141C Final Project

Team members:
Deng Zeyu
Ricardo Rendon
Julia Stelman
Wei-Kaung Lin

Project topic:
House price prediction challenge
https://www.kaggle.com/c/house-prices-advanced-regression-techniques

I. Data Exploration
In the dataset, there are 19 discrete variables. There are 27 categorical variables which we have formatted as ordinal. There are 32 non-ordinal categorical variables.There 3 continuous variables.

Missing value
Missing value percentage of variables: MiscFeature have more than 50% of missing values, so it was removed from prediction. For most other variables, a mean value was used to replace the missing value. The exception was GarageYrBult, for which missing values were replaced with the mean - 3.5* standard deviation.

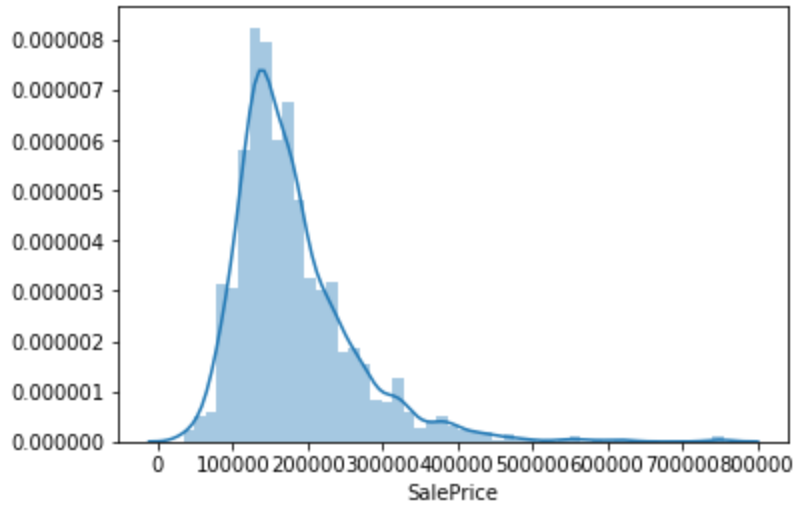
	Variable	Missing Percentage
3	LotFrontage	0.177397
25	MasVnrType	0.005479
26	MasVnrArea	0.005479
42	Electrical	0.000685
59	GarageYrBlt	0.055479
74	MiscFeature	0.963014

Analysing distribution of the response variable, Sales Price:
First, we want to analysing the sales price in this area. Housing price is the most important things need to consider when people buying a house.

Here is our result:

count 1460.000000
mean 180921.195890
std 79442.502883
min 34900.000000
25% 129975.000000
50% 163000.000000
75% 214000.000000
max 755000.000000

