FLIP ROBO

HOUSING PRICE PREDICTION

Submitted by:

Aditi Sharma

ACKNOWLEDGMENT

I have taken the help of many sites for conceptual knowledge as well as for coding purpose. The sites include analyticsvidhya, geeksforgeeks,medium.com,Kaggle.com.

INTRODUCTION

- **Business Problem Framing**
  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. The company is looking at prospective properties to buy houses to enter the market. You are required to build a model using Machine Learning in order to predict the actual value of the prospective properties and decide whether to invest in them or not. For this company wants to know:
  - Which variables are important to predict the price of variable?
  - How do these variables describe the price of the house?

- **Conceptual Background of the Domain Problem**
  Domain related concepts that will be useful for the project are data science, linear regression, treating missing values, encoding of data.

Analytical Problem Framing

- Data Sources and their formats

```
df.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
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
```

- **Data Pre-processing Done**

  The feature having cardinality and only 1 value all=over were removed.

  The features which have missing values more than 60% were also removed.

  The features which were less important and showed not much correlation were also removed.

  The feature which had many missing values but was important and showed positive correlation was kept.

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

  This is shown through correlation between features and sale price.

- **Hardware and Software Requirements and Tools Used**

  Hardware used: Laptop

  Software used: Anaconda Navigator(Jupiter Notebook)

Libraries used: Pandas, numpy,Sklearn, seaborn, matplotlib.

Model/s Development and Evaluation

- Identification of possible problem-solving approaches (methods)

  The methods I used for solving problem are:

  Decision Tree Regressor

  KNeighbors Regressor

  AdaBoost Regressor

  Linear Regression

  Gradient Boosting Regressor

- Testing of Identified Approaches (Algorithms)

  Listing down all the algorithms used for the training and testing.

- Run and Evaluate selected models

```
model=[DecisionTreeRegressor(),KNeighborsRegressor(),AdaBoostRegressor(),LinearRegression(),GradientBoostingRegressor()]
max_r2_score=0
for r_state in range(40,90):
    train_x,test_x,train_y,test_y=train_test_split(x,y,random_state=r_state,test_size=0.33)
    for i in model:
        i.fit(train_x,train_y)
        pre=i.predict(test_x)
        r2_sc=r2_score(test_y,pre)
        print("R2 score correspond to random state",r_state,"is",r2_sc)
        if r2_sc>max_r2_score:
            max_r2_score=r2_sc
            final_state=r_state
            final_model=i

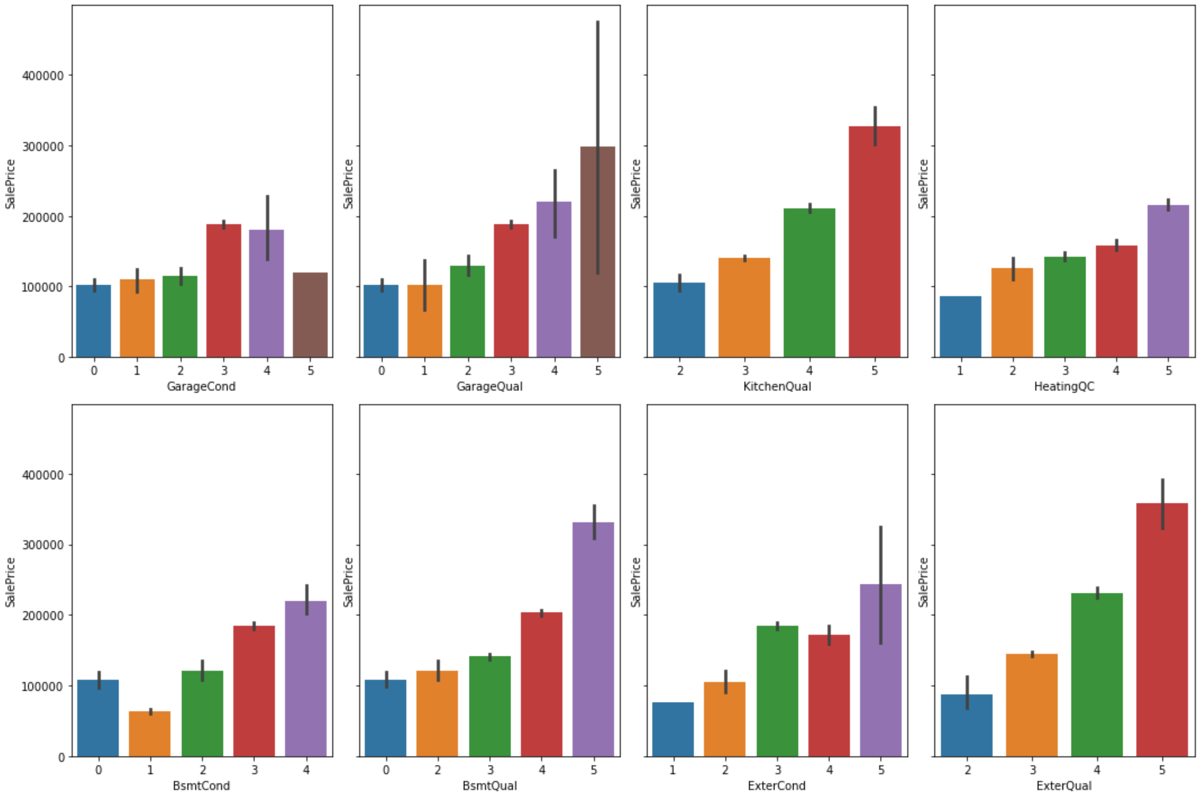
print()
print()
print()
print()
print("max R2 score correspond to random state",final_state,"is",max_r2_score,"and model is",final_model)
```

