**STA 141C Final Project Report**

Jiahui Li (999825419) | An Liu (912308764) | Yingtong Chen (914382733)

**Using Three Regression Algorithms to Predict House Price**

**Introduction**

Predicting housing price has always been a popular topic. We found a housing dataset on Kaggle and would like to analyze the dataset for our final project. The train.csv dataset includes 79 explanatory variables describing almost every aspect of residential homes in Ames, Iowa, and the sale price variable. In this project, we are going to focus on solving the problem of predicting house prices for house buyers and house sellers using different regression algorithms. We will compare three different methods, lasso regression, ridge regression and gradient boosting regression to get the best model that is the most effective and has the highest accuracy rate for predicting the house price. The report is organized in three sections. We first need to prepare the data for our analysis, such as replacing the missing values and separate the quantitative and qualitative variables. The second section is the three regression analyese. In the last section, we will perform cross validation to compare model performance.

1. **Data Processing**

Our train.csv data contains 1460 observations, which we split the data into 80% train, and 20% test data for our regression algorithm. The data contains explanatory variables such as ID, MSSubClass, LotArea, etc. Since some of the variables are quantitative and some are qualitative, we then separately analyze the relationship between the variables and the Sale Price for both quantitative and qualitative variables. First of all, we prepared the train data by replacing the missing values for the quantitative with the median value of its column. Then we calculated the correlation matrix. We decided to choose the quantitative variables that have high correlation(at least > 0.4), which are below list of index.

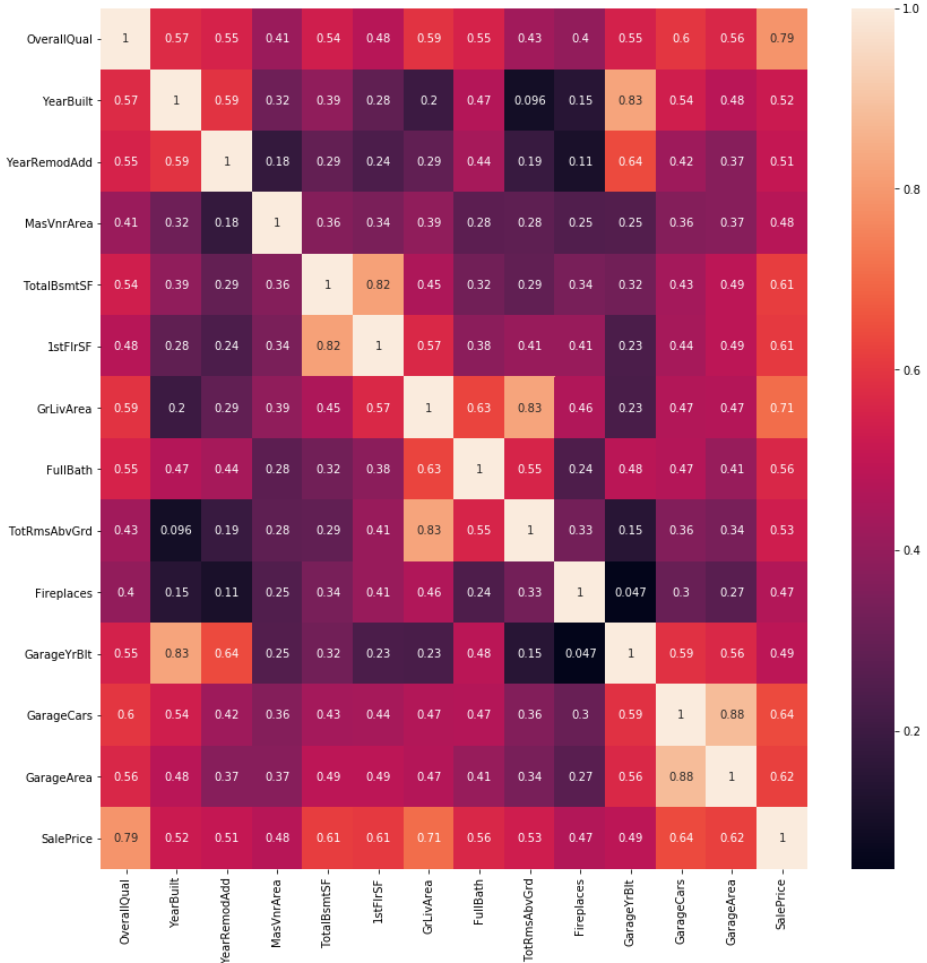
Index(['OverallQual', 'YearBuilt', 'YearRemodAdd', 'MasVnrArea', 'TotalBsmtSF',

