House Pricing Prediction

DS502 Final Project

Yufei Lin, Jingfeng Xia, Jinhong Yu, Shijing Yang, Yanze Wang

Nov 29 2020

Introduction

Description of the Problem

Being able to predict the price of a house tends to be an important skill for both the seller and consumers. For the seller, they could make better sales and consumers could have better understanding when they try to make a purchase. Therefore, in this project, we are planning to make prediction of house price based on the 79 different predictors provided by Kaggle dataset to determine values of residential homes in Ames, Iowa. We have noticed Sale Price has a typical right-skewed distribution, and decided to process it using two ways, logrithmic with base e and square root for a distribution that is much closer to the shape of a Gaussian distribution for better performance in models like linear regression. In this analysis, we will perform random forest on original y-value for importance of variables and all the rest of models on both absolute and processed y-values to see which way would each model do better and provide an ensemble of models at the end of our study.

Description of the Dataset

In terms of the dataset, the entire data set consists of two pieces of data organized as training data set and test data set respectively. Whereas for each of the dataset, approximately 80 columns corresponding parameters would be evaluated with the prediction of house price. Some noteworthy predictors include the location classification, utilities, environment of neighborhood, house style and condition, area, year of built, and number of functioning rooms. There are over 1400 row of data points in both the training data set and the test data set. The sale prices in the train dataset are given as a parameter in the form of five or six figure full flat integers. The test data set will be applied to different regression models in order to distinguish the disparities of different model performances.

Approaches

Given that our data is aimed at predicting Sale Price of a house, it is unreasonable to require a model to fit the exact value of the dataset but only to reach an estimation within a certain range. Therefore, we have decided to use both regression and classification approaches to look at the problem on both the original and processed value. For regression method, we are going to look at if a prediction is within the range of the actual price ±5%, we will say it is an accurate prediction. For classification prediction, we will be tagging the data into several different groups, and would be fitting the threshold accordingly with models like SVM and K-Means clustering.

Data Processing

Read in Data

We have chosen to eliminate the Id column from this dataset because Id has nothing to do with our prediction and would mess up our prediction. We save data in “train.csv”" from Kaggle into a variable named **HousePricing** for further processing and we will separate it into training and testing set. For each model Bootstrapping will be performed before each model’s training process.

Data Exploration

```
## [1] "Original training data set has 1460 rows and 84 columns"
```

```
## [1] "The percentage of data missing in the original training data set is 0.28%"
```

```
## [1] "The number of duplicated rows are 0"
```

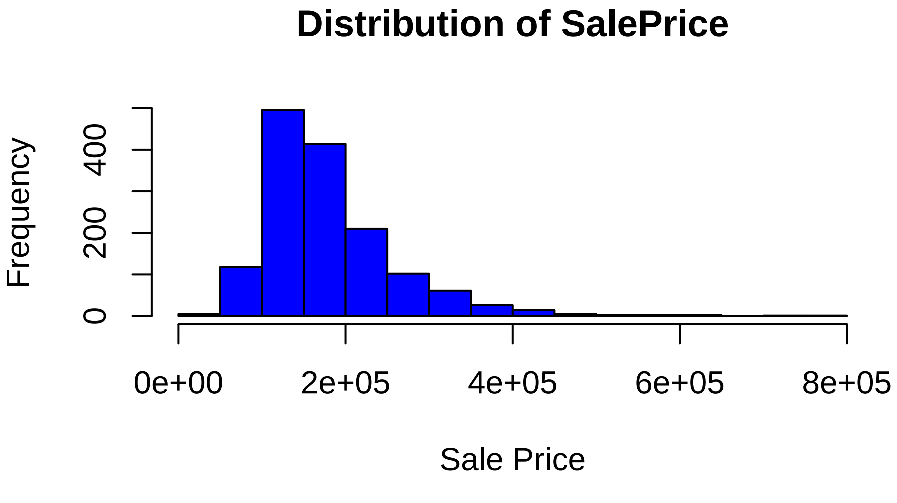
```
## [1] "Number of Factors: 43"
```

```
## [1] "Number of Numeric: 34"
```

