



HOUSING: PRICE PREDICTION

Submitted by:

RUPAMANANDA NANDI

ACKNOWLEDGMENT

I am overwhelmed in all humbleness and gratefulness to acknowledge my depth to all those who have helped me to put these ideas, well above the level of simplicity and into something concrete.

I would like to express my special thanks of gratitude to my teacher as well as my mentor who gave me the golden opportunity to do this wonderful project on the topic (Housing price prediction), which also helped me in doing a lot of Research and I came to know about so many new things. I am really thankful to them.

Any attempt at any level can't be satisfactorily completed without the support and guidance of my parents.

INTRODUCTION

- **Business Problem Framing**

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.

- **Conceptual Background of the Domain Problem**

To predict the price of a house there are lot of factors that are crucial for the pricing. For example: Location, furnished, condition, type of foundation.

- **Review of Literature**

Data contains 1460 entries each having 81 variables. We need to concentrate on the null values, missing values, Discrete features. We need to find the relation between each discrete feature. Perform graphs and logarithm transformation. Also check for the outliers in this project.

- **Motivation for the Problem Undertaken**

This project helps me find the correct pathway of a dataset using python and distinguish between parameters and increase the accuracy. Also help me increase my rank in GitHub global ranking.

Analytical Problem Framing

- Mathematical/ Analytical Modeling of the Problem

In simpler words, it is a process of comparing variables at a 'neutral' or 'standard' scale. It helps to obtain same range of values. Normally distributed data is easy to read and interpret. As shown below, in a normally distributed data, 99.7% of the observations lie within 3 standard deviations from the mean. Also, the mean is zero and standard deviation is one. Transformations for Skewed Distribution

- Data Sources and their formats

Out[3]:

	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	...	PoolArea	PoolQC	Fence	MiscFeature	MiscVal	...
0	127	120	RL	NaN	4928	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	...
1	889	20	RL	95.0	15865	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	...
2	793	60	RL	92.0	9920	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	...
3	110	20	RL	105.0	11751	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	MnPrv	NaN	0	...
4	422	20	RL	NaN	16635	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0	...

5 rows × 81 columns

- Data Preprocessing Done

For data cleaning we have to remove null values, missing values, data integration, data reduction, data transformations, find outliers

- Data Inputs- Logic- Output Relationships

Establish relation between discrete and categorical features.

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

The categorical features are real life experience ,for example Lot Frontage, Street, Utilities, HouseStyle, OverallCond.This all factors decide the pricing of the house

- Hardware and Software Requirements and Tools Used

```
In [74]: #Loading libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
plt.rcParams['figure.figsize'] = (10.0, 8.0)
import seaborn as sns
from sklearn.model_selection import RandomizedSearchCV
from scipy import stats
from scipy.stats import norm
```

Model/s Development and Evaluation

- Identification of possible problem-solving approaches (methods)