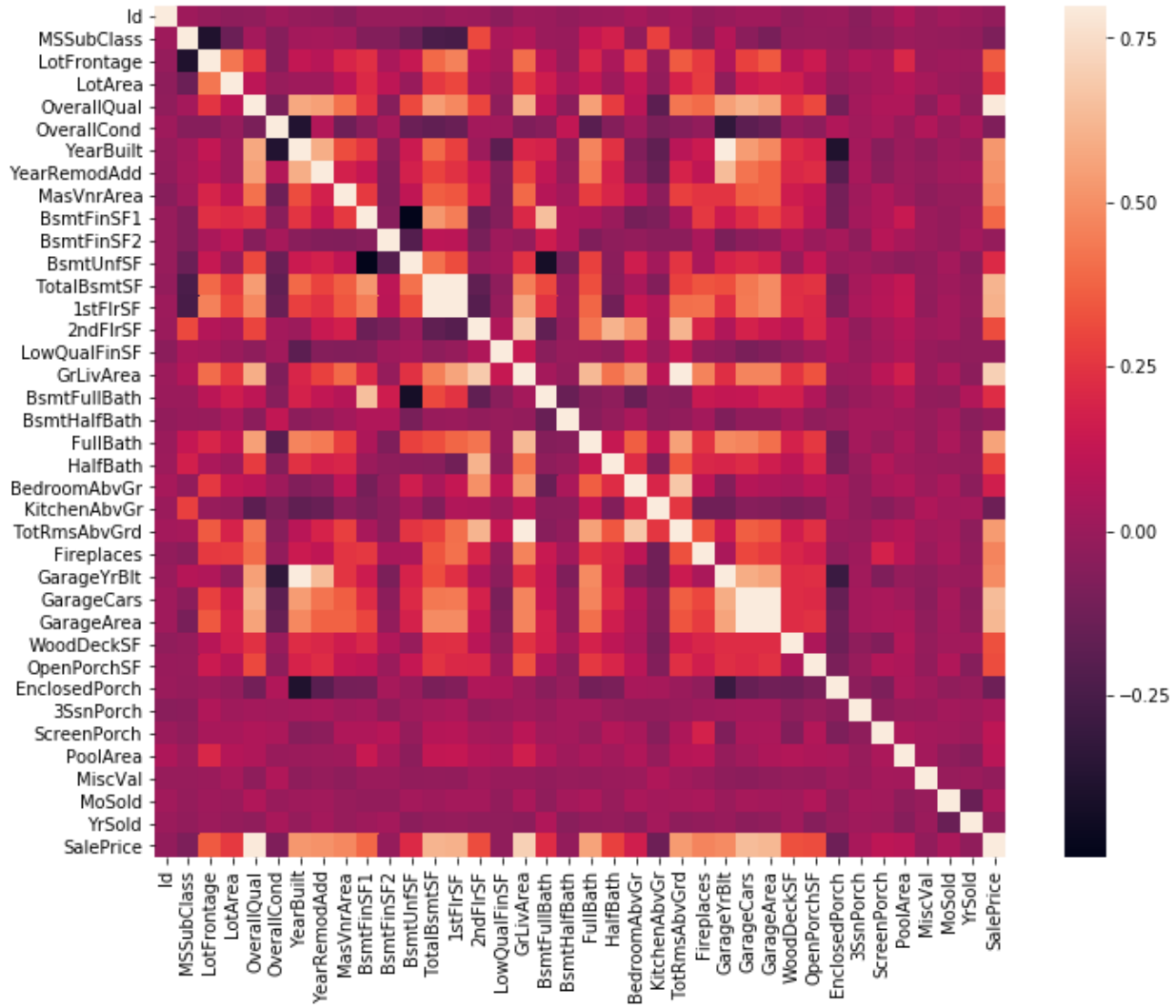
We have found the minimum price is 34900, it is larger than 0, so we can say there is no impossible data on this variable.



