Smith
SCHOOL OF BUSINESS
Queen's University

Master of
Management Analytics
Toronto

MMA 867
Predictive Modelling

Professor Jue Wang

Assignment 1: House Prices Kaggle Competition
April 30th, 2022

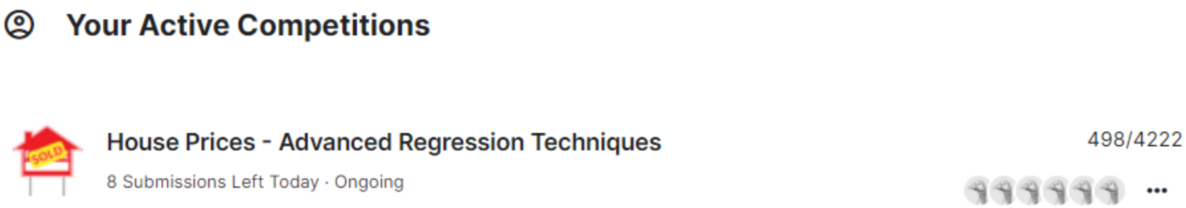
Team Carlton

Student Name	Student Number
Nishchay Vermani	20289622
Guowei (Louis) Zhao	20317832
Yanlu (Sherry) Sun	20311278
Elizabeth Kim	20312997
Jane Zhou	20321175
Ramandeep Singh	20313357
Isaac Elfaks	20313867

Order of files:

Filename	Pages	Comments and/or Instructions
MMA867_Assignment1_Carlton	9	pdf
House Price Model		txt
Predicted House Prices Ridge		xlsx

Position on Leaderboard: Top 11.79%



Your Active Competitions

House Prices - Advanced Regression Techniques

8 Submissions Left Today · Ongoing

498/4222

Kaggle House Prices – Advanced Regression Techniques

The objective of the competition is to predict the final price of each home using 79 explanatory variables describing all aspects of homes

1. **[20 pts]:** Read the instructions on Kaggle. Learn how to join a Kaggle competition and submit your results. You will find that some predictors contain "missing data," NA. Figure out how to handle missing data in regression.

According to the data description provided from the competition, NA does not mean the values are missing for the variables listed below. NA means that the particular feature of the house is not present for these variables:

- Alley
- Basement Height (BsmtQual)
- Basement Condition (BsmtCond)
- Basement Exposure (BsmtExposure)
- Basement Finished Area Type 1 (BsmtFinType1)
- Basement Finished Area Type 2 (BsmtFinType2)
- Fireplace Quality (FireplaceQu)
- Garage Type
- Garage Finish
- Garage Quality (GarageQual)
- Garage Condition (GarageCond)
- Pool Quality (PoolQC)
- Fence Quality (Fence)
- Miscellaneous Features (MiscFeature)

Of these variables, the percentage of non-present features (NA) were calculated. Variables that had more than 50% of data showing as NA: Alley, Pool Quality, Fence, Miscellaneous Features, Basement Condition, Basement Finished Area Type 2, and Garage Condition were removed from the training and testing dataset assuming that they will have no predictive power. Since the majority of the houses have those features missing, we are assuming that it will not have a significant effect on house prices. The same logic was used to further exclude categorial variables: Street, Utilities, Condition2, RoofMat1, Heating, and Land Contour for both training and testing set.

Of the variables that contain NA, Basement Exposure, Basement Finished Type 1, Fireplace Quality, Garage Type, Garage Finish, Garage Quality remained in the new training and testing dataset. NAs for these variables were remapped as "None" to be incorporated in our analysis because R takes NA as missing values (See Appendix A).

For true missing values if the data was integer or numeric, median was used to replace the missing value. For categorical variables including character data, mode, the most frequent observation was used to replace the missing value. This was done in both the training and testing dataset (See Appendix B).

2. **[60 pts]:** Build a regression model for house price prediction and write a report explaining how you approached the task, the steps you took, and how you revised your model (**must explore both LASSO and Ridge regression**) as your analyses progressed, etc. Comment on the quality of your predictions. Include your model as an Appendix in your report. Submit the PDF of your report, and your model file(s) (code, spreadsheet, etc.)

Step 1: Loaded both training and test data sets to R and completed exploratory data analysis.

