HOUSING: PRICE PREDICTION

Submitted by:

VICKY

ACKNOWLEDGMENT

Reference	:FlipRobo Technologies
Research Paper	:From FlipRobo Technologies
Data Sources	:Database of FlipRobo Technologies
Professionals Help	: Ms. Sapna Verma(SME)

INTRODUCTION

Houses are one of the necessary need of each and every person around the globe and therefore housing and real estate market is one of the markets which is one of the major contributors in the world's economy. It is a very large market and there are various companies working in the domain.

- ## Problem Statement

Data science comes as a very important tool to solve problems in the domain to help the companies increase their overall revenue, profits, improving their marketing strategies and focusing on changing trends in house sales and purchases. Predictive modelling, Market mix modelling, recommendation systems are some of the machine learning techniques used for achieving the business goals for housing companies. Our problem is related to one such housing company.

A US-based housing company named Surprise Housing has decided to enter the Australian market. The company uses data analytics to purchase houses at a price below their actual values and flip them at a higher price. For the same purpose, the company has collected a data set from the sale of houses in Australia.

- **Conceptual Background of the Domain Problem**

  It thus becomes obvious that domain knowledge is important both in the framework as well as the body of a data science project. It will make the project faster, cheaper and more likely to yield a useful answer.

  Some concepts are as under that help to understand project and also get better result:

  - ❖ Exploratory Data Analysis.
  - ❖ Precision
  - ❖ Accuracy
  - ❖ Representativeness: Does the dataset reflect all relevant aspects of the domain?
  - ❖ Significance: Does the dataset reflect every important behavior/dynamic in the domain?

- **Motivation for the Problem Undertaken**

  As we know , the Housing Price Prediction case is a real world problem and all data which is I have, is totally based on real data so the main thing is that, with the help of real data we will predict future outcomes and that is beneficial for our business. The real time problem is the only thing that motivate me lot.

Data Preprocessing/Assumptions

Apply Exploratory Data Analysis (EDA):

- ✓ Shape of Train Data: 1168 Rows and 81 Columns.
- ✓ Shape of Test Data: 292 Rows and 80 Columns
- ✓ Data Types: Some features are Numerical in Nature and some are object data types.
- ✓ Find data information.
- ✓ Apply Unique Values approach to find all features unique values.
- ✓ Find Null Values: Null Values present in both datasets.
- ✓ Data Cleaning: Treatment of Null Values and dropping not highly related features.

Data Sources and their formats

The data source is FlipRobo Technologies clients database and the other details about data are as under:

1. Data information is provide below:

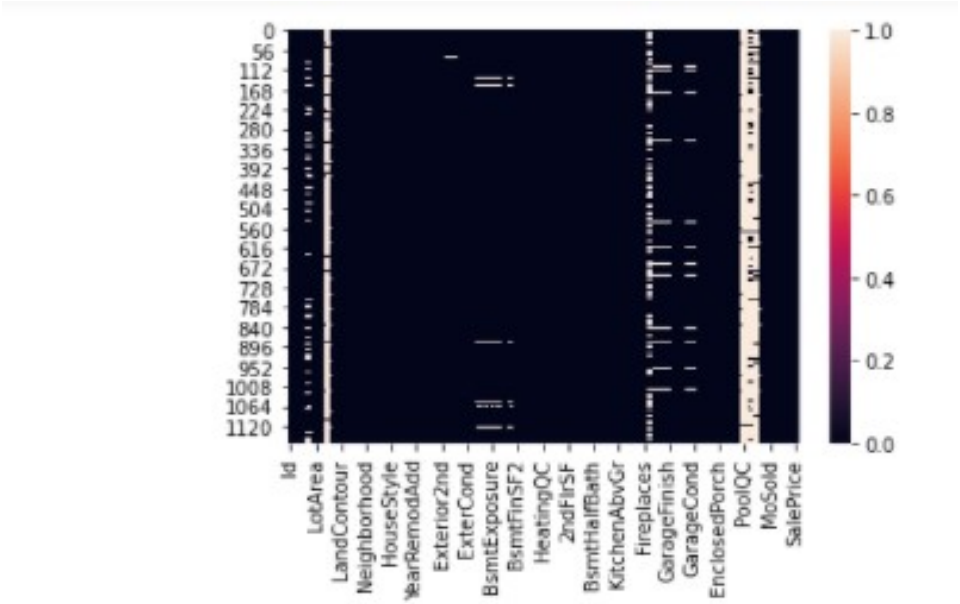
```
In [9]: # Checking information about both data.
        train.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1168 entries, 0 to 1167
Data columns (total 81 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Id             1168 non-null   int64
 1   MSSubClass     1168 non-null   int64
 2   MSZoning       1168 non-null   object
 3   LotFrontage    954 non-null    float64
 4   LotArea        1168 non-null   int64
 5   Street         1168 non-null   object
 6   Alley          77 non-null     object
 7   LotShape       1168 non-null   object
 8   LandContour    1168 non-null   object
 9   Utilities      1168 non-null   object
 10  LotConfig      1168 non-null   object
 11  LandSlope      1168 non-null   object
 12  Neighborhood   1168 non-null   object
 13  Condition1     1168 non-null   object
 14  Condition2     1168 non-null   object
 15  BldgType       1168 non-null   object
 16  HouseStyle     1168 non-null   object
 17  OverallQual    1168 non-null   int64
 18  OverallCond    1168 non-null   int64
 19  YearBuilt      1168 non-null   int64
 20  YearRemodAdd   1168 non-null   int64
 21  RoofStyle      1168 non-null   object
 22  RoofMatl       1168 non-null   object
 23  Exterior1st    1168 non-null   object
 24  Exterior2nd    1168 non-null   object
 25  MasVnrType     1161 non-null   object
 26  MasVnrArea     1161 non-null   float64
 27  ExterQual      1168 non-null   object
 28  ExterCond      1168 non-null   object
 29  Foundation     1168 non-null   object
 30  BsmtQual       1138 non-null   object
 31  BsmtCond       1138 non-null   object
 32  BsmtExposure   1137 non-null   object
 33  BsmtFinType1   1138 non-null   object
 34  BsmtFinSF1     1168 non-null   int64
 35  BsmtFinType2   1137 non-null   object
 32  BsmtExposure   1137 non-null   object
 33  BsmtFinType1   1138 non-null   object
 34  BsmtFinSF1     1168 non-null   int64
 35  BsmtFinType2   1137 non-null   object
 36  BsmtFinSF2     1168 non-null   int64
 37  BsmtUnfSF      1168 non-null   int64
 38  TotalBsmtSF    1168 non-null   int64
 39  Heating        1168 non-null   object
 40  HeatingQC      1168 non-null   object
 41  CentralAir     1168 non-null   object
 42  Electrical     1168 non-null   object
 43  1stFlrSF       1168 non-null   int64
 44  2ndFlrSF       1168 non-null   int64
 45  LowQualFinSF   1168 non-null   int64
 46  GrLivArea      1168 non-null   int64
 47  BsmtFullBath   1168 non-null   int64
 48  BsmtHalfBath   1168 non-null   int64
 49  FullBath       1168 non-null   int64
 50  HalfBath       1168 non-null   int64
 51  BedroomAbvGr   1168 non-null   int64
 52  KitchenAbvGr   1168 non-null   int64
 53  KitchenQual    1168 non-null   object
 54  TotRmsAbvGrd   1168 non-null   int64
 55  Functional     1168 non-null   object
 56  Fireplaces     1168 non-null   int64
 57  FireplaceQu    617 non-null    object
 58  GarageType     1104 non-null   object
 59  GarageYrBlt    1104 non-null   float64
 60  GarageFinish   1104 non-null   object
 61  GarageCars     1168 non-null   int64
 62  GarageArea     1168 non-null   int64
 63  GarageQual     1104 non-null   object
 64  GarageCond     1104 non-null   object
 65  PavedDrive     1168 non-null   object
 66  WoodDeckSF     1168 non-null   int64
 67  OpenPorchSF    1168 non-null   int64
 68  EnclosedPorch  1168 non-null   int64
 69  3SsnPorch      1168 non-null   int64
 70  ScreenPorch    1168 non-null   int64
 71  PoolArea       1168 non-null   int64
 72  PoolQC         7 non-null      object
 73  Fence          237 non-null    object
 74  MiscFeature    44 non-null     object
 75  MiscVal        1168 non-null   int64
 76  MoSold         1168 non-null   int64
 77  YrSold         1168 non-null   int64
 78  SaleType       1168 non-null   object
 79  SaleCondition  1168 non-null   object
 80  SalePrice      1168 non-null   int64
dtypes: float64(3), int64(35), object(43)
memory usage: 739.2+ KB
```