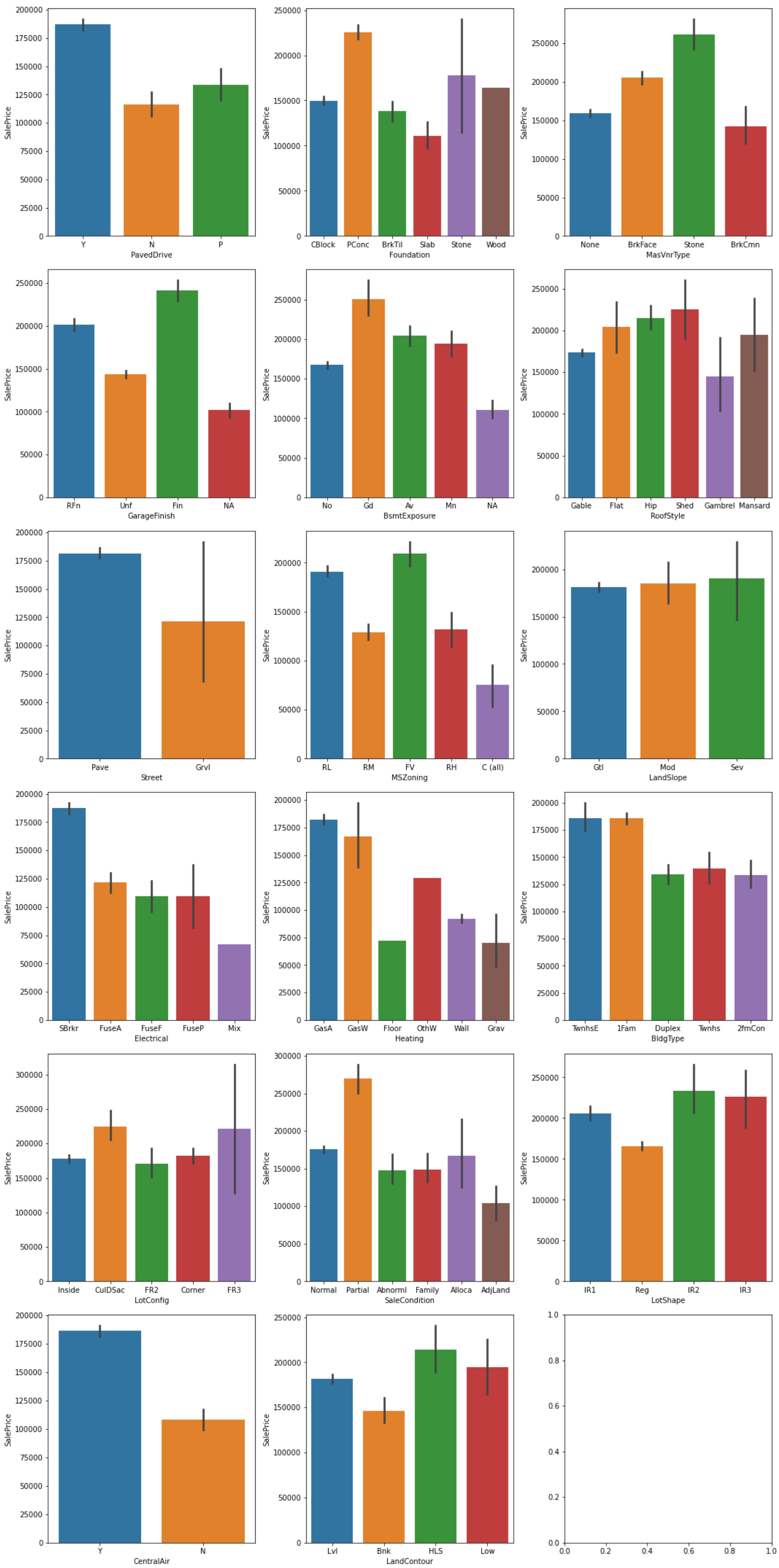
- Key Metrics for success in solving problem under consideration