'1stFlrSF', 'GrLivArea', 'FullBath', 'TotRmsAbvGrd', 'Fireplaces',

'GarageYrBlt', 'GarageCars', 'GarageArea', 'SalePrice'],

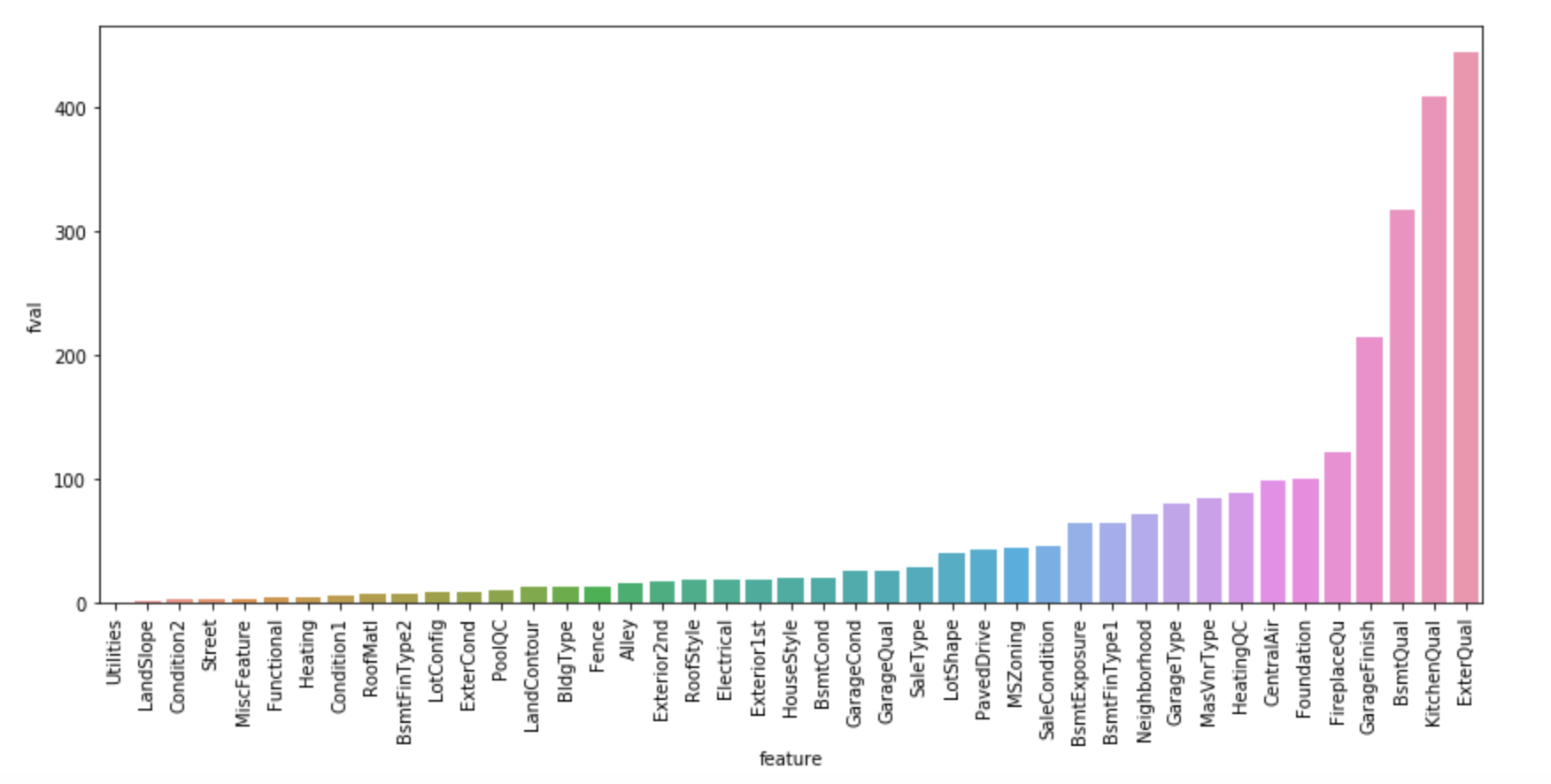
dtype='object')

We also tested for multicollinearity within these quantitative variables, which we got the below heatmap.