Step 2: Data cleansing

- Training data set sales price column has been verified that there is no missing data, thus no further action is required.
- Handling missing data in predictors.
    - Replaced “NA” with “None” when “NA” means the particular feature of the house is not present.
    - Replaced missing integer or numeric values with variable median, replaced missing character values with variable mode.
- Handling outliers.
    - We also removed some lines of data because they are considered outliers and are significantly from other similar inputs (See Appendix K).
    - Initially, histograms were created for each continuous variable to check the value distribution, univariate outliers were removed. However, this action decreased the model prediction quality. This might be due to overfitting. Thus, no univariate outliers were removed.
    - Cook’s Distance plots were used to identify any multivariate outliers, removing identified rows of records increased model prediction quality.
    - Slight outliers were kept in the model since we do not want overfitting.
- Removing predictors that have less predicting power.
    - Bar charts were created for each categorical variable to check value distribution. If bar chart shows one option has remarkably high ratio, the variable was removed from the model as it has less predicting power.

Step 3: Feature engineering (See Appendix C)

- Regarding to the variables that are not appearing normally distributed, we took log of these variables to correct the skewness of them. We also took log of the dependent variable “Sale Price” into our model to make our prediction more accurate.
- An assumption was made that if highly correlated variables are added into one predictor, it could become a stronger predictor, even if they may be insignificant individually. This feature engineering method was used to create variables total square feet, total porch, house age, last modified, and remodelled variables.
- For the new bathroom variable, the four variables for Bathroom (BsmtFullBath, BsmtHalfBath, FullBath, HalfBath) were added together but half bath was only counted as 0.5 because it has only two of the four main bathroom components compared to the full bath.

- The last two feature engineering were added after the first submission and improved the model prediction quality.

Step 4: Splitting the training data set into training and validation data sets.

- To explore both LASSO and Ridge Regression, training dataset was first randomly split into training (80%) and validation sets (20%).

Step 5: Creating the y variable and matrix of x variables

- First the Y is log (Sale Price) was put into the training set and the original variables, as well as the different interactions based on the correlation heat map (See Appendix L) and some continuous variables were put into log () depending on the distribution. We then divided the matrix X into training and validation to run LASSO and Ridge Regression (Appendix D).

Step 6: Creating regression models

- Lasso regression plot for the training set can be viewed in Appendix E. Ridge regression plot for the training set can be viewed in Appendix G. When we used cross validation to identify the lambda that minimize the training error (mean squared error (MSE)): the result was -9.637 for Lasso and -3.3901 for Ridge (See Appendix I). After identifying the lambda with the smallest mean squared error, we used these lambdas to find the most optimal model. From looking at the optimal lambda point from the plots and viewing the coefficients, most variables stayed in the model. The low lambda point meant that we were able to better fit the training data by retaining most of the variables.
- When we used the optimal model to calculate mean squared error of our validation data set, MSE for Lasso was 532354282.03 and MSE for Ridge was 754707329.02 (See Appendix J), and the mean absolute percentage error for ridge was 7.67 and Lasso 7.87. In terms of house prices, Naturally, RMSE (Root Mean Squared Error) results are similar as well. RMSE for Lasso was 23073 and RMSE for Ridge regression was 27472.
- The entire training set was used to train the model before creating the final prediction results.
- Our team decided to submit both house prices from Lasso and Ridge Regression. We found that the Ridge Regression results always scored higher on the leaderboard compared to our results from Lasso regression although the MSE for Lasso was lower.

