HOUSING PRICE PREDICTION

Submitted by :

Faiyaz Ansari

ACKNOWLEDGMENT

I want to extend my sincere regards to the below mentioned sources and references who helped me a lot in completion of my Project:

Team FlipRobo

Team DataTrained

scikit-learn official documentation

https://scikit-learn.org/stable/

geeksforgeeks

https://www.geeksforgeeks.org/

programiz

https://www.programiz.com

Machine Learning Mastery

https://machinelearningmastery.com/

Medium

https://www.medium.com

INTRODUCTION

- Business Problem Framing

We need to built a machine learning model that can predict the price of house for a given dataset. 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.

- Conceptual Background of the Domain Problem

The problem is related to housing price of Australia . Dataset has been given by the client Surprise Housing. We can convert this data into a structured Machine Learning model which can predict the actual value of the porperties and client can decide whether whether to invest in buying the houses or not.

- Review of Literature

We are required to model the price of houses with the available independent variables. This model will then be used
by the management to understand how exactly the prices vary with the variables. They can accordingly manipulate the
strategy of the firm and concentrate on areas that will yield high returns. Further, the model will be a good way for the
management to understand the pricing dynamics of a new market.

Analytical Problem Framing

- Mathematical/ Analytical Modeling of the Problem

  This is a supervised regression problem where we need to predict whether the house price , so that the company can sold it to a higher prices . Thus, we have used linear models like Linear Regression, tree based models like Decision Tree, Ensemble models like Random Forest for the modelling of this task.

  The models were tested on various performance metrices such as R2 score, mean absolute error and root mean square error to ensure generalization on future data.

- Data Sources and their formats

  The data was provided to us by our client which is a us based company. The datset contains 1460 rows i.e entries and 81 features i.e variables. Below I have attached the datsets with their data types.

```
train2.info()
```

```
 18  YearRemodAdd   1168 non-null   int64
 19  RoofStyle      1168 non-null   object
 20  RoofMatl       1168 non-null   object
 21  Exterior1st    1168 non-null   object
 22  Exterior2nd    1168 non-null   object
 23  MasVnrType     1168 non-null   object
 24  MasVnrArea     1168 non-null   float64
 25  ExterQual      1168 non-null   object
 26  ExterCond      1168 non-null   object
 27  Foundation     1168 non-null   object
 28  BsmtQual       1168 non-null   object
 29  BsmtCond       1168 non-null   object
 30  BsmtExposure   1168 non-null   object
 31  BsmtFinType1   1168 non-null   object
 32  BsmtFinSF1     1168 non-null   int64
 33  BsmtFinType2   1168 non-null   object
 34  BsmtFinSF2     1168 non-null   int64
 35  BsmtUnfSF      1168 non-null   int64
 36  TotalBsmtSF    1168 non-null   int64
 37  Heating        1168 non-null   object
 38  HeatingQC      1168 non-null   object
 39  CentralAir     1168 non-null   object
 40  Electrical     1168 non-null   object
 41  1stFlrSF       1168 non-null   int64
 42  2ndFlrSF       1168 non-null   int64
 43  LowQualFinSF   1168 non-null   int64
 44  GrLivArea      1168 non-null   int64
 45  BsmtFullBath   1168 non-null   int64
```

```
 48  HalfBath       1168 non-null   int64
 49  BedroomAbvGr   1168 non-null   int64
 50  KitchenAbvGr   1168 non-null   int64
 51  KitchenQual    1168 non-null   object
 52  TotRmsAbvGrd   1168 non-null   int64
 53  Functional     1168 non-null   object
 54  Fireplaces     1168 non-null   int64
 55  GarageType     1168 non-null   object
 56  GarageYrBlt    1168 non-null   float64
 57  GarageFinish   1168 non-null   object
 58  GarageCars     1168 non-null   int64
 59  GarageArea     1168 non-null   int64
 60  GarageQual     1168 non-null   object
 61  GarageCond     1168 non-null   object
 62  PavedDrive     1168 non-null   object
 63  WoodDeckSF     1168 non-null   int64
 64  OpenPorchSF    1168 non-null   int64
 65  EnclosedPorch  1168 non-null   int64
 66  3SsnPorch      1168 non-null   int64
 67  ScreenPorch    1168 non-null   int64
 68  PoolArea       1168 non-null   int64
 69  MiscVal        1168 non-null   int64
 70  MoSold         1168 non-null   int64
 71  YrSold         1168 non-null   int64
 72  SaleType       1168 non-null   object
 73  SaleCondition  1168 non-null   object
 74  SalePrice      1168 non-null   int64
dtypes: float64(3), int64(34), object(38)
memory usage: 684.5+ KB
```

- Data Preprocessing Done

The dataset contains lots of null values . I have replaced with median for continuous data and mode for categorical features .

In pre-processing, we have applied several techniques to find what works best. For example, all the outliers were first replaced with the threshold value as per the Inter Quartile Range. Then, we tried removing them as well.

The skewness of the data was removed using various transformation strategies such as log and power transformations of the features.