target varaible vs. predictors

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   34900  129975  163000  180921  214000  755000
```

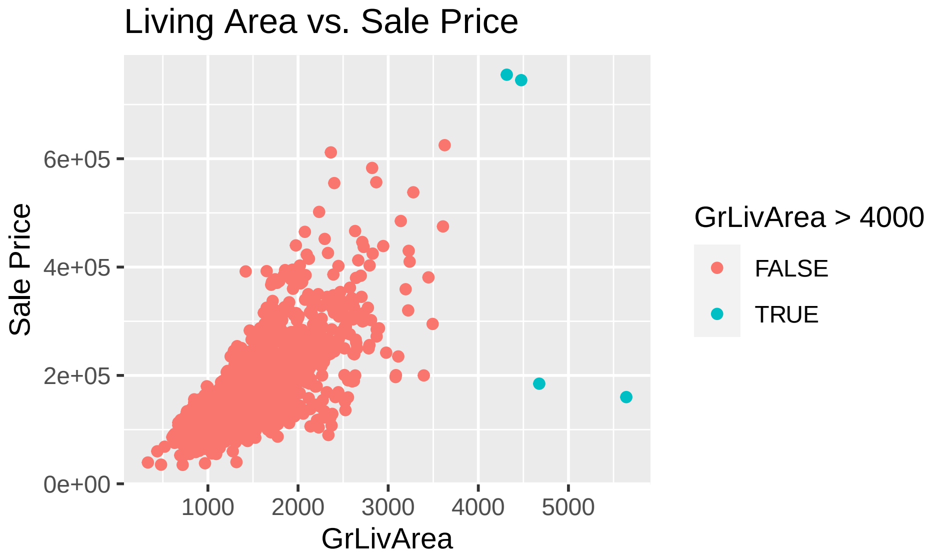
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     334    1130    1464    1515    1777    5642
```

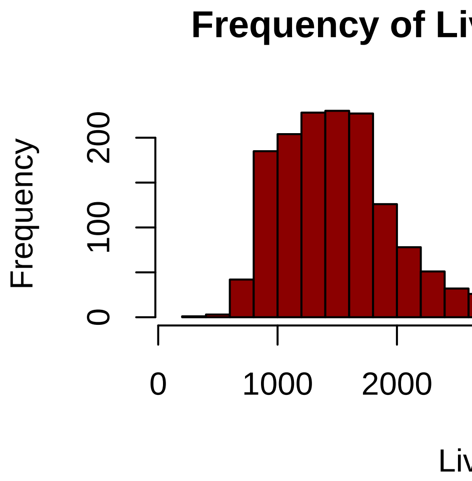


Conclusion

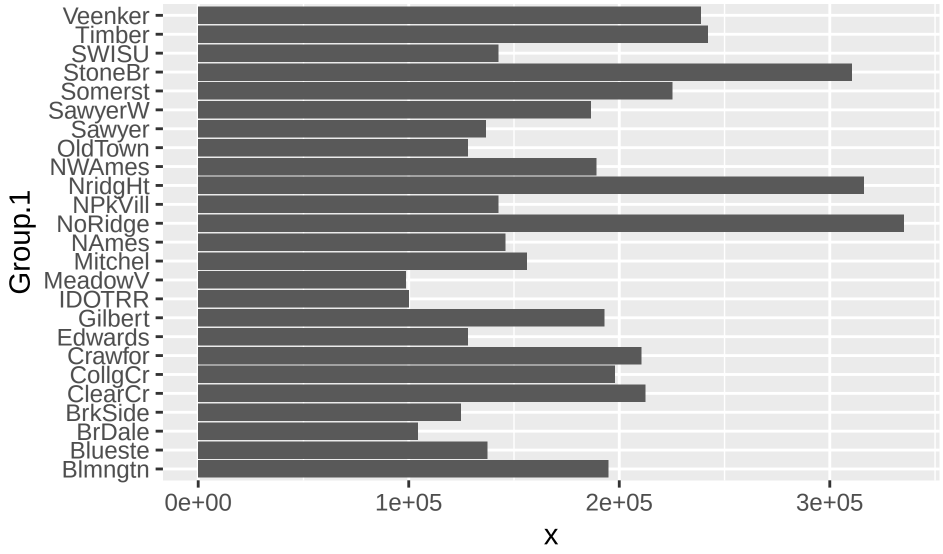
It deviates from normal distribution and it is right skewed