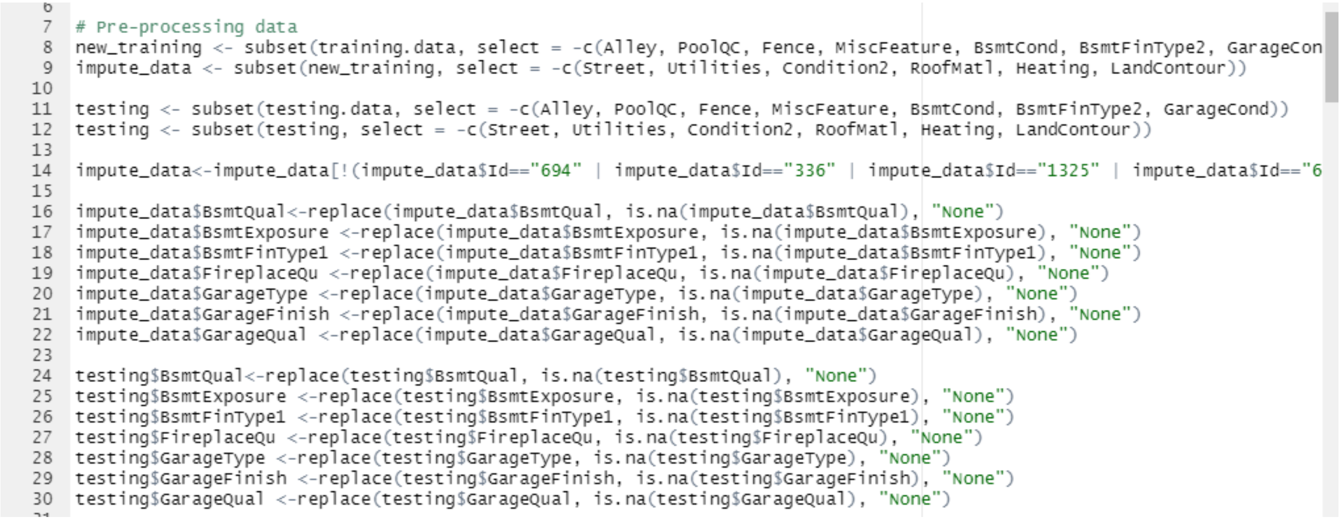
Taking our leaderboard score into consideration, our group concluded that the predicted prices from Ridge would be the better prediction of houses prices. The MSE results from our model was calculated using the training data provided from the Kaggle competition. However, the performance on the leaderboard is calculated from sale price for testing dataset that is not provided to the participants of the competition. Therefore, there could be inconsistencies between the data.

Our models have good prediction power. Our LASSO regression model has an RMSE value of $23,073, and Ridge regression model has an RMSE value of $27,472. Since the house price from the training data set provided has a mean of $180,921 and median of $163,000, comparing to these values, the RMSE values of our models are fairly low, indicating that our models have sufficient prediction power to help predict house prices.

3. **[20 pts]:** On the front page of your report, include your position on the leaderboard at the time of your last submission. Please also include the screenshot showing your team's position on the leaderboard in the Appendix.

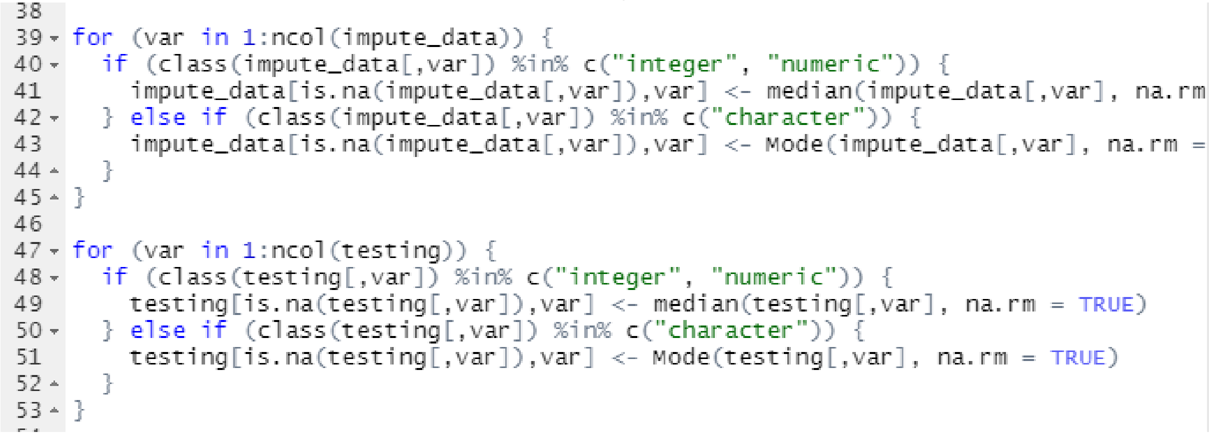
Our team's last submission was made on April 24th. At the time of the submission, our position on the leader board was 498th out of 4222, putting us at the top 11.79% (See Appendix M). The sale prices from Ridge regression were used to achieve this high score. The Kaggle submission file is included as a supporting document to this report.

