Capstone Project Final Report

Predict House Sales Price

Jiyoung Chung

Introduction

There are so many factors affecting the sales price of a house. We are going to find which factors can make an impact when deciding the sales price of a house and the result would be useful for both buyers and sellers - buyers would be able to find a reasonable price for the house when making a deal with the seller, and sellers would also take advantages of it by selling the property at the attractive price without lagging time. We will check this out with the house sales price in Ames, Iowa.

Problem Statement

As either a home buyer or seller, it is difficult to price the property accordingly. If it is too high, many buyers would think it’s overpriced and the seller should wait for a long time or lower the price eventually. Conversely, if it is too low, buyers would be interested in that property but the seller obviously would not be happy with it.

Data Sources

I've got the data from a competition in Kaggle, "House Prices - Advanced Regression Techniques", but the original data came from "The Ames Housing dataset", which was compiled by Dean De Cock for use in data science education. It's an incredible alternative for data scientists looking for a modernized and expanded version of the often cited Boston Housing dataset.

About the data

There are two datasets - train & test dataset in each CSV file, and one feature description in txt file.

- train.csv - the training set
- test.csv - the test set
- data_description.txt - full description of each column, originally prepared by Dean De Cock but lightly edited to match the column names used here

Data Wrangling

The data has 1,460 records with 79 columns in total. Below is an overview on various types of data wrangling I've dealt with.

A. Removing null data

   I ordered the columns by the number of null values in each and decided to remove columns that have more than 80% of missing values as these columns will not work effectively on the house price.

   → ['PoolQC', 'MiscFeature', 'Alley', 'Fence'] have been dropped.

B. Replacing values by one-hot encoding

Checking the missing values in ‘FireplaceQu’ column, the missing values were not actually null, instead they meant ‘No fireplace’, so I filled the null with an integer 0 and replaced the grades - ‘Ex’, ‘Gd’, ‘TA’, ‘Fa’, and ‘Po’ to numbers - 5,4,3,2, and 1. e.g. Python code

```
df['FireplaceQu'].replace('Ex', 5, inplace=True)
df['FireplaceQu'].replace('Gd', 4, inplace=True)
df['FireplaceQu'].replace('TA', 3, inplace=True)
df['FireplaceQu'].replace('Fa', 2, inplace=True)
df['FireplaceQu'].replace('Po', 1, inplace=True)
df['FireplaceQu'].fillna(0, inplace=True)
```

Note. This method has been also applied to many other columns having the same situation, like garage or basement related columns.

C. Filling null values out from another column

In the case of ‘GarageYrBlt’, we can assume that the garage was built in the same year that the house was built with high chances, so I filled the null values in the column with the year of house built.

D. Filling null values out with the statistical data of the column

For columns having a very small number of null values, like ‘Electrical’, there was only one null value and fortunately, ‘SBrkr’ was the majority value of the column, so I replaced the null value with ‘SBrkr’.