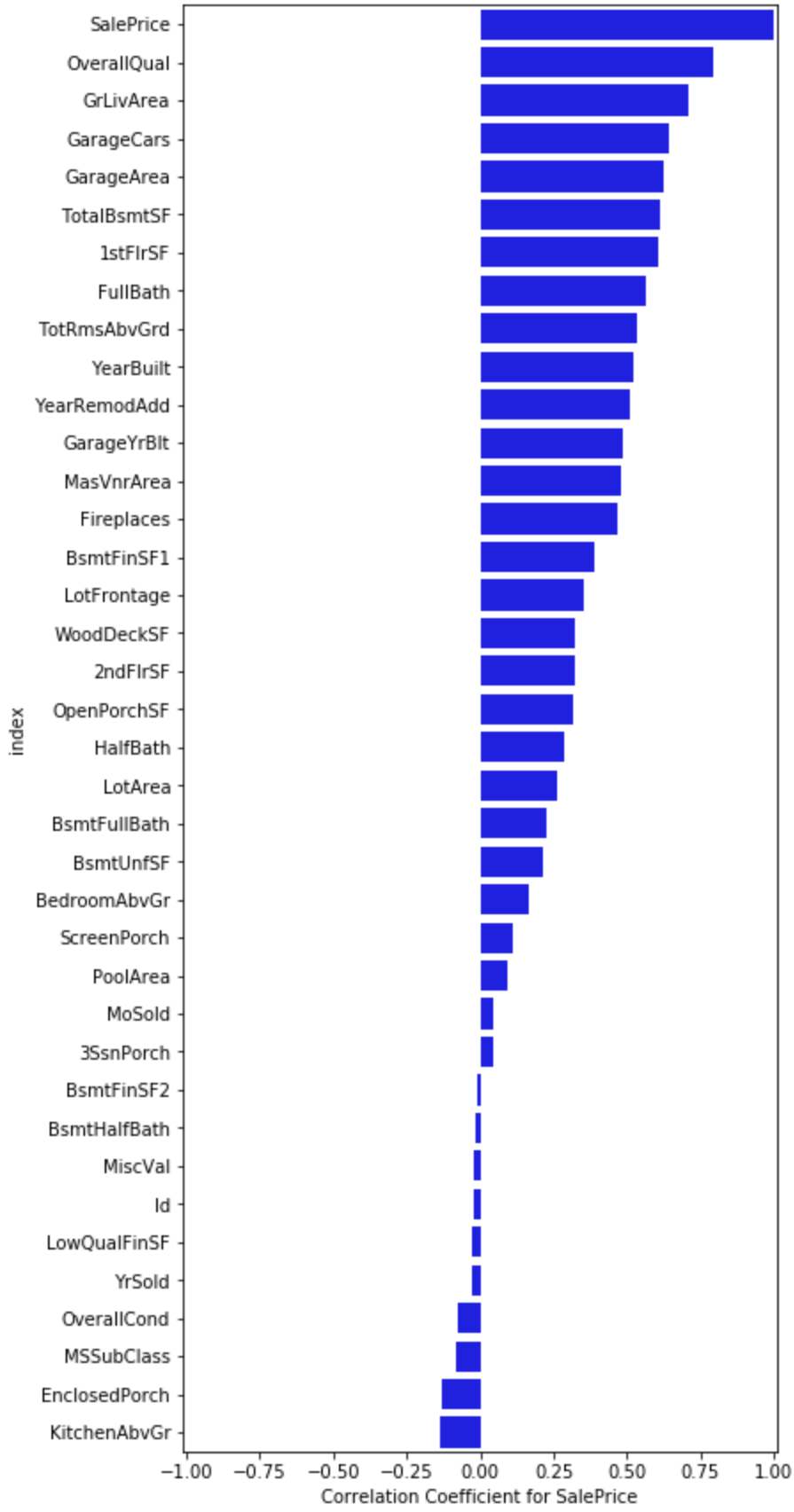
Skewness: 1.882876
Kurtosis: 6.536282

As shown by the plot above, the distribution of sale price is skewed right. Since the distribution is not totally normal, we chose to use a non-parametric approach, Kruskal-Wallis, as one of our methods.

Correlation analysis and the heatmaps for numeric and ordinal-categorical variables



The heatmap is the best way to display the relationship between every pair of variables. There are several off-diagonal white squares in this heatmap. For example, between the variables “GarageYrBlt” and “YearBuilt”. Another is between “GarageCars” and “GarageArea”. This is because there is a very high correlation within each of these pairs, as is suggested by intuition (e.g., often times, an old house has an old garage). Essentially, the information the first item in a given one of there pairs provides is almost the same as the information provided by the second item in the same given pair. So we can expect to find a high presence of multicollinearity at the sites of these pairs. Notice how the bottom edge and the right edge of the heatmap stand out by being lighter in hue compared to their surroundings. The squares in these locations represent correlations between our explanatory variables and our dependent variable. The higher these correlations are, the better. In this case, the heatmap shows an encouraging trend.
For the dependent variable “SalePrice”, we can see strong relationships between it and “OverallQual”, “GrLivArea”, “GarageCars”, “TotalBsmtSF”, and “YearBuit”, among others.


SalePrice
OverallQual
GrLivArea
GarageCars
GarageArea
TotalBsmtSF
1stFlrSF
FullBath
TotRmsAbvGrd
YearBuilt
YearRemodAdd
GarageYrBlt
MasVnrArea
Fireplaces
BsmtFinSF1
LotFrontage
WoodDeckSF
2ndFlrSF
OpenPorchSF
HalfBath
LotArea
BsmtFullBath
BsmtUnfSF
BedroomAbvGr
ScreenPorch
PoolArea
MoSold
3SsnPorch
BsmtFinSF2
BsmtHalfBath
MiscVal
Id
LowQualFinSF
YrSold
OverallCond
MSSubClass
EnclosedPorch
KitchenAbvGr
index
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00
Correlation Coefficient for SalePrice

The correlation coefficients for “SalePrice” are represented in the bargraph above. Clearly, “OverallQual”, “GrLivArea”, “GarageCars”, “GarageArea”, “TotalBsmtSF”, “1stFirSF”, “FullBath”, “TotRmsAbvGrd”, “YearBuilt”, “YearRemodAdd”, “GarageYrBit”, “MasVnrArea”, “MasVnrArea”, “Fireplaces”, “BsmtFinSF1”, “LotFrontage”, “WoodDeckSF”, “2ndFlrSF”, “OpenPorchSF”, “HalfBath”, “LotArea”, “BsmtUnfSF”, “BedroomAbvGr”, “ScreenPorch”, “PoolArea”, “MoSold” and “3SsnPorch” have positive relationship.
“BsmtHalfBath”, “MiscVal”, “ Id”, “LowQualFinSF”, “YrSold”, “OverallCond”, “MSSubClass”, “EnclosedPorch” and KitchenAbvGr” are negative correlation. (Id is negligible as an explanatory variable because it is only an arbitrary index introduced in the data generating phase.)