Appendix A
Converting NA to “None”



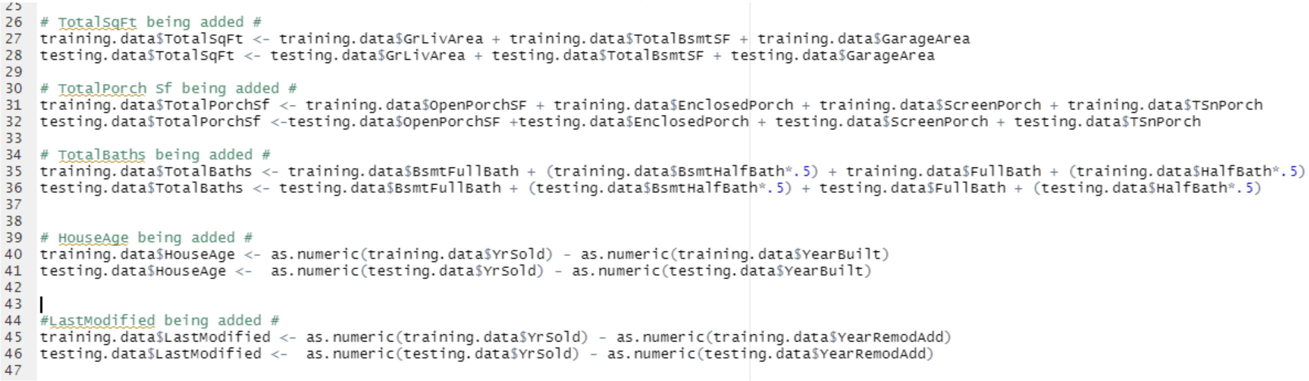
```
# Pre-processing data
new_training <- subset(training.data, select = -c(Alley, PoolQC, Fence, MiscFeature, BsmtCond, BsmtFinType2, GarageCon
impute_data <- subset(new_training, select = -c(Street, Utilities, Condition2, RoofMatl, Heating, LandContour))

testing <- subset(testing.data, select = -c(Alley, PoolQC, Fence, MiscFeature, BsmtCond, BsmtFinType2, GarageCond))
testing <- subset(testing, select = -c(Street, Utilities, Condition2, RoofMatl, Heating, LandContour))

impute_data<-impute_data[!(impute_data$Id=="694" | impute_data$Id=="336" | impute_data$Id=="1325" | impute_data$Id=="6

impute_data$BsmtQual<-replace(impute_data$BsmtQual, is.na(impute_data$BsmtQual), "None")
impute_data$BsmtExposure <-replace(impute_data$BsmtExposure, is.na(impute_data$BsmtExposure), "None")
impute_data$BsmtFinType1 <-replace(impute_data$BsmtFinType1, is.na(impute_data$BsmtFinType1), "None")
impute_data$FireplaceQu <-replace(impute_data$FireplaceQu, is.na(impute_data$FireplaceQu), "None")
impute_data$GarageType <-replace(impute_data$GarageType, is.na(impute_data$GarageType), "None")
impute_data$GarageFinish <-replace(impute_data$GarageFinish, is.na(impute_data$GarageFinish), "None")
impute_data$GarageQual <-replace(impute_data$GarageQual, is.na(impute_data$GarageQual), "None")

testing$BsmtQual<-replace(testing$BsmtQual, is.na(testing$BsmtQual), "None")
testing$BsmtExposure <-replace(testing$BsmtExposure, is.na(testing$BsmtExposure), "None")
testing$BsmtFinType1 <-replace(testing$BsmtFinType1, is.na(testing$BsmtFinType1), "None")
testing$FireplaceQu <-replace(testing$FireplaceQu, is.na(testing$FireplaceQu), "None")
testing$GarageType <-replace(testing$GarageType, is.na(testing$GarageType), "None")
testing$GarageFinish <-replace(testing$GarageFinish, is.na(testing$GarageFinish), "None")
testing$GarageQual <-replace(testing$GarageQual, is.na(testing$GarageQual), "None")
```