We confirmed that there are four pairs of variables that are highly correlated with each other; the coefficient is higher than 0.8. They are TotalBsmtSF(Total square feet of basement area) and 1stFlrSF(First Floor square feet), GrlivArea(Above ground living area square feet) & ToRmsAbvGrd(Total rooms above grade that does not include bathrooms), GarageCars(Size of garage in car capacity) & GarageArea(Size of garage in square feet), and GarageYrblt(Year garage was built) & YearBuilt(Original construction date). Since the pairs are highly correlated with each other, we can eliminate one in each pair. Our final explanatory quantitative variable set is ['OverallQual', 'YearBuilt', 'YearRemodAdd', 'MasVnrArea', 'TotalBsmtSF', 'FullBath', 'TotRmsAbvGrd', 'Fireplaces','GarageArea']. This list makes intuitive sense because the area of the home and the year built will directly impact the SalePrice.

Also, we separately analyze the association between the individual qualitative variables with the SalePrice. Since the SalePrice is a continuous variable, and the explanatory variables are categorical, we will use ANOVA one-way test to see the relationship. Below is the visualization of the strength of association based on the f-value. It clearly shows that the quality of the feature of the house matters.

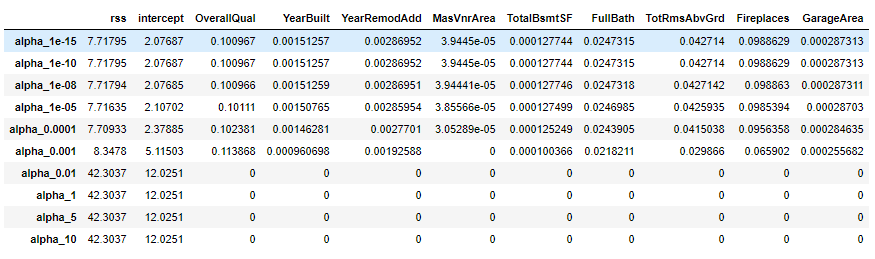


1. **Modeling**

There are many options we can choose when selecting a predicting model. However, the goal is to find a model that is most efficient this dataset and has a highest accuracy rate. It is difficult to know which model fit the best to this particular data, so we select three of the common regression algorithms and make comparisons between them. The three algorithms used here are lasso regression, ridge regression, and gradient boosting regression. On each algorithm, we will start with getting the coefficients for the selected quantitative variables. The selected quantitative variables are from 14 highest correlated variables and remain to 9 after removing multicollinearity. For our analysis, we will be testing on different alphas to get their residual sum of squares (rss) value and see which alpha has the lowest rss. Finally, we will use the score function from scikit-learn and cross validation to get the accuracy rate and conclude our findings.

***Lasso Regression***

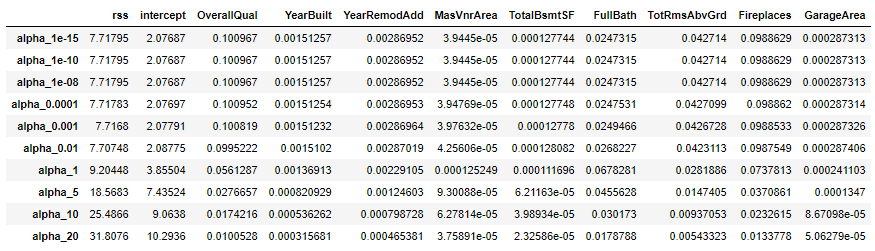
The first predicting model we used is Lasso Regression. The word “LASSO” stands for Least Absolute Shrinkage and Selection Operator which it adds up the absolute value of magnitude of the coefficients to penalize its loss function. One of the computational advantages about lasso regression is that it performs feature selections thus has very well handle on high dimension data. Lasso regression automatically drops the features that are highly correlated and reduce the features’ coefficient to zeros. Below is a table of coefficients and rss produced by lasso regression. From the table we can see when alpha gets larger, more features has zeros on their coefficient. At alpha equals to 1\*e^-15 where none of the feature coefficients is zero, rss turns out to be the smallest. Since we already manually picked out the features that are highly correlated and the dimension of dataset is relatively small, using lasso regression is not optimal in this case. It’s clear to see when alpha goes smaller, rss eventually gets smaller.