Plotting 'GrLivArea' too see if there are any outliers





Average Price of Each Neighborhood

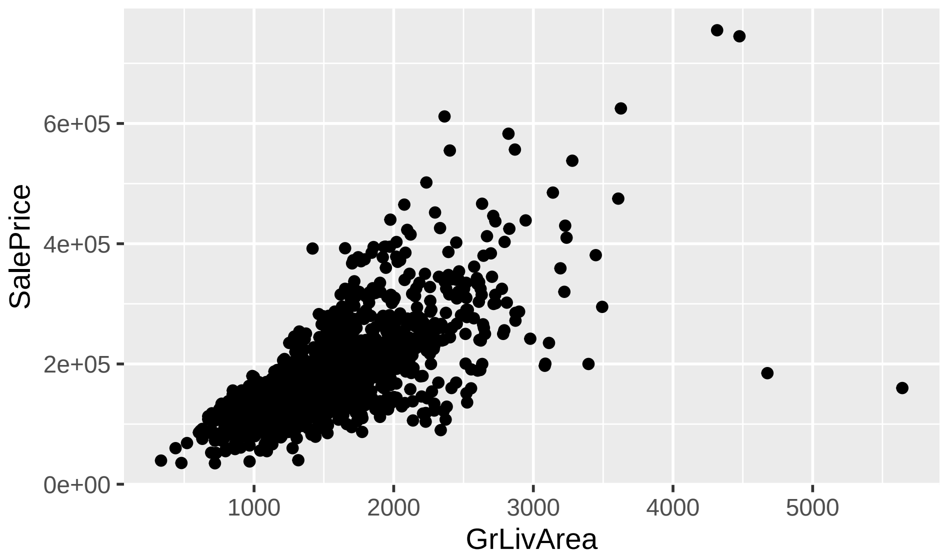


Feature Engineering

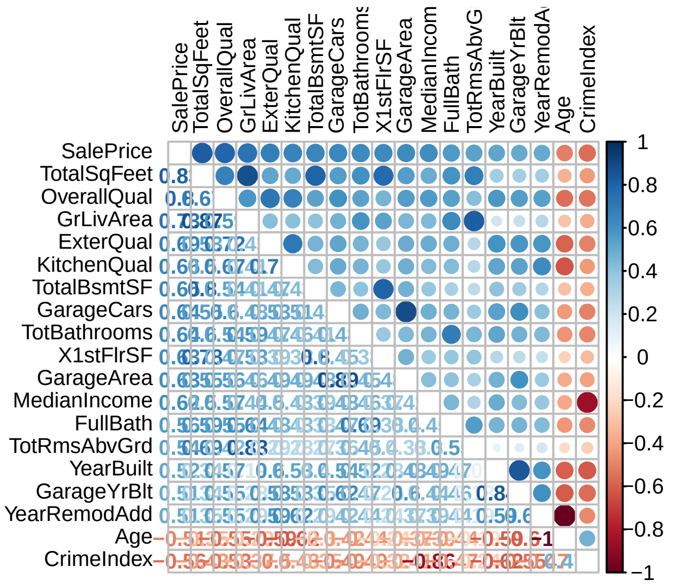
In this section, we convert all missing value based on the following rules:

1. Categorical: fill in most common
2. Numeric: fill in median/average

Convert all train to HousePricing

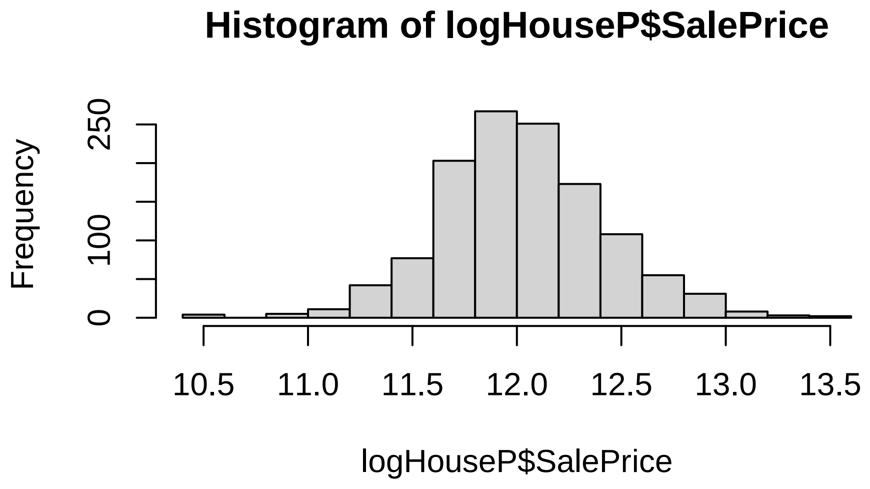


Correlation between the numerical variables

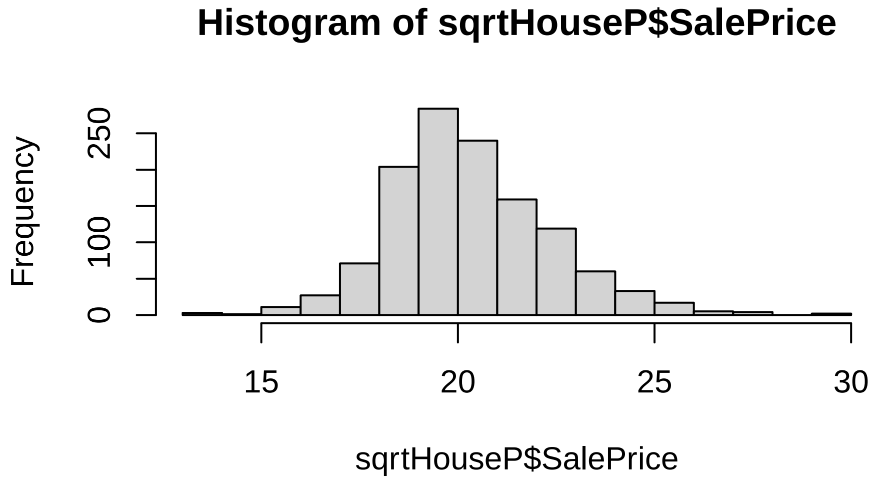


As discussed before, we have decided to use logrithmic with base e and square root to process the data. We have also saved 15% of our data into a variable named vault for the final test of each model.

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   34900  130000  164250  181442  214925  755000
```



```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   10.46   11.78   12.01   12.03   12.28   13.53
```



```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   13.67   18.99   20.13   20.32   21.53   29.48
```

```
##   0   1   2
## 316 614 310
```

```
##   0   1   2
## 316 614 310
```

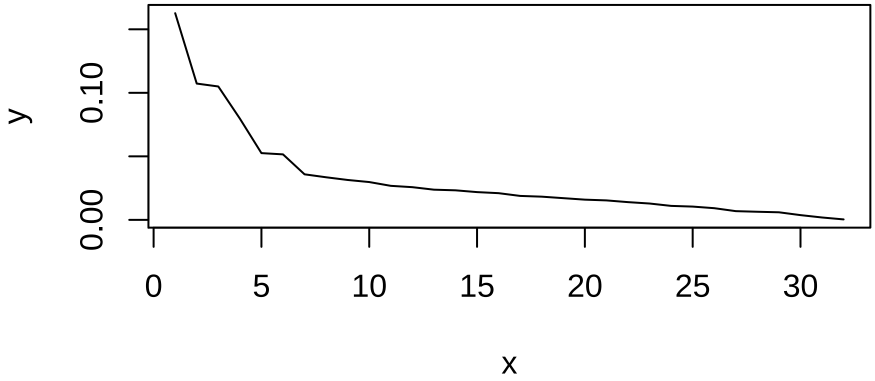
```
##   0   1   2
## 316 614 310
```

PCA

In order to reduce the dimension convincing by finding a proper Manifold, we apply PCA method. Fortunately, we successfully reduced the dimension from 216 to no more than 32. 32 PCs have totally 99.997% of variance proportion. 29 PCs are enough for totally getting 85% of variance proportion.

In the summary we can find all variance proportions behind 32th PC are 0, which means 32 PCs are totally enough important to describe all features of the original data. In fact, we can take less than 32 PCs and find the best number of PCs with cross validation in training process. The sum of (0.1627, 0.1073, 0.105, 0.07973, 0.05255, 0.05155, 0.03586, 0.03352, 0.03141, 0.02978, 0.02679, 0.02574, 0.02376, 0.02327, 0.02185, 0.02102, 0.01885, 0.01826, 0.01709, 0.01591, 0.01533, 0.01396, 0.01287, 0.0110, 0.01046, 0.00921, 0.00687, 0.00639, 0.00596, 0.00374, 0.00187, 0.00037) is 0.99997, which means 32 PCs totally have 99.997% of variance proportion. Moreover, 29 PCs are enough for totally getting 85% of variance proportion.

In this plot, you can see an elbow at 7. Perhaps it can be a good number of PCs.



This is generated by plotting all variance proportions of 32 PCs. We can see the elbow at 7 more clearly than the previous one. However, the square of tail on the right side of 7 is quite thick, which means maybe a number on the tail can be the best one for training. Last but not least, the exact best number can only be revealed by cross validation.

This is to show the effect of dimension reduction and how can we take and use the new data modified by PCA method.

Seperate into Test and Training Set

Spearate by 70% train, 30% test.

Prediction Algorithms

We choose to use PCR, Random Forest, GAM, Lasso and Ridge, Splines and Linear Regression to look at how each model would be suitable for our regression analysis.

Each model needs a cross validation algorithm Remember to report RMSE

Regression Methods

1. Linear Regression

Explanation

We have chosen this model to understand how each numeric variable is linear related to our House Price prediction.

Check Accuracy