Appendix B
Data Imputation



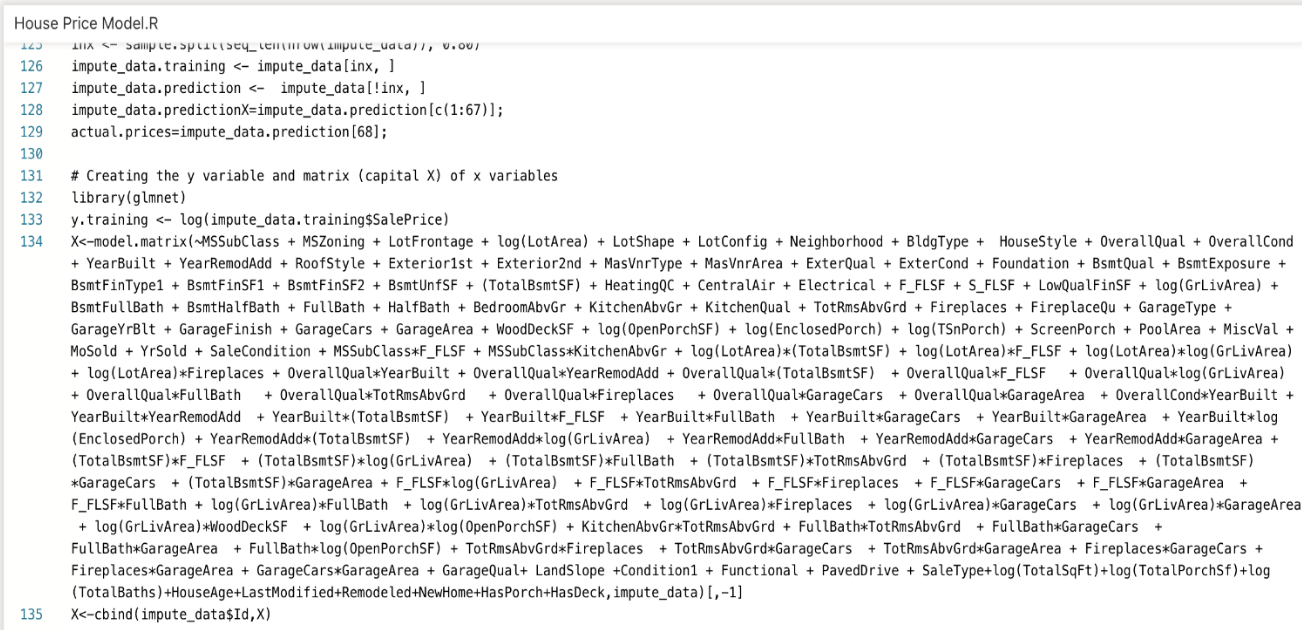
```
for (var in 1:ncol(impute_data)) {
  if (class(impute_data[,var]) %in% c("integer", "numeric")) {
    impute_data[is.na(impute_data[,var]),var] <- median(impute_data[,var], na.rm = TRUE)
  } else if (class(impute_data[,var]) %in% c("character")) {
    impute_data[is.na(impute_data[,var]),var] <- Mode(impute_data[,var], na.rm = TRUE)
  }
}

for (var in 1:ncol(testing)) {
  if (class(testing[,var]) %in% c("integer", "numeric")) {
    testing[is.na(testing[,var]),var] <- median(testing[,var], na.rm = TRUE)
  } else if (class(testing[,var]) %in% c("character")) {
    testing[is.na(testing[,var]),var] <- Mode(testing[,var], na.rm = TRUE)
  }
}
```

Appendix C
Feature Engineering



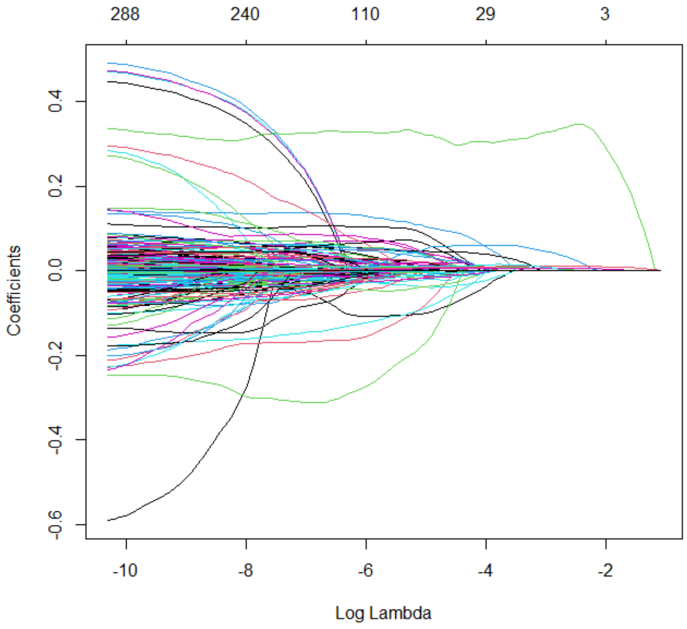
```
# TotalSqFt being added #
training.data$TotalSqFt <- training.data$GrLivArea + training.data$TotalBsmtSF + training.data$GarageArea
testing.data$TotalSqFt <- testing.data$GrLivArea + testing.data$TotalBsmtSF + testing.data$GarageArea

# TotalPorch Sf being added #
training.data$TotalPorchSf <- training.data$OpenPorchSF + training.data$EnclosedPorch + training.data$ScreenPorch + training.data$TSnPorch
testing.data$TotalPorchSf <-testing.data$OpenPorchSF +testing.data$EnclosedPorch + testing.data$ScreenPorch + testing.data$TSnPorch

# TotalBaths being added #
training.data$TotalBaths <- training.data$BsmtFullBath + (training.data$BsmtHalfBath*.5) + training.data$FullBath + (training.data$HalfBath*.5)
testing.data$TotalBaths <- testing.data$BsmtFullBath + (testing.data$BsmtHalfBath*.5) + testing.data$FullBath + (testing.data$HalfBath*.5)


# HouseAge being added #
training.data$HouseAge <- as.numeric(training.data$YrSold) - as.numeric(training.data$YearBuilt)
testing.data$HouseAge <-  as.numeric(testing.data$YrSold) - as.numeric(testing.data$YearBuilt)


#LastModified being added #
training.data$LastModified <- as.numeric(training.data$YrSold) - as.numeric(training.data$YearRemodAdd)
testing.data$LastModified <-  as.numeric(testing.data$YrSold) - as.numeric(testing.data$YearRemodAdd)
```