This method is done to handle categorical features.

```
Out[38]: 39

In [39]: def category_onehot_multcols(multcolumns):
df_final=final_df
i=0
for fields in multcolumns:

    print(fields)
    df1=pd.get_dummies(final_df[fields],drop_first=True)

    final_df.drop([fields],axis=1,inplace=True)
    if i==0:
        df_final=df1.copy()
    else:
        df_final=pd.concat([df_final,df1],axis=1)
    i=i+1

df_final=pd.concat([final_df,df_final],axis=1)

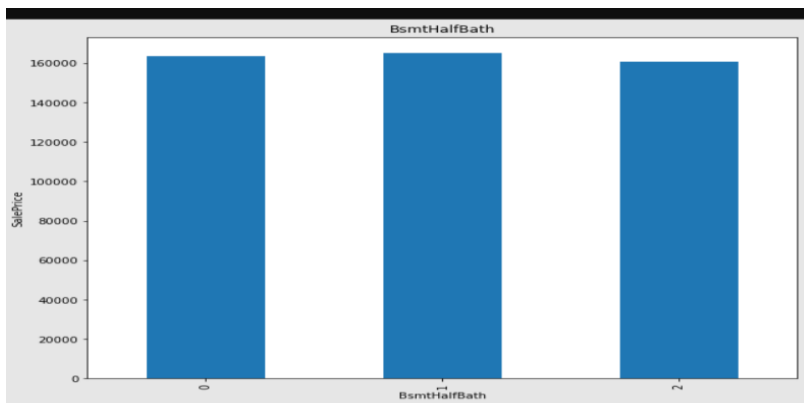
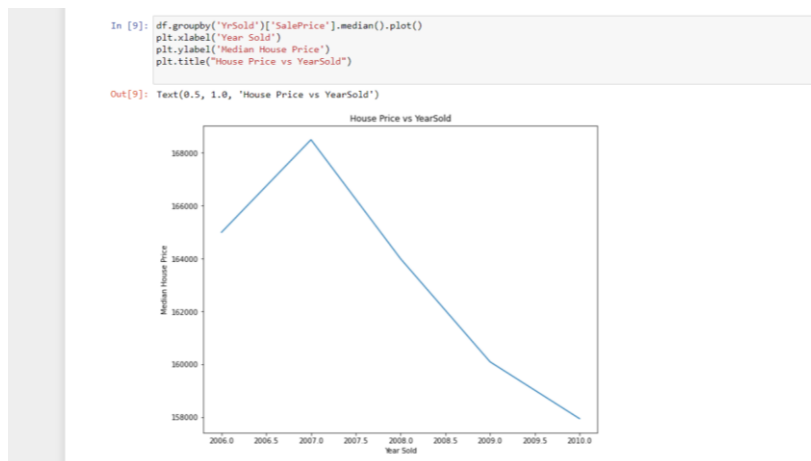
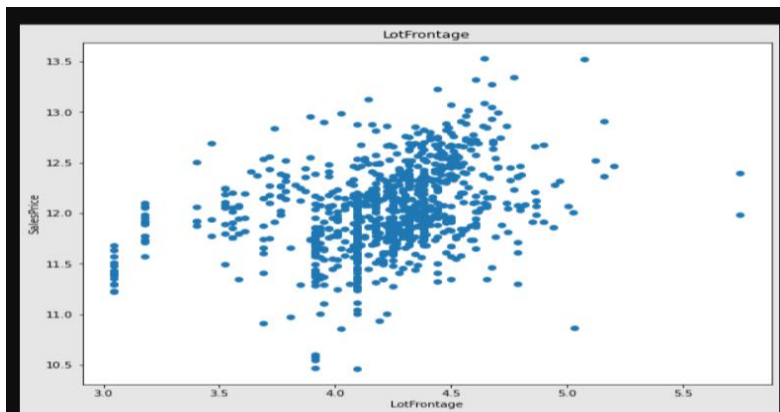
return df_final
```

- Run and Evaluate selected models

Models used classifier=xgboost.XGBRegressor() for Hyper parameter optimization and randomCV to increase the accuracy of the model.

- Visualizations

Graphs, histogram, plot bar, scatter plot are used.



Comparison between discrete features with sales price






```
MSSubClass',
'OverallQual',
'OverallCond',
'LowQualFinSF',
'BsmtFullBath',
'BsmtHalfBath',
'FullBath',
'HalfBath',
'BedroomAbvGr',
'KitchenAbvGr',
'TotRmsAbvGrd',
'Fireplaces',
```

CONCLUSION

- Key Findings and Conclusions of the Study

Youtube, GitHub, Stackoverflow

- Learning Outcomes of the Study in respect of Data Science

-  Linear regression: OLS, regularization, linear classifier
-  Develop an appreciation for what is involved in Learning models from data
-  Understand a wide variety of learning algorithms
-  Understand how to evaluate models generated from data.
-  Apply the algorithms to a real problem, optimize the models learned and report on the expected accuracy that can be achieved by applying the models

- Limitations of this work and Scope for Future Work

This dataset can help a buyer in terms of price when searching for a house. The factors which are highly dependable for the price.

It might not provide the actual price though with the visualization the buyer can have a high spectrum of deciding a fair price.