Capstone Project Report

House Price Prediction Problem Statement:

Predict the final price of each home according to the market prices taking into account different features of the house.

Description:

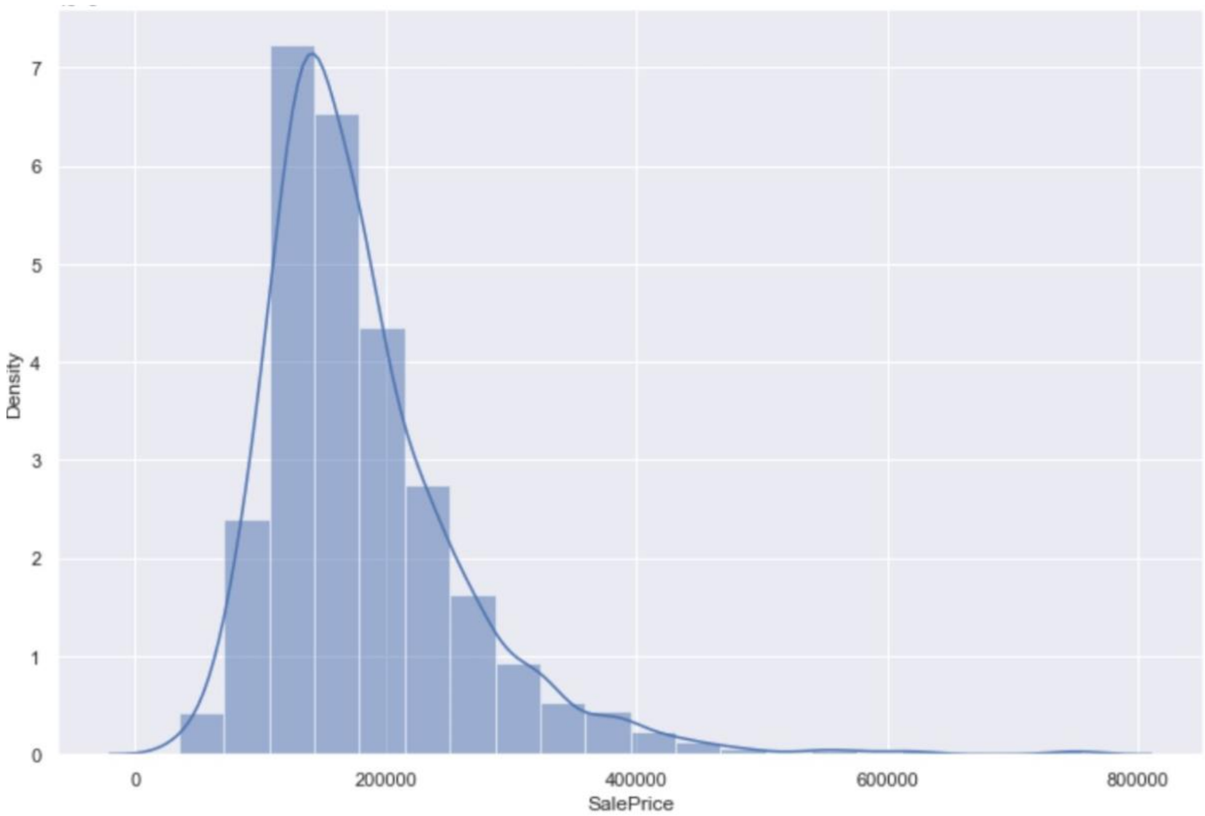
Buyers are generally not aware of factors that influence the house prices. Real estate agents are trusted with the communication between buyers and sellers as well as laying down a legal contract for the transfer. This just creates a middle man and increases the cost of houses tremendously. The common features people believe it depends on are the houses neighbourhood, square footage and number of bedrooms. But it depends upon many factors such as number of floors, areas outside the house and the number of rooms on different floors. It is important to predict housing prices without bias to help both the buyers and sellers make their decisions and a data science technique is required to do the job of a middle man hence the buyers can save their unnecessary expenditures.

Data Wrangling:

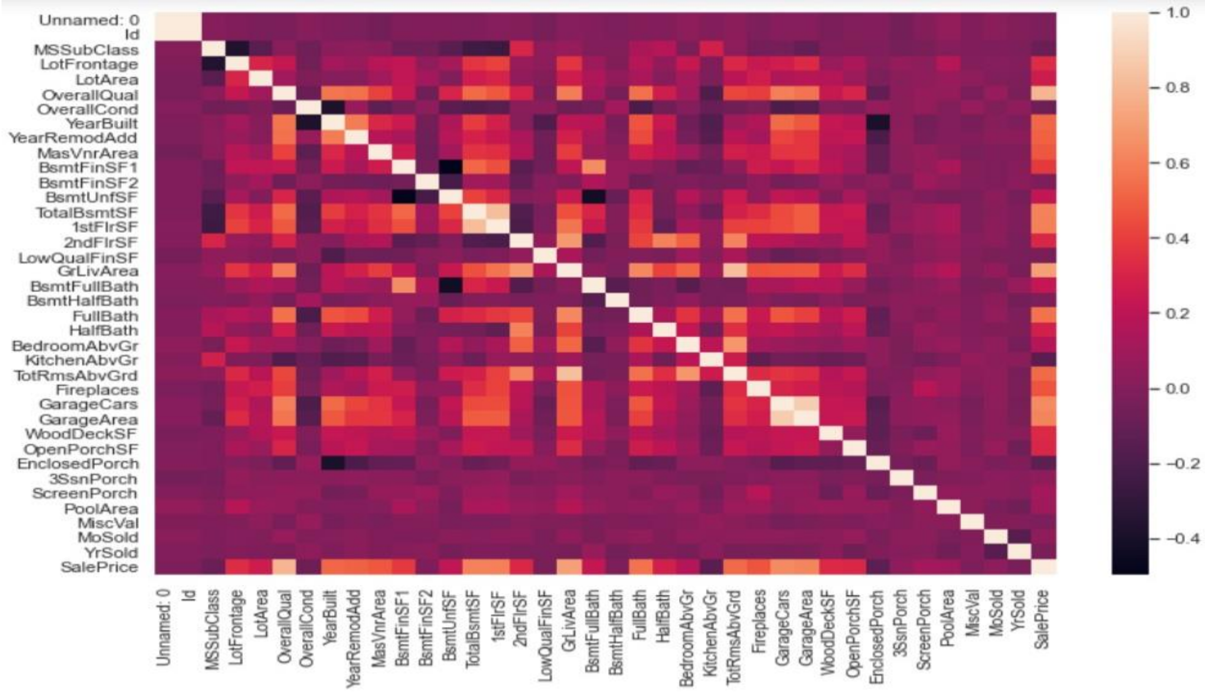
The raw dataset from Ames Housing dataset contained 79 explanatory variables (features) describing every aspect of residential homes in Ames, Iowa and there are 1460 observations and SalePrice feature is the target variable. The dataset contained int, float and object types of data as features along with some null values in between. There are some feature columns which had null values more than 80 per cent so I dropped those columns since they have least effect on the result. Outlier data points were looked at individually to determine whether the number was an incorrect entry or legitimate. The former were either corrected or made null while the latter were kept in the dataset. Null values were filled based on the data type of the particular feature. Null values in categorical feature were filled with the most frequently occurring category and integer as well as float features were filled with mean value of the particular feature column. Hence the final shape of the dataset was 1460 rows and 76 columns.

Exploratory Data Analysis:

Since the dataset contains the many numerical features then it’s better to know the distribution of those numerical values with the target variable. The following histogram shows the distribution:



The above histogram is right skewed. This indicates there are some outliers.



The above heatmap shows the high positive correlation between OverallQual, GrLivArea, GarageCars, GarageArea, TotalBsmtSF, 1stFlrSF features and target feature SalePrice.

Initially it was I thought the year sold was not affecting the target variable but the following plot shows its negative correlation on the target variable.

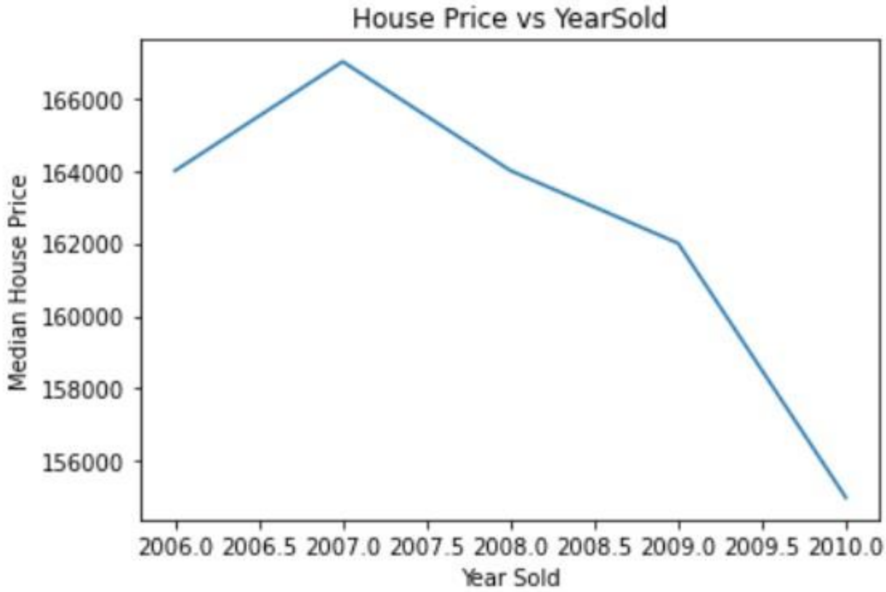


Figure 5

Feature Engineering:

In this step I filtered the features and selected only those features which have either positive or negative impact on target feature SalePrice.

The features such as 'LotFrontage', 'LotArea', '1stFlrSF', 'GrLivArea', 'SalePrice' have skewed values (outliers) therefore converting them to log normal distribution. Similarly YearBuilt and YearRemodAdd features are big number therefore converting them to small number by subtracting them from the YearSold feature. The features which are correlated either positively or negatively on the target feature with the same amount (same value) then I selected any one among them. I dropped two more feature columns and was left with 73 columns. Next step is to convert the categorical features into dummy features. I did this with the help of dummies() function. It creates a separate column for each category in the feature with numerical values. The training set contains a known output and the model learns on this data in order to be generalized to other data later on. I have the test dataset (or subset) in order to test our model’s prediction on this subset. Except the target variable (SalePrice) consider the dataset as one variable called X and target variable as y. I did this using the Scikit-Learn library and specifically the train_test_split method.

Modelling:

Model 1 (Linear regression):

R^2 = -18584463715.234547
RMSE = 3167375859.3228564

Model 2 (Lasso Regression):

R^2 is: 0.9029961365303035
RMSE is: 0.016532502638027132

Conclusion:

I used machine learning algorithms to predict the house prices. I have mentioned the step by step procedure to analyse the dataset and finding the correlation between the parameters. Thus I can select the parameters which are not correlated to each other and are independent in nature. These feature set were then given as an input to three algorithms and observed their performance. I calculated the performance of each model using RMSE metric and compared them and the lasso regression came out with high percentage of success. For future projects, I recommend that working on large dataset would yield a better predicting model.