```
## [1] 0.3371648 0.2413793 0.2681992 0.2720307 0.2988506

## [1] 0.2835249

## [1] "We have the accuracy of the linear model approximately 28.35%"

## [1] 0.3065134 0.3486590 0.2681992 0.3486590 0.3026820

## [1] "We have the accuracy of the linear model after log transformation approximately 31.49%"

## [1] 0.3295019 0.3678161 0.2950192 0.3256705 0.3141762

## [1] "We have the accuracy of the linear model after sqrt transformation approximately 32.64%"
```

2. Random Forest

Explanation

We have chosen this model because random forest is based on a collection of decision trees that could help us get better understanding of which tree and division contribute to which section such that we could have a better picture of the overall importance of each different factor in the prediction.

Prepare Model

We have 199 independent variables in the data set, therefore we have set mtry(Number of randomly selected variables for each split) to be the square root of that number for maximum performance of the model.

The following is the result from Random Forest algorithm:

Call: randomForest(formula = SalePrice ~ ., data = train_ori, mtry = sqrt(totalIV), importance = TRUE) Type of random forest: regression Number of trees: 500 No. of variables tried at each split: 15

Mean of squared residuals: 1326313726 % Var explained: 77.97

Check Accuracy

We then need to check accuracy, as assumed before, we would look at whether the predicted data is within the ±5% range. The following is the result.