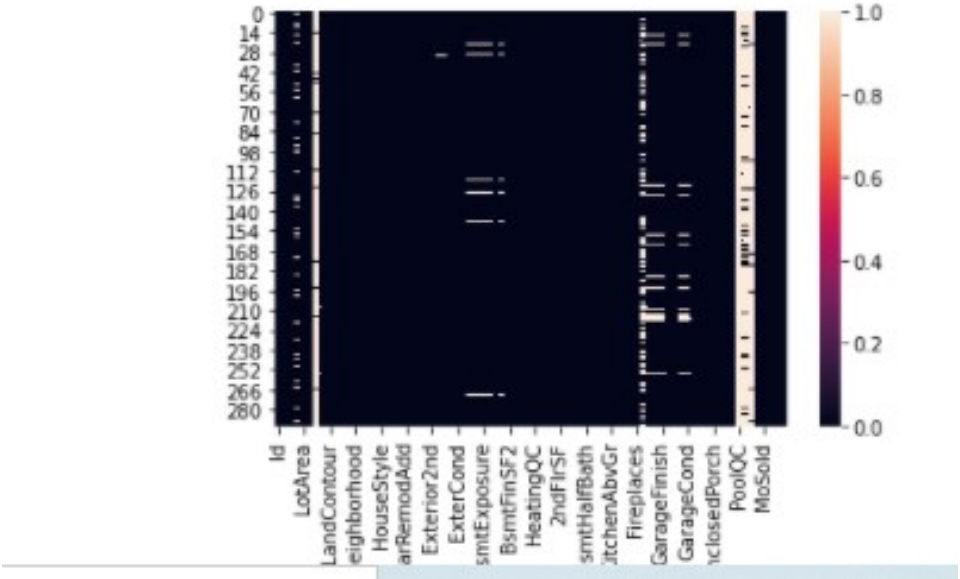
Visualization

- ✓ Checking Null values by Heatmap.
- ✓ Analyzing all Features by Univariate Analysis.
- ✓ Correlation.
- ✓ Statistical Summary.