Kruskal-Wallis Hypothesis Test for non-ordinal categorical variable
Null hypothesis (same for each of the categorical variables we test here):
H_0: All distributions are equal
Alternative hypothesis:
H_1: At least one distribution is different from one other distribution

Since the sale price is not a normal distribution, a non parametric test is appropriate to test whether there is significant difference among groupings of SalePrice by the individual categories of each given categorical variable. There are three non-ordinal categorical variables where there is no significant difference in sale price. These variables are: Street, LandSlope, and Condition2.

II. Method:

1. Regression/Lasso:

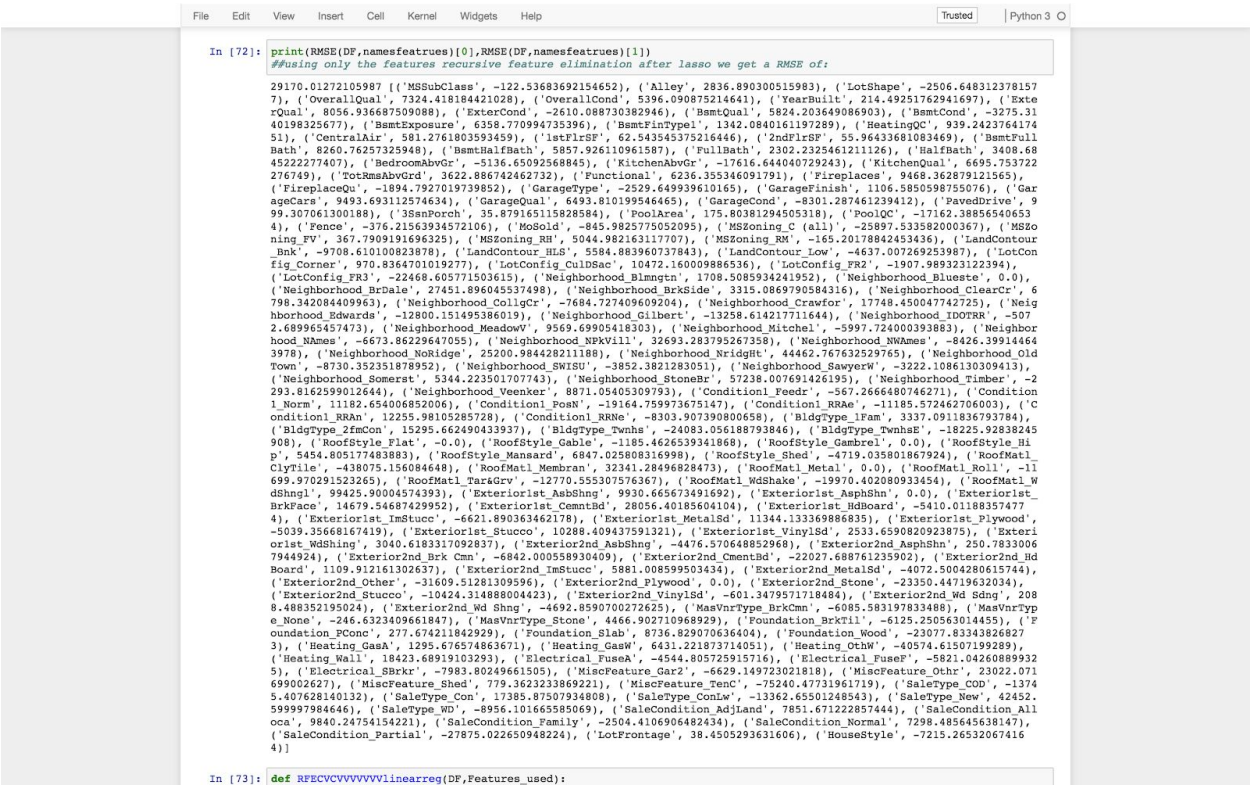
After eliminating the categorical features that did not influence price using the Kruskal-Wallis H Test, we run some models to try to estimate price with the remaining features.