Appendix D
Feature Selection in Model

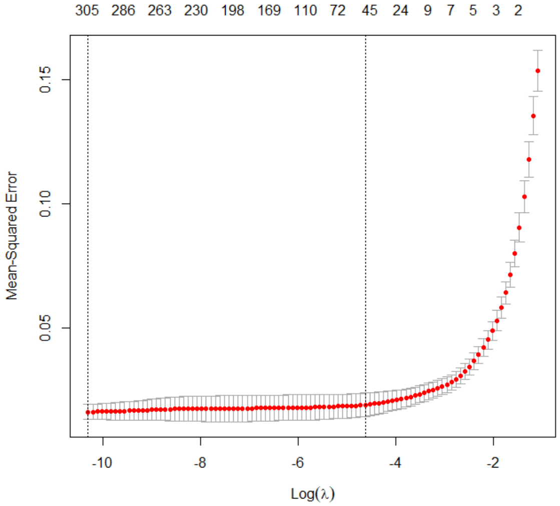


```
inx <- sample.split(seq_len(nrow(impute_data)), 0.80)
impute_data.training <- impute_data[inx, ]
impute_data.prediction <-  impute_data[!inx, ]
impute_data.predictionX=impute_data.prediction[c(1:67)];
actual.prices=impute_data.prediction[68];

# Creating the y variable and matrix (capital X) of x variables
library(glmnet)
y.training <- log(impute_data.training$SalePrice)
X<-model.matrix(~MSSubClass + MSZoning + LotFrontage + log(LotArea) + LotShape + LotConfig + Neighborhood + BldgType +  HouseStyle + OverallQual + OverallCond
+ YearBuilt + YearRemodAdd + RoofStyle + Exterior1st + Exterior2nd + MasVnrType + MasVnrArea + ExterQual + ExterCond + Foundation + BsmtQual + BsmtExposure +
BsmtFinType1 + BsmtFinSF1 + BsmtFinSF2 + BsmtUnfSF + (TotalBsmtSF) + HeatingQC + CentralAir + Electrical + F_FLSF + S_FLSF + LowQualFinSF + log(GrLivArea) +
BsmtFullBath + BsmtHalfBath + FullBath + HalfBath + BedroomAbvGr + KitchenAbvGr + KitchenQual + TotRmsAbvGrd + Fireplaces + FireplaceQu + GarageType +
GarageYrBlt + GarageFinish + GarageCars + GarageArea + WoodDeckSF + log(OpenPorchSF) + log(EnclosedPorch) + log(TSnPorch) + ScreenPorch + PoolArea + MiscVal +
MoSold + YrSold + SaleCondition + MSSubClass*F_FLSF + MSSubClass*KitchenAbvGr + log(LotArea)*(TotalBsmtSF) + log(LotArea)*F_FLSF + log(LotArea)*log(GrLivArea)
+ log(LotArea)*Fireplaces + OverallQual*YearBuilt + OverallQual*YearRemodAdd + OverallQual*(TotalBsmtSF)  + OverallQual*F_FLSF   + OverallQual*log(GrLivArea)
+ OverallQual*FullBath   + OverallQual*TotRmsAbvGrd   + OverallQual*Fireplaces   + OverallQual*GarageCars  + OverallQual*GarageArea  + OverallCond*YearBuilt +
YearBuilt*YearRemodAdd  + YearBuilt*(TotalBsmtSF)  + YearBuilt*F_FLSF  + YearBuilt*FullBath  + YearBuilt*GarageCars  + YearBuilt*GarageArea  + YearBuilt*log
(EnclosedPorch) + YearRemodAdd*(TotalBsmtSF)  + YearRemodAdd*log(GrLivArea)  + YearRemodAdd*FullBath  + YearRemodAdd*GarageCars  + YearRemodAdd*GarageArea +
(TotalBsmtSF)*F_FLSF  + (TotalBsmtSF)*log(GrLivArea)  + (TotalBsmtSF)*FullBath  + (TotalBsmtSF)*TotRmsAbvGrd  + (TotalBsmtSF)*Fireplaces  + (TotalBsmtSF)
*GarageCars  + (TotalBsmtSF)*GarageArea + F_FLSF*log(GrLivArea)  + F_FLSF*TotRmsAbvGrd  + F_FLSF*Fireplaces  + F_FLSF*GarageCars  + F_FLSF*GarageArea  +
F_FLSF*FullBath + log(GrLivArea)*FullBath  + log(GrLivArea)*TotRmsAbvGrd  + log(GrLivArea)*Fireplaces  + log(GrLivArea)*GarageCars  + log(GrLivArea)*GarageArea
 + log(GrLivArea)*WoodDeckSF  + log(GrLivArea)*log(OpenPorchSF) + KitchenAbvGr*TotRmsAbvGrd + FullBath*TotRmsAbvGrd  + FullBath*GarageCars  +
FullBath*GarageArea  + FullBath*log(OpenPorchSF) + TotRmsAbvGrd*Fireplaces  + TotRmsAbvGrd*GarageCars  + TotRmsAbvGrd*GarageArea + Fireplaces*GarageCars +
Fireplaces*GarageArea + GarageCars*GarageArea + GarageQual+ LandSlope +Condition1 + Functional + PavedDrive + SaleType+log(TotalSqFt)+log(TotalPorchSf)+log
(TotalBaths)+HouseAge+LastModified+Remodeled+NewHome+HasPorch+HasDeck,impute_data)[,-1]
X<-cbind(impute_data$Id,X)
```