1. Missing Values are present in our data.




```
In [16]: sns.heatmap(test.isnull())
Out[16]: <AxesSubplot:>
```

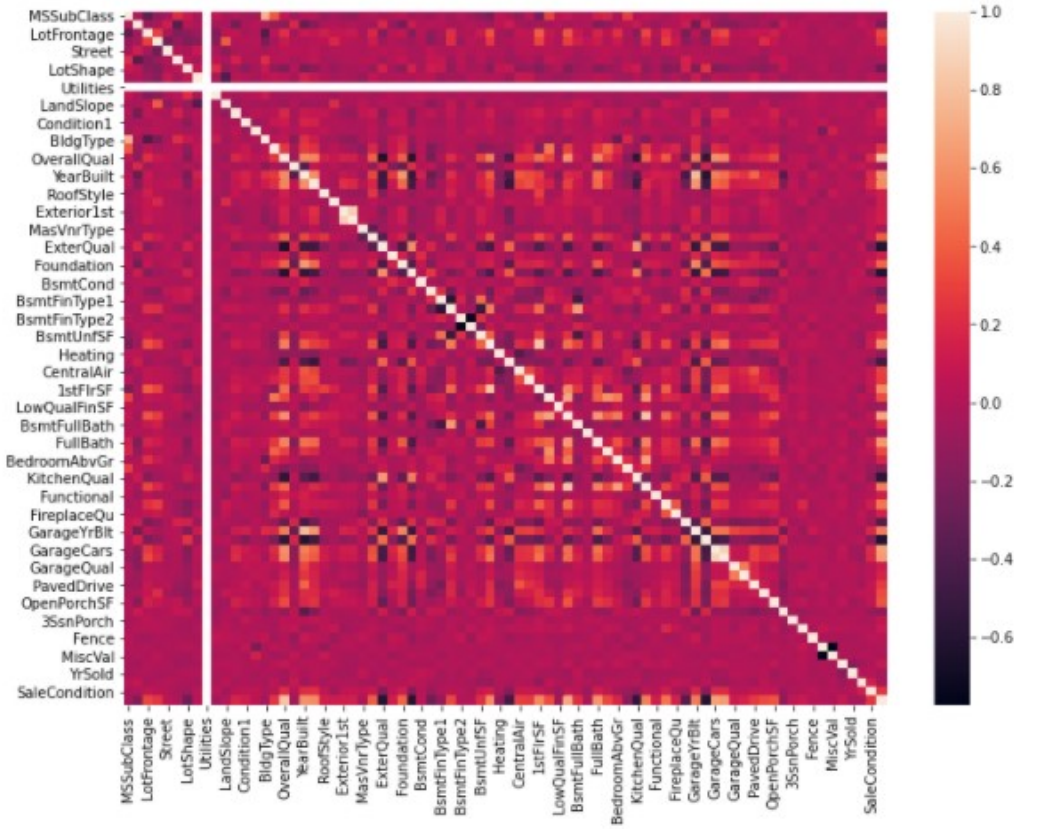


2. We drop column “ID, PoolArea and PoolQC” because these are not holding much information.

- **Data Inputs- Logic- Output Relationships**

  Some columns have good relationship as we are getting by correlation diagram.

```
In [71]: fig, ax = plt.subplots(figsize=(12,9))
         sns.heatmap(train.corr(), ax=ax);
```



We can see that there are many correlated variables in our dataset. Wwe notice that Garage Cars and Garage Area have high positive correlation which is reasonable because when the garage area increases, its car capacity increases too. We see also that Gr Liv Area and TotRms AbvGrd are highly positively correlated which also makes sense because when living area above ground increases, it is expected for the rooms above ground to increase too.

Regarding negative correlation, we can see that Bsmt Unf SF is negatively correlated with BsmtFin SF 1, and that makes sense because when we have more unfinished area, this means that we have less finished area. We note also that Bsmt Unf SF is negatively correlated with Bsmt Full Bath which is reasonable too.

Most importantly, we want to look at the predictor variables that are correlated with the target variable (SalePrice). By looking at the last row of the heatmap, we see that the target variable is highly positively correlated with Overall Qual and Gr Liv Area. We see also that the target variable is positively correlated with Year Built, Year Remod/Add, Mas Vnr Area, Total Bsmt SF, 1st Flr SF, Full Bath, Garage Cars, and Garage Area.

Relationships Between the Target Variable and Other Variables

High Positive Correlation

Firstly, we want to visualize the relationships between the target variable and the variables that are highly and positively correlated with it, according to what we saw in the heatmap. Namely, these variables are Overall Qual and Gr Liv Area. We start with the relationship between the target variable and Overall Qual, but before that, let's see the distribution of each of them. Let's start with the target variable SalePrice:

Statistical Approaches

- ✓ Statistical Summary: Summary provides information about statistical parameters of data i.e. Count, Mean, Standard Deviation, Minimum value, 25%, 50% 75% and Maximum Value. It helps to check data structure and try to understand chronology of data.

Statistical Summary

```
In [72]: train.describe()
```

Out[72]:

	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	LotConfig	...	3SsnPorch	Scre
count	1168.000000	1168.000000	1168.000000	1168.000000	1168.000000	1168.000000	1168.000000	1168.000000	1168.0	1168.000000	...	1168.000000	116
mean	56.767979	3.013699	70.988470	10484.749144	0.996575	0.030822	1.938356	2.773973	0.0	3.004281	...	3.639555	1
std	41.940650	0.633120	22.437056	8957.442311	0.058445	0.172909	1.412262	0.710027	0.0	1.642667	...	29.088867	5
min	20.000000	0.000000	21.000000	1300.000000	0.000000	0.000000	0.000000	0.000000	0.0	0.000000	...	0.000000	
25%	20.000000	3.000000	60.000000	7621.500000	1.000000	0.000000	0.000000	3.000000	0.0	2.000000	...	0.000000	
50%	50.000000	3.000000	70.988470	9522.500000	1.000000	0.000000	3.000000	3.000000	0.0	4.000000	...	0.000000	
75%	70.000000	3.000000	79.250000	11515.500000	1.000000	0.000000	3.000000	3.000000	0.0	4.000000	...	0.000000	
max	190.000000	4.000000	313.000000	164660.000000	1.000000	1.000000	3.000000	3.000000	0.0	4.000000	...	508.000000	48