E. Removing columns having one dominant value

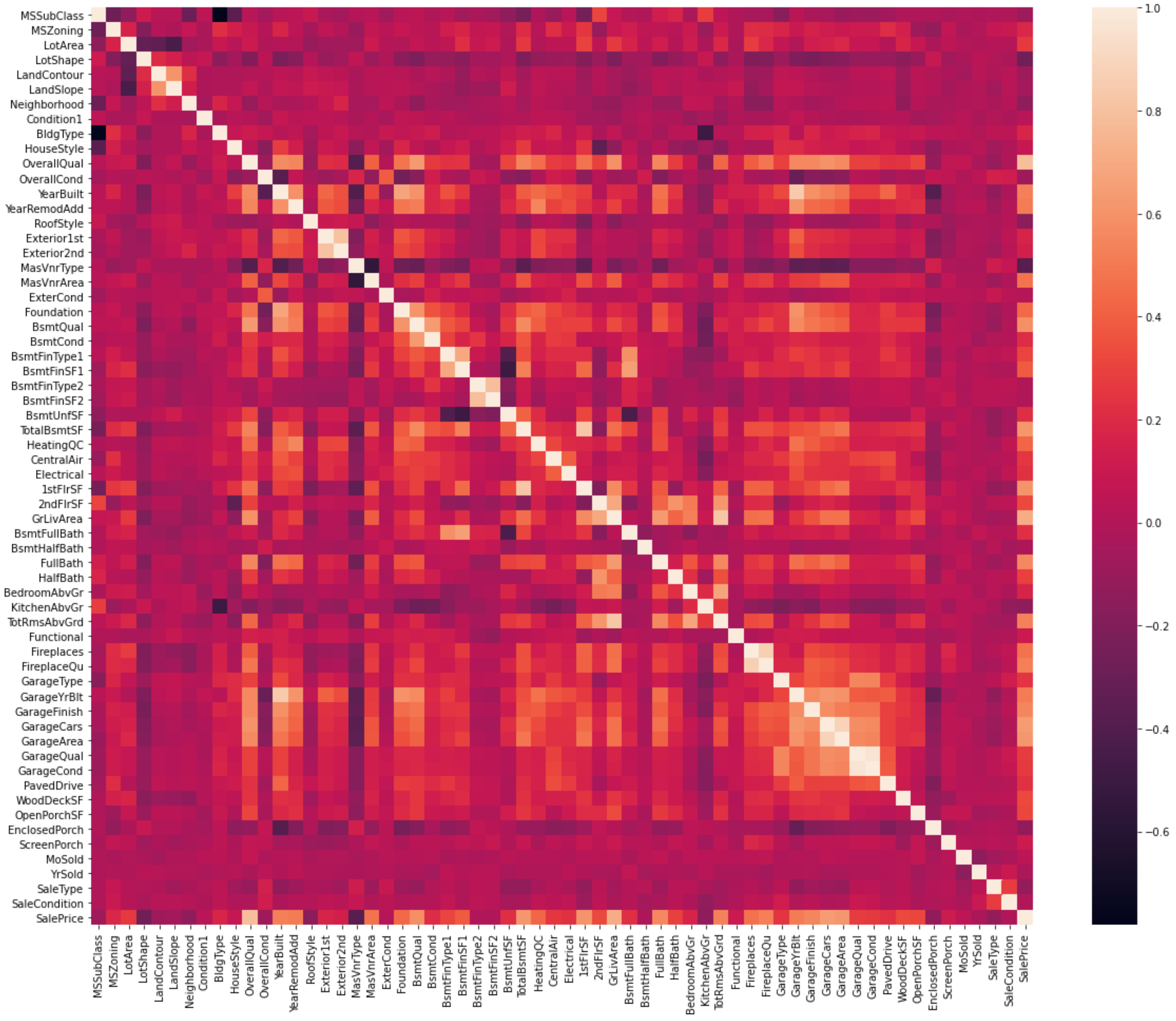
If one value is dominant over a whole column, I dropped the column as these columns will also not work effectively on the house price.

→ [‘Street’, ‘Utilities’, ‘Condition2’, ’RoomMatl’, ‘Heating’, ‘LowQualFinSF’, ‘3SsnPorch’, ‘PoolArea’, ‘MiscVal’] have been dropped.