All the illogical fractional and negative values were either removed or treated with appropriate replacement.

- Data Inputs- Logic- Output Relationships

  The data is high-dimensional and can not be visualized directly, but the fact that linear regression is also performing well on the data indicates that data is almost linearly seperable in higher dimensional space.

  We have also used feature importance of Random Forest to have an insight of which features are the most important ones for the classification. Because the data is multi-collinear, we can not use feature weights by linear Regression to estimate a relationship between features and target. But, the pearson correlation gives an idea about how input and output are correlated.

- State the set of assumptions (if any) related to the problem under consideration

  We have assumed that any values above and below 1.5 times the IQR is an outlier and will either be treated (replaced) or removed.We have also assumed the features are gaussian distributed and skewness is removed using proper transformation.

- Hardware and Software Requirements and Tools Used
- The size of data is very small, therefore any system running on Windows 7 or higher, Mac or Linux based operating systems with 4 GB of RAM is more than sufficient for the given task. We can use any Python IDE or Jupyter notebooks or Google Colab for modelling.
- Below is the list of tools used for the task:

- sklearn for model building,
- pandas for reading and manipulation of data,
- numpy for numerical operations,
- matplotlib and seaborn for data visualization
- scipy for scientific operations and outlier detection
- joblib for saving the model

Model/s Development and Evaluation

- Identification of possible problem-solving approaches (methods)

  Since the task is regression based on supervised learning, we can use Linear Regression, Tree based classification algorithms, Ensemble models and Nearest Neighbors approach etc.
  Since there are a lot of outliers, we should use algorithms which are robust to outliers. Also, the size of data is in lakhs, so we should use algorithms with lower time complexity otherwise training time will be huge. Still, it is not that important factor.
  We may also use artificial neural networks but the size of data is not enough, so it has a high risk of over-fitting.

- Testing of Identified Approaches (Algorithms)

  Linear Regression
  Decision Tree Regressor
  K Neighbors Regressor
  Support Vector Regressor
  Gradient Boosting Regressor
  Random Forest Regressor
  Ada Boost Regressor

- Run and Evaluate selected models

  Describe all the algorithms used along with the snapshot of their code and what were the results observed over different evaluation metrics. listed above, we have used several regression algorithms for training and evaluating the model. Gradient boosting regressor came out as the best performing algorithm of all. Below are the snapshots of the code and results over various evaluation metrices:

- Key Metrics for success in solving problem under consideration

  We use R2 score as a metric for evaluating the performace of our regressor problem and also mean square error and root mean square are also used to evaluate the models.

	Model	R2score	Cross_val_score	Mean Squared Error	Root Mean Squared Error
0	LinearRegression	0.902005	0.839972	0.238762	0.488632
1	KNeighborsRegressor	0.768872	0.734554	0.353945	0.594933
2	DecisionTreeRegressor	0.688290	0.661278	0.420872	0.648747
3	SVR	0.898877	0.839726	0.240180	0.490081

- Visualizations

  A lot of plots were made as part of data visualisation. We used libraries like matplotlib, pandas and seaborn for data visualisation.

  The key findings are:

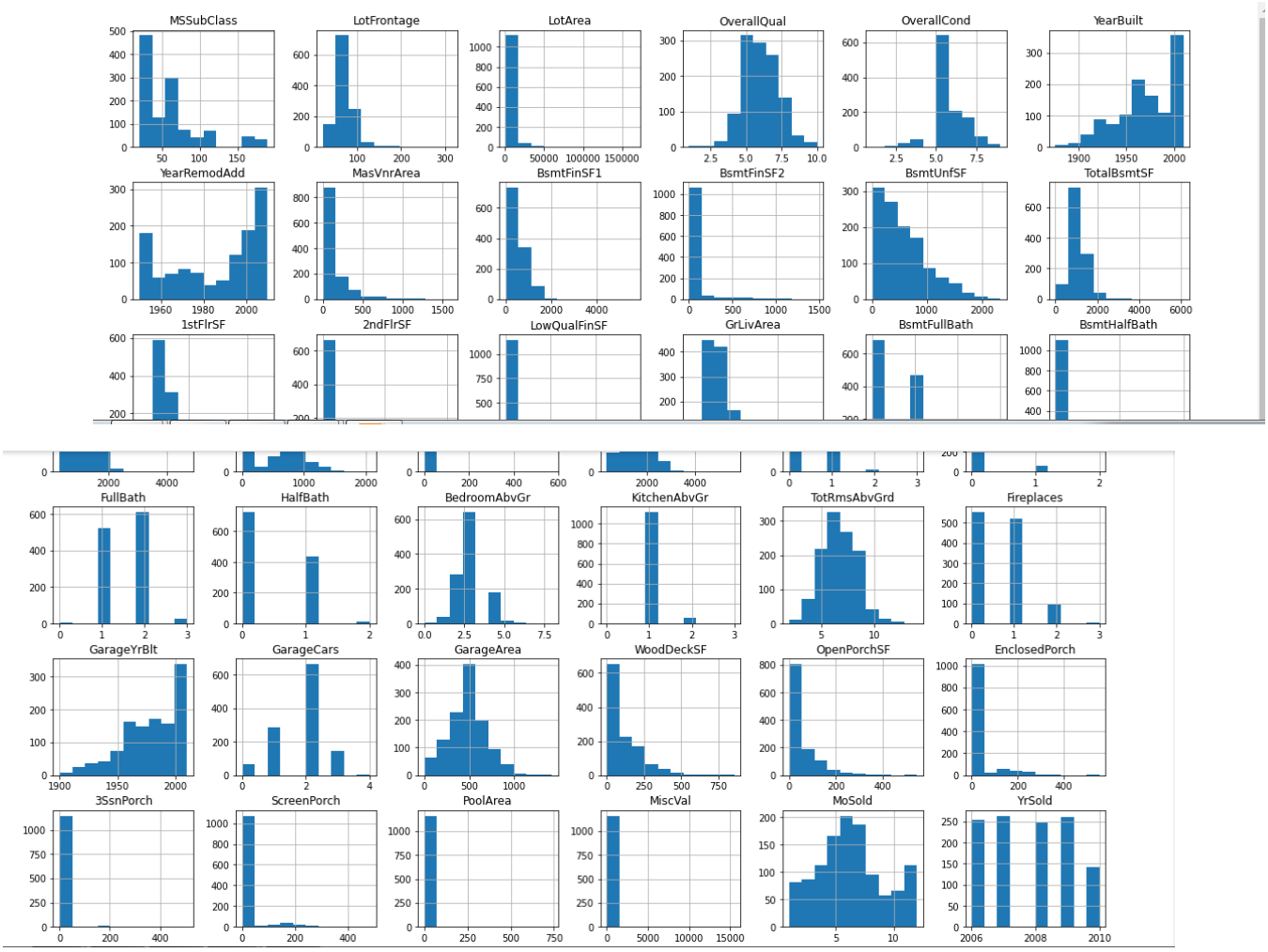
  There are positive values in all columns.

  There are exceptionally high values in many columns.

There are few columns which have fractional values.

The data is highly skewed.

Few of the columns are highly correlated.



- Interpretation of the Results

  The process for outlier removal, outlier treatment and skewness treatment were designed as per the interpretations from data visualization steps. Statistical methods were used to treat or remove

outliers. Also, we used various transformations in our pre-processing steps to remove the skewness. A lot of discrepencies were caught while visualising the data using methods like percentiles, value counts and plotting. Proper measures were adopted to clean the data and get rid of all the discrepencies without losing much data.

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

  Most of the variables are highly independent from the Target variable.

  Here is that variables and their respective correlation values.

  MSSubClass = -0.061
  LotFrontage = 0.32
  LotArea = 0.25
  OverallCond = -0.066
  BsmtFinSF1 = 0.36
  BsmtFinSF2 = -0.01
  BsmtUnfSF = 0.22
  2ndFlrSF = 0.33
  LowQualFinSF = -0.032
  BsmtFullBath = 0.21
  BsmtHalfBath == -0.011
  HalfBath = 0.3
  BedroomAbvGr = 0.16
  KitchenAbvGr = -0.13
  WoodDeckSF = 0.32
  OpenPorchSF = 0.34
  EnclosedPorch = -0.12
  3SsnPorch = 0.06
  ScreenPorch = 0.1
  PoolArea = 0.1
  MiscVal = -0.013
  MoSold = 0.073
  YrSold = -0.046

This variables can be dropped as they are not linearly dependent with target.
Also there are variables which are highly correlated with each other ,

Which can also be dropped .

GarageYrBlt = 0.78 with YearBuilt
TotRmsAbvGrd = 0.82 with GrLivArea
GarageArea = 0.88 with GarageCars

- Learning Outcomes of the Study in respect of Data Science

  Data visualization is the utmost important step for any Machine Learning project as it paves the foundation for data cleaning by giving us a detailed insight about the data. It also gives us an idea about which algorithms might work well for the given data. All the insights we received during data cleaning process are listed above. We planned our data cleaning in accordance with those insights.
  The main challenge was to clean the data without loss of it. The power of data was utilised in order to formulate proper cleaning strategies.
  Formulation of proper metrics was crucial for the problem.
  Selection of algorithm can be done according to the task in hand. For example, Linear Regression works very well for Linearly distributed data. Also, there are other factors like Latency, complexity, Interpretability that help us choose the model that can be used.
  In our case, Gradient boosting regressor was giving the best results. The model is a little complex, but we don't have a very low latency requirement as per our use case. It is highly interpretable (which is very necessary in our case) as we select features based on Pearson Correlation.

- Limitations of this work and Scope for Future Work

  The provided solution can be made better using more complex models like Artificial Neural Networks. Also, improvements can be made in data gathering and cleaning pipeline, as present data has a lot of discrepencies. We can also construct new features by consulting a domain expert.