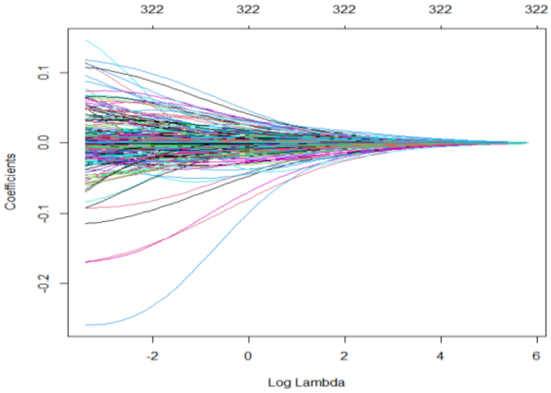
Appendix E
LASSO Plot



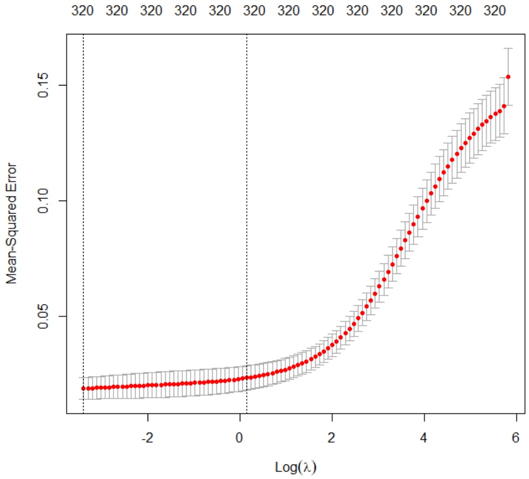
Appendix F
Cross Validation – LASSO



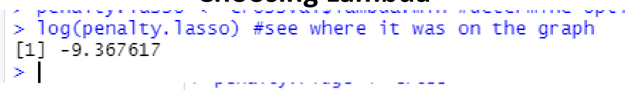
Appendix G
Ridge Plot



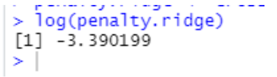
Appendix H
Cross Validation – Ridge



Appendix I
Choosing Lambda

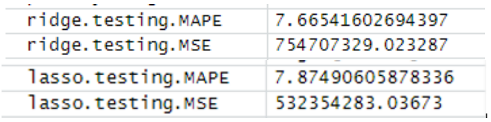


```
> log(penalty.lasso) #see where it was on the graph
[1] -9.367617
> |
```