8 rows × 78 columns

The above Summary provide Statistical information about whole data i.e. Count, Mean, Standard Deviation, Minimum value, 25%, 50% 75% and Maximum Value and statistical report is mentioned above

Encoding

- ✓ In our datasets some features are categorical and some are numerical in nature. We are encoding all features into numerical form to analysis the data statistically.
- ✓ We are using Label Encoder for encoding to encode all features.

Splitting All data into Independent variable and Target variable

- ✓ Here we are separating target and independent variable for model deployment. As we know, We applied all approaches on data i.e. Data Cleaning, Data Encoding, Data Scaling, Feature Engineering know we are applying all machine learning algorithms for prediction.

Model/s Development and Evaluation

- **Run and Evaluate selected models**

Liner Regression Algorithm:

- ✓ Liner Regression Algorithm performing Excellent and Its accuracy score is 92 % approx.

Random Forest Regressor:

- ✓ Random Forest Regressor Algorithm performing Excellent and Its accuracy score is 92 % approx.

Decision Tree Regressor:

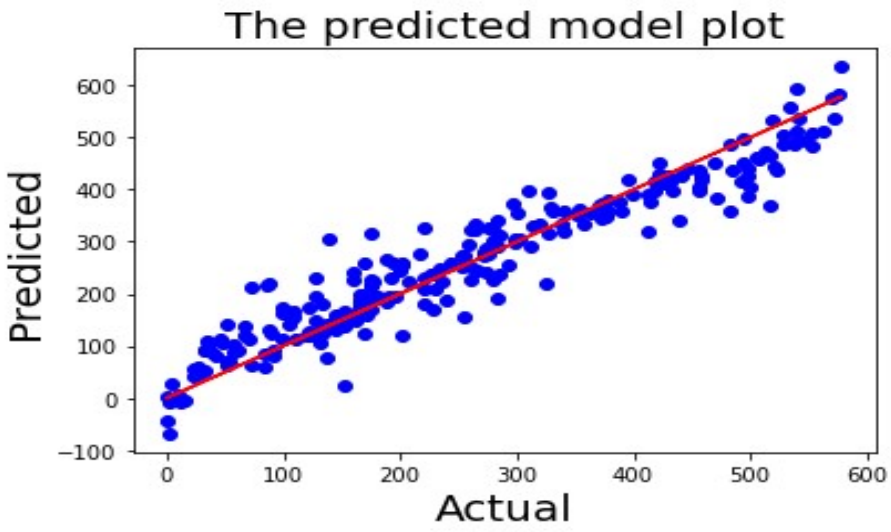
- ✓ Decision Tree Regressor Algorithm performing Very good and Its accuracy score is 80 % approx but is less than above algorithms.

AdaBoost Regressor:

- ✓ AdaBoost Regressor Algorithm performing Excellent and Its accuracy score is 92 % approx.

- **Model Evaluation:**

```
In [103]: plt.scatter(x=y_test, y=lassopred, color = "blue")
          plt.plot(y_test,y_test, color='r')
          plt.xlabel("Actual", fontsize =20)
          plt.ylabel("Predicted", fontsize=20)
          plt.title("The predicted model plot", fontsize=20)
          plt.show()
```



CONCLUSION

In this paper, we built several regression models to predict the price of some house given some of the house features. We evaluated and compared each model to determine the one with highest performance. We also looked at how some models rank the features according to their importance. In this paper, we followed the data science process starting with getting the data, then cleaning and preprocessing the data, followed by exploring the data and building models, then evaluating the results and communicating them with visualizations.