Exploratory Data Analysis

Heatmap

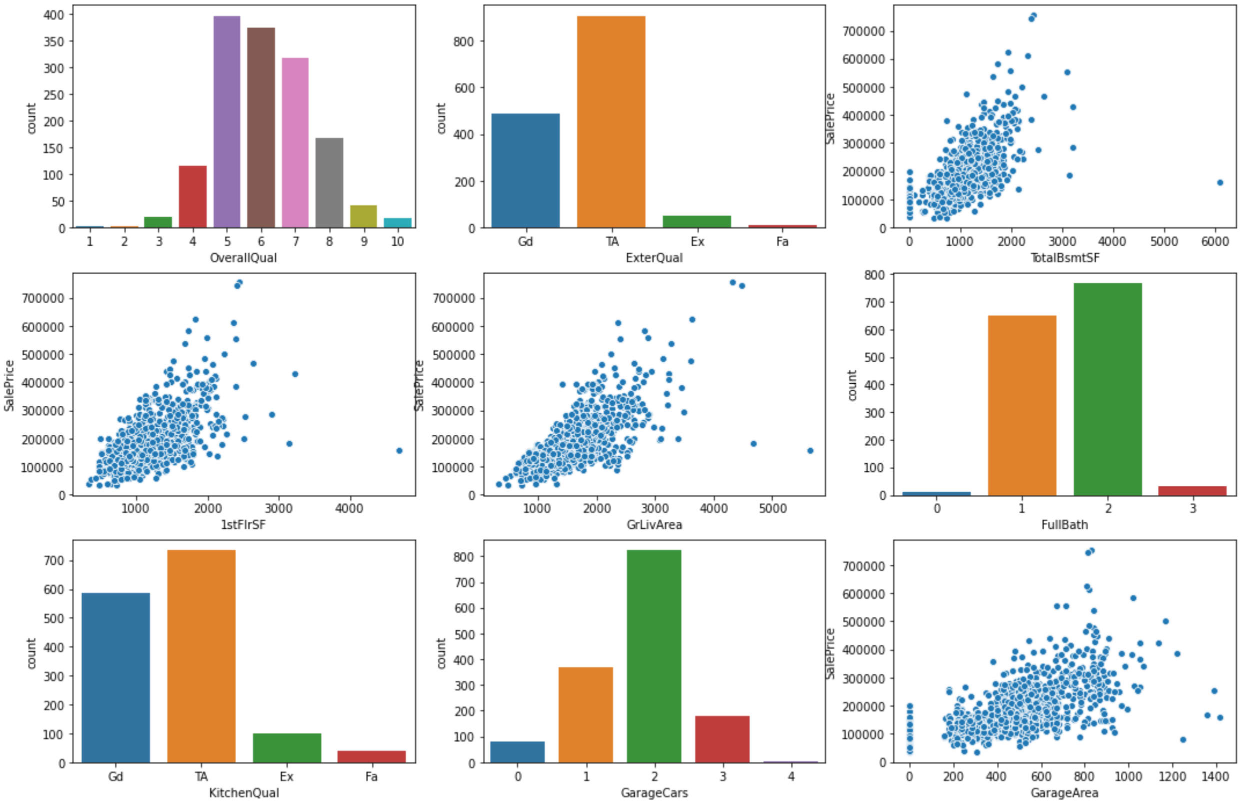
Before drawing a heatmap, I converted non-numeric columns to numeric columns so that they can be expressed in the heatmap as well. For this converting, I used one-hot Encoding.



As shown in the heatmap, I could pick below 9 columns to be a feature since they were correlated to our target, ‘SalePrice’.

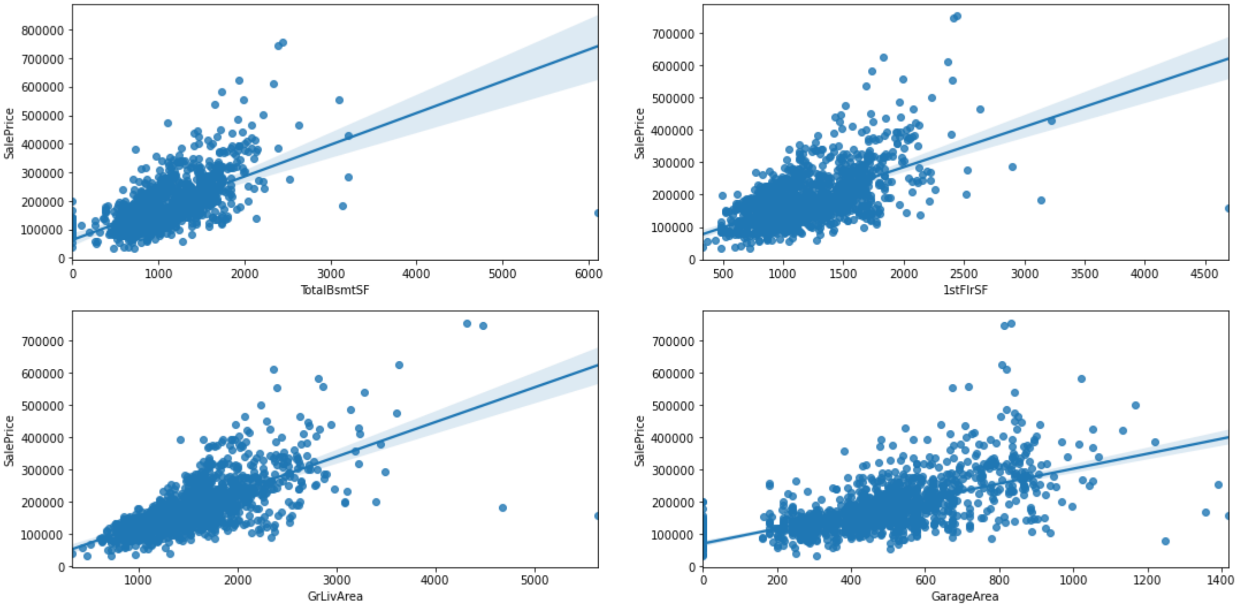
Discrete columns	'OverallQual', 'ExterQual', 'FullBath', 'KitchenQual', 'GarageCars'
Continuous columns	'TotalBsmtSF', '1stFlrSF', 'GrLivArea', 'GarageArea'