In [42]:

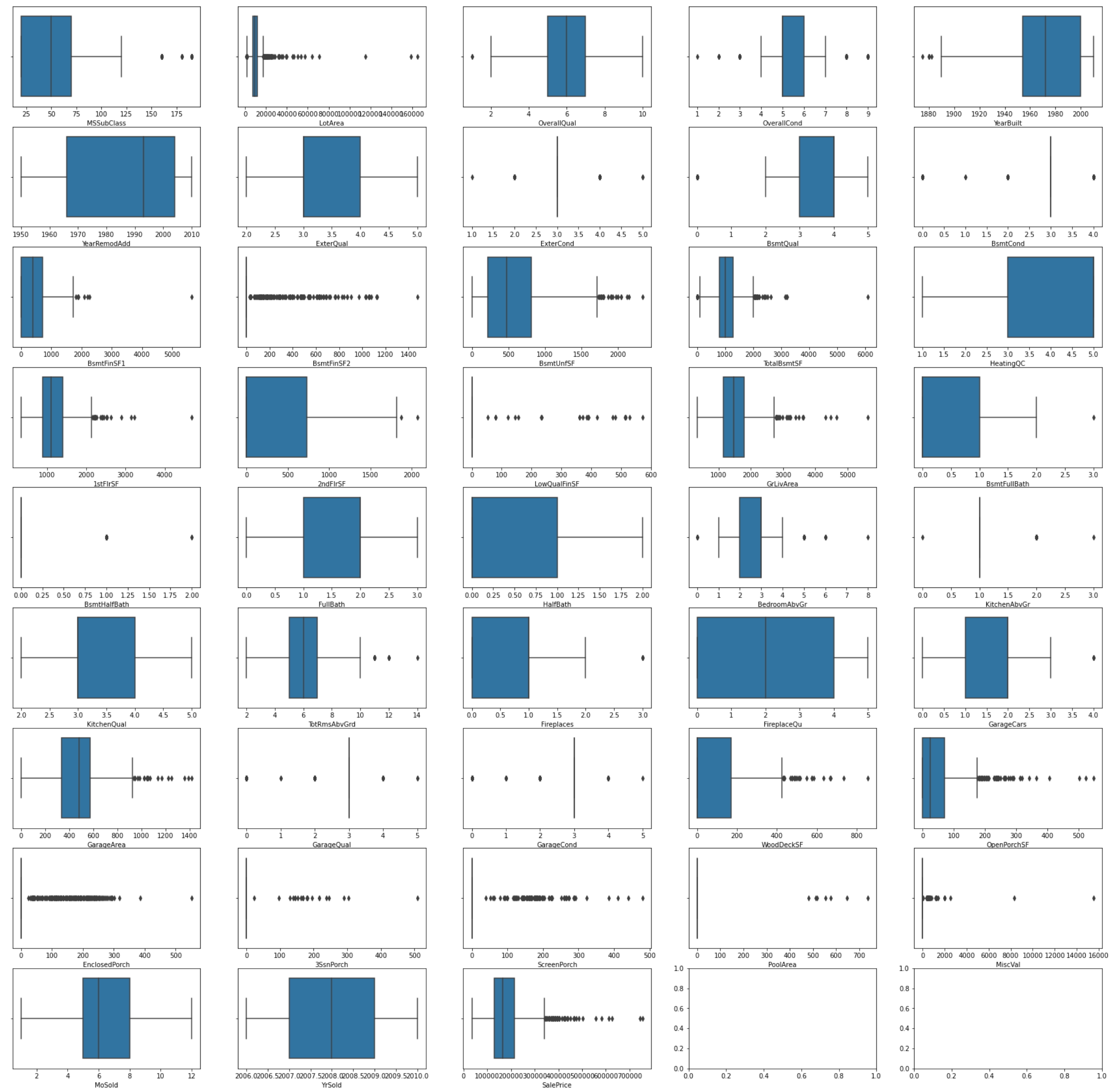
```
model = AdaBoostRegressor()
model= model.fit(train_x,train_y)
pred_y= model.predict(test_x)
print("Accuracy:",accuracy_score(test_y, pred_y))
```

- ```
  Accuracy: 1.0
  ```
- Visualizations


  ```



SalePrice
Y N P
PavedDrive
CBlock PConc BrkTil Slab Stone Wood
Foundation
None BrkFace Stone BrkCmn
MasVnrType
RFn Unf Fin NA
GarageFinish
No Gd Av Mn NA
BsmtExposure
Gable Flat Hip Shed Gambrel Mansard
RoofStyle
Pave Grvl
Street
RL RM FV RH C (all)
MSZoning
Gtl Mod Sev
LandSlope
SBrkr FuseA FuseF FuseP Mix
Electrical
GasA GasW Floor OthW Wall Grav
Heating
TwnhsE 1Fam Duplex Twnhs 2fmCon
BldgType
Inside CulDSac FR2 Corner FR3
LotConfig
Normal Partial Abnorml Family Alloca AdjLand
SaleCondition
IR1 Reg IR2 IR3
LotShape
Y N
CentralAir
Lvl Bnk HLS Low
LandContour



- Interpretation of the Results

Many features have positive correlation with the target therefore, have to be retained.

There are a lot of outliers in the dataset.But, if we check the data description file, we see that, actually, some numerical variables, are categorical variables that were saved (codified) as numbers.So, some of these data points that seem to be outliers are, actually, categorical data with only one example of some category. Therefore, we need to keep those outliers.

the better the category of a variable, the higher the price, which means these variables will be important for a prediction model.

CONCLUSION

- **Key Findings and Conclusions of the Study**

  The best model is AdaBoostRegressor with accuracy score as 1.
  Saving the model
  In machine learning, while working with scikit learn library, we need to save the trained models in a file and restore them in order to reuse them to compare the model with other models, and to test the model on new data. The saving of data is called Serialization, while restoring the data is called Deserialization.

- **Learning Outcomes of the Study in respect of Data Science**

  Learnt more about treating 3 categories of data: Numerical, Categorical, ordinal.
  Missing values & Null values treatment.
  Correlation will show the importance of the feature with respect to the target.
  Ensembling techniques may lead to better results and have higher predictive accuracy. Ensemble methods are very useful when there is both linear and non-linear type of data in the dataset.

Limitations of this work and Scope for Future Work

The biggest pain-points we have identified are: finding the right data, getting access to it, understanding tables and their purpose, clean the data, and explain in laypeople's terms how they work links to the organization's performance. There is a lot of bias in the data being cleaned and treated. The methods vary as well as the opinion about features. It is subjective.