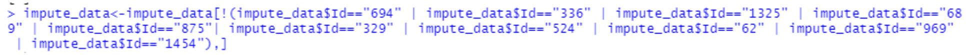
```
> log(penalty.ridge)
[1] -3.390199
> |
```

Appendix J
MSE and MAPE



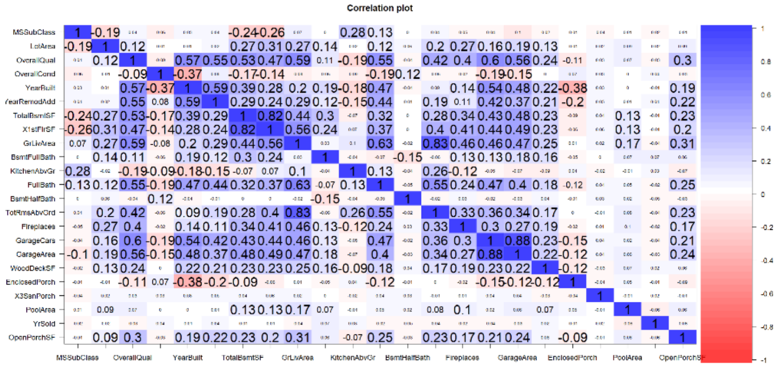
ridge.testing.MAPE	7.66541602694397
ridge.testing.MSE	754707329.023287
lasso.testing.MAPE	7.87490605878336
lasso.testing.MSE	532354283.03673

Appendix K
Removing Outliers

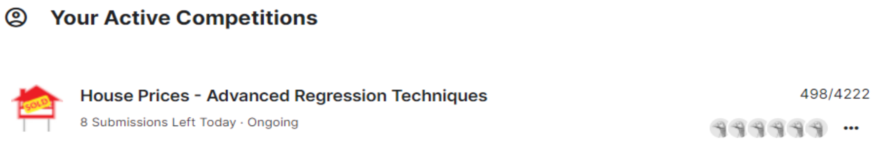


```
> impute_data<-impute_data[!(impute_data$Id=="694" | impute_data$Id=="336" | impute_data$Id=="1325" | impute_data$Id=="68
9" | impute_data$Id=="875"| impute_data$Id=="329" | impute_data$Id=="524" | impute_data$Id=="62" | impute_data$Id=="969"
 | impute_data$Id=="1454"),]
```

Appendix L
Heat Map



Appendix M
Team Carlton’s Position on the Leaderboard

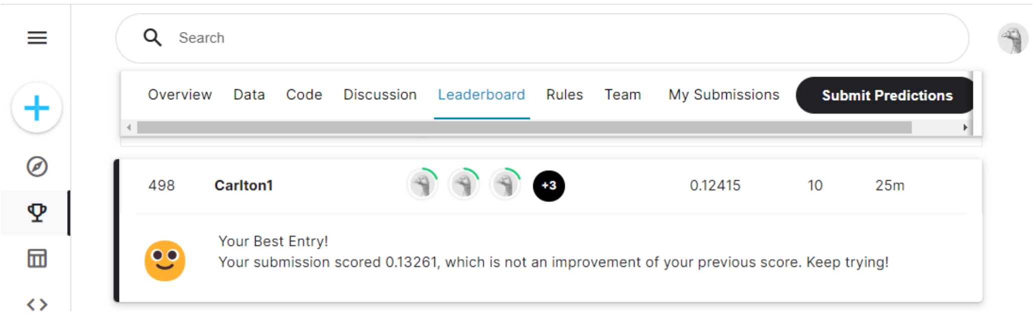


Your Active Competitions

House Prices - Advanced Regression Techniques

8 Submissions Left Today · Ongoing

498/4222



498 Carlton1 +3 0.12415 10 25m

Your Best Entry!
Your submission scored 0.13261, which is not an improvement of your previous score. Keep trying!