Countplot & Scatterplot



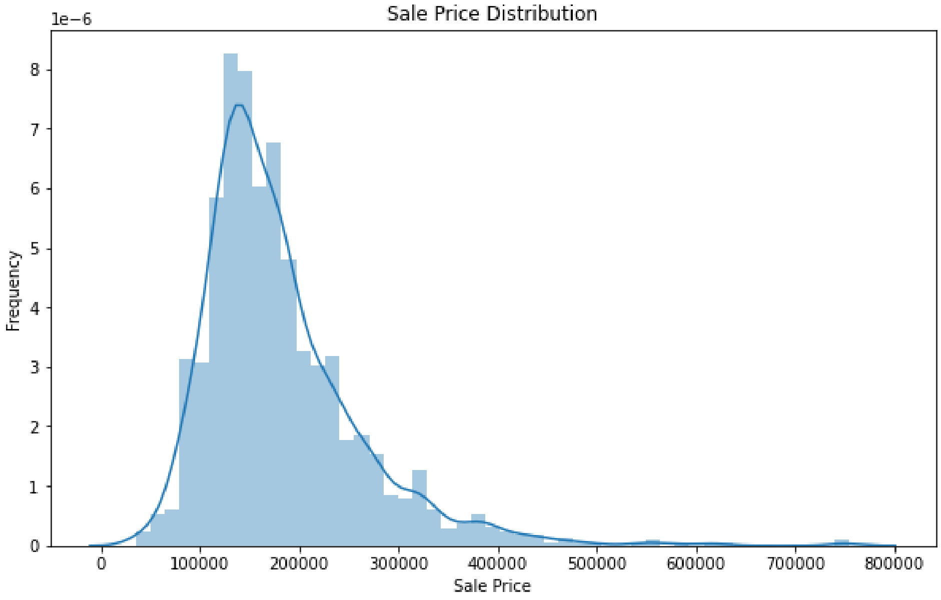
I plotted discrete columns using countplot, and continuous columns using scatterplot with Seaborn as above.

Regression Plot



Using regression plots on the continuous columns as above, we can see that they are positively correlated with the 95% confidence interval band to 'SalePrice'.

Distribution Plot



We can see the sale price is well distributed in the dataset.

Modeling

I've splitted the train file into train and test, and built and trained six different types of regression models which are Linear regression, Logistic regression, Polynomial regression, Ridge regression, Lasso regression, and Gradient Boosting regression. For all models, I've fitted each model with train data and got each model score for comparison.

Model	Score
Linear Regression	0.78356
Logistic Regression	0.08048
Polynomial Regression	0.86583
Ridge Regression	0.79112
Lasso Regression	0.99649
Gradient Boosting Regression	0.99649

It looks like both Lasso regression and Gradient Boosting regression can be good to use as they tie the highest score, 0.99649, which looks great. In the meantime, Logistic regression scored very poorly, so this regression shouldn't be used for our data to solve the case.

Model Evaluation with R-squared

However, when evaluating the models with R-squared, Gradient Boosting regression showed the best score and became the sole regression that got the highest score in both modeling and evaluating.

Model	Score
Linear Regression	0.79075
Logistic Regression	0.57718
Polynomial Regression	0.84204
Ridge Regression	0.79112
Lasso Regression	0.79079
Gradient Boosting Regression	0.84871

Logistic regression got the evaluation score below 0.60, which should be between 0.60 and 1.0, so it has been confirmed not to be used for our data.

Prediction

Using Gradient Boosting regression, I predicted the test data and got the score of 0.99474.

Conclusion

- From our data, we could see that 9 out of 79 attributes mainly affect the house sales price.
- In spite of some points to improve on the data and analysis, it is expected to be much helpful for both the sellers and the buyers to decide the proper house price if they have these types of information.
- As well, it is expected to allow the agent and brokers to provide precise market information to their customers.

Future Improvements

- There are still many factors which were not in the data and can affect the house price, such as school scores, safety of the area, etc. If they could be added to the analysis, we could predict more precisely.
- More metrics can be used to evaluate models in various ways, such as MAE (Mean Absolute Error), MSE (Mean Squared Error), or RMSE (Root Mean Squared Error).