Therefore, we use an alpha = 0.001 that minimize the rss and numbers of variables used in the regression. The accuracy score is 0.791. We will use this result and compare it to the other models.

***Ridge Regression***

The second model we picked to make predictions is ridge regression. Many people uses ridge regression to compare with lasso regression because they have similar model structure but unique advantages. Ridge regression performs L2 regularization which adds the squared magnitude as penalty to the loss function. Since ridge regression includes all the features, it shrinks the coefficients and reduces the model complexity. However, including all the features will be challenging when data dimension is large and not efficient to deal with sparse data. When creating a table with coefficients, we see the coefficients are larger with small alpha values. Rss increase when alpha increase is meaning that the model is underfitting with high alpha values. We also notice some of the coefficients are so low that they are getting close to zero. These small, nonzero values can distinguish the difference between lasso and ridge regression. Below is the coefficient table for each variables.

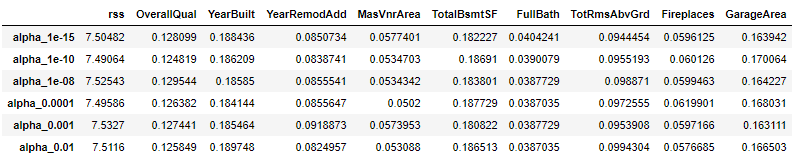


For ridge regression, we did the similar things to get the ridge score and accuracy rate. We set alpha equals to 0.001, and the accuracy score is 0.833, which is slightly higher to the ones in lasso regression

***Gradient Boosting Regression***

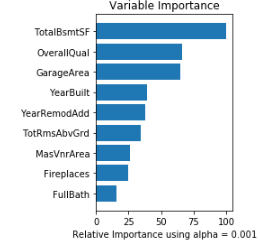
Unlike lasso and ridge regression, Gradient Boosting Regression uses decision trees to make predictions. Gradient boosting has been widely used in kaggle competitions because of its high performance in prediction. Decision trees can handle non-linear relationships which is not an option for lasso and ridge regression. However, gradient boosting builds trees one at a time which is time consuming and can cause overfitting.

Below table shows the important feature value for each variables using the feature\_importances\_ attribute under GradientBoostingRegressor.



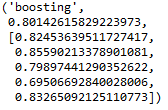
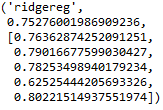
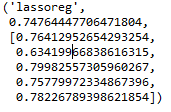
For Gradient Boosting Regression, again, setting alpha equals to 0.001, at one instance, the accuracy score is 0.827.

Below graph is showing the feature importance using alpha = 0.001. We can see that the TotalBsmSF is the most important feature in predicting the SalePrice. This also makes intuitive sense because the living area certainly plays a big role in determining the price of a home. We now use cross validation to compare the performance of the three different algorithms.



1. **Prediction Evaluation of the Models & Conclusion**

We performed 5-fold cross validation on all three methods and get their accuracy rate. Below shows the result of the mean score and each subsection accuracy rate for each model. We can see that at the same level of alpha of 0.001, Lasso Regression has the lowest accuracy rate, following is the Ridge regression, and Gradient Boosting has the highest.



In the analysis, we saw that the smaller the alpha is, the lower the RSS. Bigger alpha will leads to underfitting. We also conclude that the total area is the most important feature in predicting sales price of a house based on Gradient Boosting Regression.

1. **Reference**

**Code:** sta141c\_final.py

**Data**:h[ttps://www.kaggle.com/c/house-prices-advanced-regression-techniques](https://www.kaggle.com/c/house-prices-advanced-regression-techniques)

**Scikit-learn Machine Learning Algorithm:** <http://scikit-learn.org/stable/supervised_learning.html#supervised-learning>

# **A Complete Tutorial on Ridge and Lasso Regression in Python:** <https://www.analyticsvidhya.com/blog/2016/01/complete-tutorial-ridge-lasso-regression-python/>

**A Gentle Introduction to the Gradient Boosting Algorithm for Machine Learning**

https://machinelearningmastery.com/gentle-introduction-gradient-boosting-algorithm-machine-learning/