First, we utilized lasso (we checked that the features removed by lasso wear somewhat correlated to the remaining ones, this is because lasso is inclined to reduce the coefficient to 0, so it is important to make sure that features that may not influence the model in a great proportion but are important are not removed). After running the model, we did cross validation to get the square root of the mse, which was 31258.7858$.

The parameters we used for Lasso were:(incrementing to more iterations did not give a better result)
alpha = 2, max_iter = 5000, tol = 1e-20

Then, we utilize Recursive Feature Elimination (or RFECV), which works by recursively removing attributes and building a model on those attributes that remain. It uses the model accuracy to identify which attributes (and combinations of attributes) contribute the most to the predicting the target. We utilized the lasso coefficients from the previous model and ran the RFECV, which gave us our optimal number of features. Then, we tested the selected model with selected with our training data on our test data set and got 29170.01272 as our RMSE.

This shows us the coefficients for the optimal number of features after RFECV and Lasso:



Another approach we utilized with RFECV was linear regression. Our RMSE was 29824.8428.

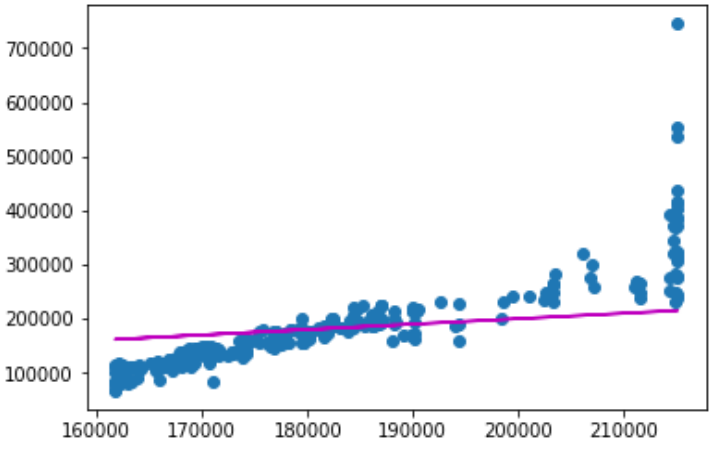
When trying to decide which of these two models to utilize, we took into consideration that the method with lasso and RFECV has some penalty if more coefficients (variables) are in the model. This penalty helps us avoid overfitting the model and it still works well with a different data set.

2. LightGBM

DataSet preparation: We divided the dataset into three parts. Training and testing datasets for model building. A true testing dataset is used for comparing algorithms.
Parameter setting: There are four major parameters in LightGBM: learning_rate, sub_feature, num_leaves, min_data, and max_depth. Our initial setting and the corresponding MSE were:

params['learning_rate'] = 0.003
params['sub_feature'] = 0.5
params['num_leaves'] = 10
params['min_data'] = 50
params['max_depth'] = 10
MSE (before optimization)= 4764732533.46014$

The prediction plot shows the overall prediction performance in the true testing dataset:

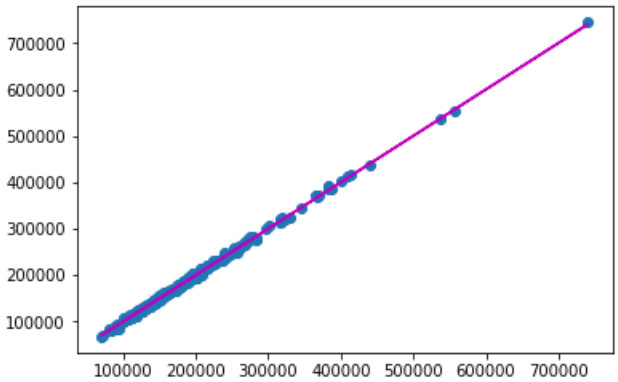


y-axis: True SalePrice ; x-axis: Prediction

After using the above parameters to optimize the coefficients on the train data, we found the values to replace the above parameters with that minimized the MSE on the test data.
in during the modeling process using training and testing datasets.

params['learning_rate'] = 0.121
params['sub_feature'] = 0.3
params['num_leaves'] = 8
params['min_data'] = 2
params['max_depth'] = 5

Then we used these parameters to predict the sale price in the true testing dataset. Our MSE for that LightGBM is 9819810.0.
The prediction plot shows the overall prediction performance in the true testing



dataset:

References :

https://www.kaggle.com/yanpapadakis/houseprices-eda anova
https://www.kaggle.com/sxliuliang9494/xgboost-lasso xgboost

https://www.kaggle.com/pmarcelino/comprehensive-data-exploration-with-python Correlation