```
## [1] "We have the accuracy of the model approximately 32.26%"
```

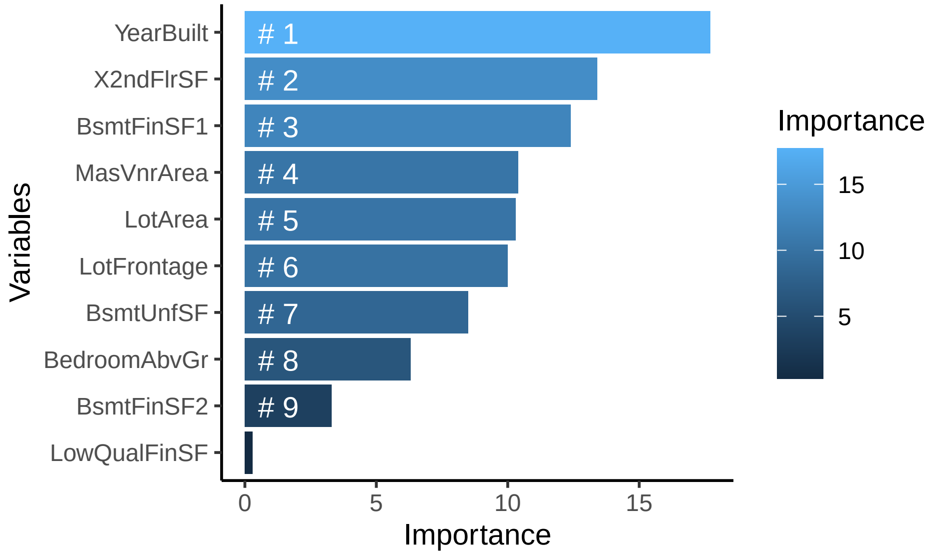
Error Metrics

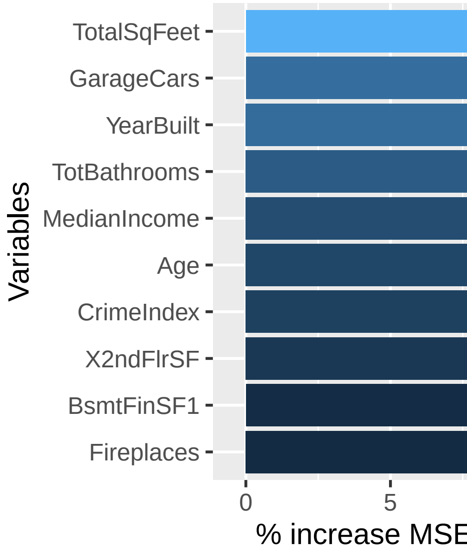
Then let us take a look at the MSE of this model:

```
## [1] "We have the MSE of the model approximately equal to 1180197621.75171"
```

Variable Importance

Here we are going to show the top 10 most important variables in predicting sale price of a house.





From the random forest analysis, we have discovered that the top three most important factors for predicting sale price are the following:

1. OverallQual (Overall Quality of the building)
2. ExterQual (Evaluates the quality of the material on the exterior)
3. YearBuilt (The year the house is built)

Cross Validation

In the cross validation, we have chosen to look at R^2, RMSE and MAE.

R2	RMSE	MAE
0.8804	34354	22307

3. PCR (Iris)

Cross Validation

4. Ridge Regression

Explanation

The reason we choose Ridge regression model is Ridge regression is very similar to linear regression, both try to minimize the RSS, but ridge regression has a penalty term, this could help us to prevent overfitting when add more predictors.

Prepare Model

1. Bootstrap Training Data

2. First, we set initial alpha to 1 to fit the ridge regression,and set the values of initial lambda ranging from 10^10 to 10^(-2), essentially covering the full range of scenarios from the null model containing only the intercept, to the least squares fit.

```
## [1] 22867.61
```

1. Then we use cross validation to choose the optimal lambda for Ridge Regression, as the following:

```
## [1] 6752.752 6804.832 5782.262 6978.871 6145.474

## [1] 6492.838

## [1] 0.06626504 0.03246681 0.03653565 0.03100590 0.02907742

## [1] 0.03907016

## [1] 0.1795723 0.1693713 0.1581334 0.1620771 0.1500434

## [1] 0.1638395

## [1] 6492.838

## [1] 0.03907016

## [1] 0.1638395
```

Check Accuracy

We then need to check accuracy, as assumed before, we would look at whether the predicted data is within ±5%. The following is the result.

```
## [1] 0.2413793 0.3141762 0.2911877 0.2413793 0.2605364

## [1] 0.2697318

## [1] "Accuracy of Ridge is approximately 30.80%"

## [1] 0.3026820 0.2873563 0.3103448 0.3333333 0.3065134

## [1] "Accuracy of Ridge with Log Transformation is approximately 30.80%"

## [1] 0.3218391 0.2835249 0.3448276 0.3448276 0.2681992

## [1] 0.3126437

## [1] "Accuracy of Ridge with Sqrt Transformation is approximately 31.26%"
```

Cross Validation

Then let us take a look at the MSE of this model:

