PSL (F20) Project 1

Vijayakumar Sitha Mohan (VS24), Waitong Matthew Leung (wmleung2)

13-Sept-2020

Introduction

The goal is to predict the final price of a home (in log scale) using AMES housing dataset. As a project requirement we need to build Two prediction models and we chose Lasso/Ridge Regression and XBGBoost as our model to predict the sale price.

Data Exploration

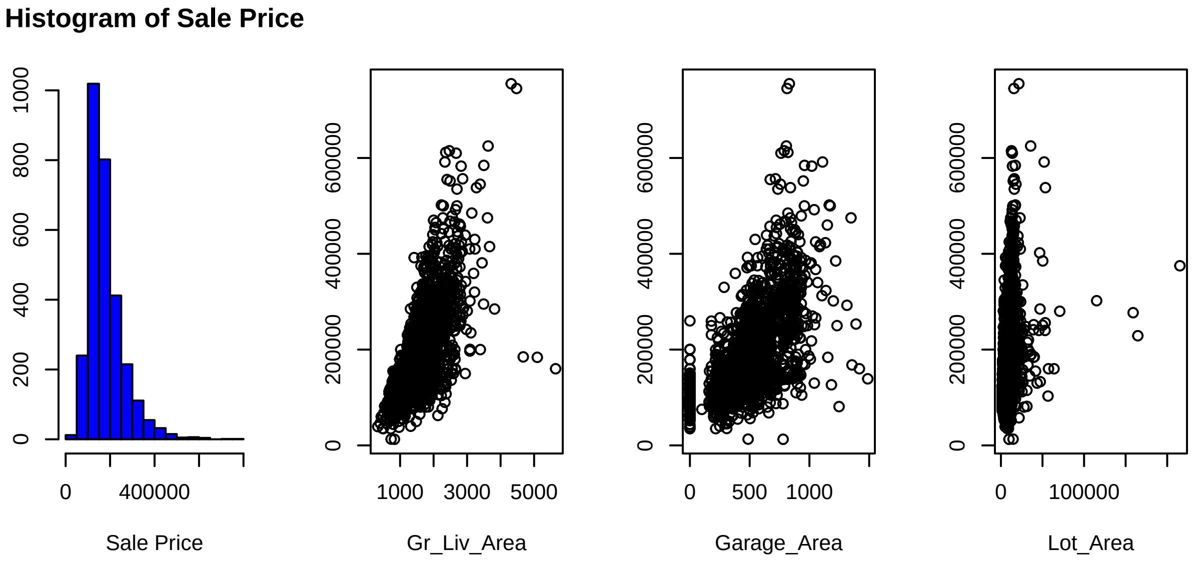
The dataset has 2930 rows (i.e., houses) and 83 columns. The first column is "PID", the Parcel identification number; The last column is the response variable, Sale_Price; The remaining 81 columns are explanatory variables describing (almost) every aspect of residential homes.

- Analysis of Response Variable(Sale Price)

  A mean and median sale price of $180,796 and $160,000 respectively, indicating a positive skew, which was confirmed by histogram of the Sale Price.

  A long right tail tells that there are outlier data points. max = 755,000 and min = 12,789

- Skewness also in quite a few variables



Overall Approach

We chose the following two models to tackle this home price prediction problem. The general approach of both of the models are in the sequence of data prepreprocessing, optimal hyperparameters search, train model based on train data set and finally predict sale price with test data set.

- Model 1: Lasso and Ridge mixed model. We use Lasso 10-fold cross validation to select relevant variables and lambda value first. Subsequently, We use selected variables and lambda value from Lasso to train Ridge model in order to remove colinearity of remaining variables.
- Model 2: XGBoost model. We spent quite a lot of time (~16 hours) to search for the optimal value for the following hyperparameters, namely eta, max_depth, subsample. We randomly generated 10,000 combination of the three hyperparameters to select the set that give us minimal RMSE.

Data PreProcessing

- Remove some predictors with imbalance data or can be inferred from other predictors, i.e. ‘Street’, ‘Utilities’, ‘Condition_2’, ‘Roof_Matl’, ‘Heating’, ‘Pool_QC’, ‘Misc_Feature’, ‘Low_Qual_Fin_SF’, ‘Pool_Area’, ‘Longitude’,‘Latitude’.
- Impute Missing Values with median value using Caret package
- Winsorize numerical variables by 95 percentile value
- Encode categorical variables to numerical value

- For model 2 XGBoost model, remove outliners in train set based on cook's distance (remove train data with cook's distance > 4/n)

Model Building

Model 1: LASSO/Ridge Regression Model

- Lasso 10 Fold Cross Validation to choose Lamda hyperparameter
- Lasso Feature Engineering to select variables with non zero coefficients
- Ridge Fit & Predict using optimal hyperparameters

Model 2: XGBoost - Boosting Model

- Train model to choose eta, subsample, max_depth hyperparameter: Ran model selection by running 10000 times and chose the lowest test rmse and corresponding hyperparameter values. Run time was around ~16 hours
- Optimal hyperparameters: (eta = 0.0474239, max_depth = 4, subsample = 0.541795)
- Fit & Predict using optimal hyperparameters

Results

- The below two tables shows the individual split train and test errors for both models and other table shows the mean errors of the same.
- 10 Train & Test Split Error for Lasso & XGBoost Model

Split #	Lasso Test Error	XGBoost Test Error
1	0.1149	0.0694
2	0.1203	0.0724
3	0.1130	0.0653
4	0.1204	0.1173
5	0.1086	0.0704
6	0.1292	0.0694
7	0.1345	0.0724
8	0.1280	0.0653
9	0.1340	0.1168
10	0.1241	0.0704

- Mean Train & Test Error for Lasso & XGBoost Model

```
##   lasso.train.error lasso.test.error xgboost.train.error xgboost.test.error
## 1            0.1019           0.1227             0.06161            0.07891
```

- Running time

System: MSI laptop, Intel i7, 2.60GHz, 16GB, Win 10

Lasso total train & test time (second)	XGBoost total train & test time (second)
4.863	4.951

Discussion

These are important observations we had during model building.

1. We were not able to satisfy bench mark RMSE requirement for regression model by using only Lasso or Ridge regularization alone. We ended up need to have extra train data preprocessing to remove outliners based on cook's distance. We also need to use Lasso to drop insignificant variables, and cross-validation to get lambda min first. We then use Ridge to regularize the remaining variables which have colinearity among them.
2. The hyperparameter tuning for XGBoost runs too long and to tune Lambda, MaX_Depth and Sub-Sample, took 16 hours to find the optimal value by running 10000 models in my laptop. Need to work on parallelizing the code with doMC package or SParkR to improve the performance.
3. There are quite a few variables with skewness. We tried to put in transformation to skewed variables but no improvement in prediction accuracy.