GarageCars +
               Fireplaces + TotRmsAbvGrd + KitchenAbvGr + BedroomAbvGr +
               HalfBath + FullBath + BsmtHalfBath + BsmtFullBath + GrLivArea +
               LowQualFinSF + X2ndFlrSF  + TotalBsmtSF + BsmtUnfSF +
               BsmtFinSF1  + YearRemodAdd + YearBuilt + OverallCond +
               OverallQual + LotArea + MSSubClass + Street + LandContour +
               LandSlope + ExterQual + CentralAir + KitchenQual + PavedDrive)


## Construct Predictions CSV ##

y_hat = as.numeric(predict(bc_m2, house2))
output = data.frame(Id = data_test$Id, SalePrice = ((y_hat*lambda)+1)^(1/lambda))
write.csv(output, file = "my_submission.csv", row.names = FALSE)
```