```
## [1] 30064.67 35728.77 33728.06 33525.17 36288.48
```

R2	RMSE	MAE
0.8122	33867	22868

```
## [1] 27816.16 27470.76 35349.08 31146.25 38892.26
```

R2	RMSE	MAE
0.8383	32135	20242

```
## [1] 664590285803 367770414726 371720863171  87440601997  90945258270
```

R2	RMSE	MAE
0.2493	3.165e+11	3.37e+10

5. Lasso Regression

Explanation

Lasso regression is pretty similar to Ridge regression. But compare to ridge, Lasso is more interpretable. It will make some predictors' coefficient to be exactly 0, which could help us find out which predictor is what Lasso thinks is important.

Prepare Model

1. Set the initial alpha is equal to 1 (Ridge regression is 0), and also use the same initial lambda, then try to use cross validation to choose the optimal lambda for Lasso.
2. With cross validation, we find out the optimal lambda as following:

```
## [1]   68.64116   77.22762  825.63063   35.86498 2239.35939

## [1] 649.3448

## [1] 0.0003274430 0.0030740835 0.0008898992 0.0027045694 0.0061688537

## [1] 0.00263297

## [1] 0.032822552 0.015480962 0.011332340 0.009074930 0.006957921

## [1] 0.01513374

## [1] 649.3448

## [1] 0.00263297

## [1] 0.01513374
```

Coefficient From Lasso Regression

Here we are going to show the predictors lasso choosed.

```
##  [1] "(Intercept)"   "TotalSqFeet"   "MedianIncome"  "GarageCars"
##  [5] "YearBuilt"     "CollegeDegree" "MasVnrArea"    "CrimeIndex"
##  [9] "PoolArea"      "LotArea"

##  [1] "(Intercept)"  "TotalSqFeet"  "GarageCars"   "MedianIncome" "YearBuilt"
##  [6] "TotBathrooms" "Fireplaces"   "BsmtFinSF1"   "X2ndFlrSF"    "LotFrontage"

##  [1] "(Intercept)"  "TotalSqFeet"  "YearBuilt"    "GarageCars"   "MedianIncome"
##  [6] "Fireplaces"   "LotArea"      "X2ndFlrSF"    "TotBathrooms" "LotFrontage"
```

Check accuracy

We then need to check accuracy, as assumed before, we would look at whether the predicted data is within the ±5% range. The following is the result.

```
## [1] 0.2030651 0.2183908 0.2720307 0.3026820 0.3141762

## [1] 0.262069

## [1] "Accuracy of Lasso is approximately 26.21%"
## [1] 0.3103448 0.2835249 0.3141762 0.3716475 0.3678161
```

```
## [1] "Accuracy of Lasso with Log Transformation is approximately 32.95%"
```

```
## [1] 0.3524904 0.3218391 0.3563218 0.3295019 0.3793103
```

```
## [1] 0.3478927
```

```
## [1] "Accuracy of Lasso with Sqrt Transformation is approximately 34.79%"
```

Cross Validation

Then let us take a look at the MSE of this model:

```
## [1] 35917.17 27339.31 34374.34 33827.18 31733.55
```

R2	RMSE	MAE
0.8109	32638	22158

```
## [1] 28954.97 26474.29 28823.85 28235.98 28533.81
```

R2	RMSE	MAE
0.8656	28205	18989

```
## [1] 27869.52 24407.86 27079.90 26841.90 27504.39
```

R2	RMSE	MAE
0.8583	26741	17851

6. GAM

Explanation

We have chosen GAM as one of our models because it produces an analysis on those factors that have less linear relationship with the result, for instance LotFrontage, YearRemodAdd, and MasVnrArea that are having relatively high importance but also high p-value that makes them not very linear related to SalePrice.

1) GAM1

In this model, we have LotFrontage, YearRemodAdd and MasVnrArea as predictors, with YearRemodAdd having a degree of freedom 2. We obtain the following result:

2) GAM2

In this model, we have LotFrontage, YearRemodAdd and MasVnrArea as predictors. None of them has a degree of freedom in the fit. We obtain the following result:

3) GAM3

In this model, we have LotFrontage, YearRemodAdd and MasVnrArea as predictorswith LotFrontage having a degree of freedom of 3. We obtain the following result:

GAM Summary

We then take an ANOVA test to understand which model is the best and we have the following result:

We can see that from the anova test that P-value for the second model is the smallest, therefore, it is the most preferred.

Cross Validation

Then, we conduct a cross-validation on the second model only.

Evaluation of different models

Root MSE

Choose best fit model

Conclusion

1. Classfication

Discussion & Future Development

Resources