HOUSE PRICE PREDICTION

Mini Project
(Fundamentals of Machine Learning)

Submitted by:

Anshu Kumar (9919103112)

Department of CSE/IT
Jaypee Institute of Information Technology University, Noida

December 2021

ABSTRACT:

Home price forecasting is an important part of any real estate business. Here, we used machine learning algorithm to trace past local activities and to find a useful model of real estate buyers and sellers as well. We see a huge set of unparalleled housing prices variations it's good. Previously predictions were made in person and prices were calculated in pen and paper. We can remove all this distraction and the values will be calculated on the machine learning model. This the model will take various input parameters about the location and based on that it will predict values. To do this we will be using retrospective techniques and will be training our model in a variety of ways a separate data set. This model will be ideal for both real estate agents and buyers. It will help to maintain openness between the two of them. To make this model stronger we will train this a large and comprehensive set of databases to produce approximately results.

INTRODUCTION:

Having lived in India for so many years if there is something that I can talk about for sure is that rental and housing prices keeps growing. The housing prices keeps on increasing at an effective rate on daily basis. After the housing crisis in the year 2008, the housing prices grew at higher rate especially in the various housing brands. So, to maintain the transparency among customers and also the comparison can be made easy through this model. If any customer is finding the price of the house higher than the price predicted by the model then he/she can simply crossly verify the prices or reject the house. This model will ultimately remove dealers and third-party sellers from the picture and in turn would increase the productivity for the real estate business.

BACKGROUND STUDY:

The project mainly involves two parts. The first is to develop a model using machine learning algorithm and finally train it over a set of big data to produce more accurate results. To improve the model, we will be using the Google Collab as IDE, python as programming language to use machine learning algorithms.

Stage 1: Selection of Data

In the first step I looked at whether all the features I was going to use had missing numbers and their type the type of data they have (Numbers or categories). To address the shortages, I have replaced the missing one's prices have normal values. On the side of the numerical values, I have converted them all into categories prices on numerical values as I have decided to use all the features at the same time, I realize that the score may be reduced if the feature is not appropriate. I have rearranged all the category values so I can get a better data structure when using the model as it will only use numerical values.

Stage 2: Pre-Processing by Standardization

After selecting the features that I would use in the model, I had to check that the data contained external objects using the box plot as shown in the graphs. From the plans, I have seen from the first data that many of the features often contained some external features. I removed the outside by deleting all the deviant data from it Mean with a larger number than I specified in that particular data.

Stage 3: Data Modelling

This is the last step in starting a model that I should use. I have considered different ways to back off such as, Lasso, Linear Regression, Ridge etc. However, I noticed that different models offer different results depending on how the data is processed. For my data Linear Regression were giving me an acceptable performance unlike Lasso. So, overall, in the project I used Linear Regression Model.

ALGORITHM:

LINEAR REGRESSION:

Linear Regression is a machine learning algorithm based on supervised learning. Perform a regression function. Regression Models gives a prediction value based on independent variables. It is widely used to find relationships between variables and predictions. Different regression models vary depending on - the type of relationship between dependent and independent variables. Linear regression performs the task to predict a dependent variable value (y) based on a given independent variable (x). So, this regression technique finds out a linear relationship between x (input) and y(output). Hence, the name is Linear Regression.

Hypothesis Function for Linear Regression:

$$y = \theta_1 + \theta_2.x$$

While training the model we are given:

x: input training data (univariate – one input variable(parameter))

y: labels to data (supervised learning)

When training the model – it fits the best line to predict the value of y for a given value of x. The model gets the best regression fit line by finding the best θ1 and θ2 values.

θ1: intercept

θ2: coefficient of x

Once we find the best θ1 and θ2 values, we get the best fit line. So, when we are finally using our model for prediction, it will predict the value of y for the input value of x.

IMPLEMENTATION:

CODE:

```
import pandas as pd
import numpy as np
import time
```

```
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
def main():
    df = load_data()
    df = preprocess_data(df)
    print('House Price Prediction for Silicon Valley of India -
 Bangalore')
    print('Just Enter the following details and we will predict the price
 of your **Dream House**')
    print('Only Enter Numeric Values in the Following Fields')
    print("Total BHK")
    bhk = int(input())
    print("Area in Square Feet")
    area = int(input())
    print("Total Bathrooms")
    baths = int(input())
    print("Total Balcony, ['0', '1', '2', '3']")
    balcony = int(input())
    submit = True
    if submit:
        if bhk and area and baths and balcony:
          print('Predicting...')
          time.sleep(2)
          bhk, area, baths, balcony = int(bhk), int(area), int(baths), in
t(balcony)
          x_test = np.array([[bhk, area, baths, balcony]])
          prediction = predict(df, x_test)
          print("Your **Dream House** Price is",predict(df,x_test),"lacs"
)
        else:
            print('Please Enter All the Details Again')
def train_model(df):
    global scaler
    X, y = df.iloc[:, :-1].values, df.iloc[:, -1].values
    scaler = StandardScaler().fit(X)
    X = scaler.transform(X)
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0
.2, random_state=23)
    model = LinearRegression().fit(X_train, y_train)
    return model, scaler
def predict(df, x_test):
    model, scaler = train_model(df)
    X_test = scaler.transform(x_test)
    y_pred = model.predict(X_test)
    return round(y_pred[0], 2)
def load_data():
    return pd.read_csv('/content/Bengaluru_House_Data.csv')
def preprocess_data(df):
```

```
    df = df.loc[:, ['size', 'total_sqft', 'bath', 'balcony', 'price']]
    df.dropna(inplace=True)
    df = df[df['size'].str.contains('BHK', na=False)]
    df['size'] = df['size'].str.replace(r'\D', '').astype(int)
    df['total_sqft'] = df['total_sqft'].str.extract(r'(\d+)', expand=Fals
e)
    df['bath'] = df['bath'].astype(int)
    df['balcony'] = df['balcony'].astype(int)
    df['total_sqft'] = df['total_sqft'].astype(int)
    return df
if __name__ == '__main__':
    main()
```

EXPERIMENTAL RESULT:

```
House Price Prediction for Silicon Valley of India - Bangalore
Just Enter the following details and we will predict the price of your **Dream House**
Only Enter Numeric Values in the Following Fields
Total BHK
4
Area in Square Feet
1500
Total Bathrooms
3
Total Balcony, ['0', '1', '2', '3']
2
Predicting...
Your **Dream House** Price is 96.27 lacs
```

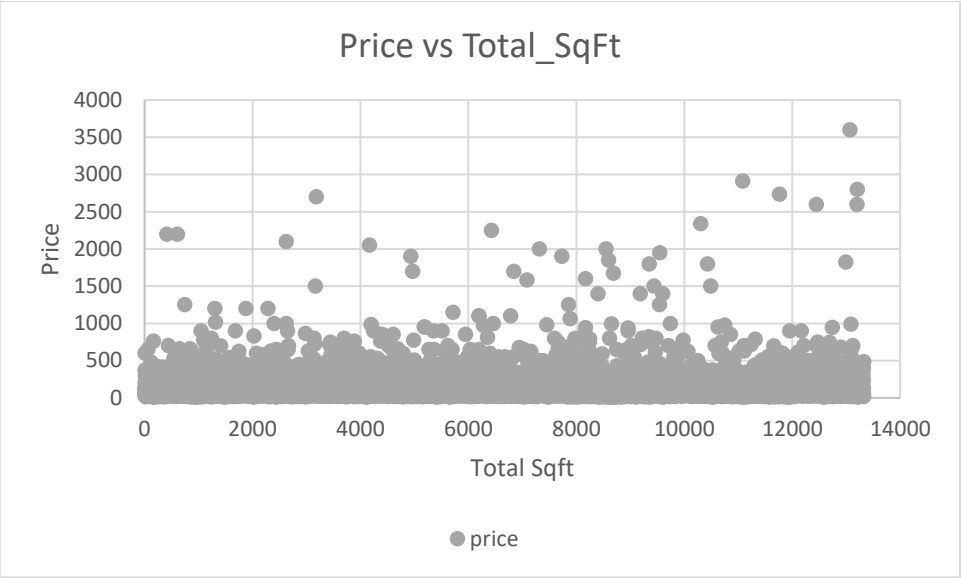
About Dataset Used:

The Dataset contain 9 columns named as Area type, Availability, Location, Size, Society, Total_sqft, Bathroom, Balcony, Price.

	A	B	C	D	E	F	G	H	I
1	area_type	availability	location	size	society	total_sqft	bath	balcony	price
2	Super built-up Area	19-Dec	Electronic City Phase II	2 BHK	Coomee	1056	2	1	39.07
3	Plot Area	Ready To Move	Chikka Tirupathi	4 Bedroom	Theanmp	2600	5	3	120
4	Built-up Area	Ready To Move	Uttarahalli	3 BHK		1440	2	3	62
5	Super built-up Area	Ready To Move	Lingadheeranahalli	3 BHK	Soiewre	1521	3	1	95
6	Super built-up Area	Ready To Move	Kothanur	2 BHK		1200	2	1	51
7	Super built-up Area	Ready To Move	Whitefield	2 BHK	DuenaTa	1170	2	1	38
8	Super built-up Area	18-May	Old Airport Road	4 BHK	Jaades	2732	4		204
9	Super built-up Area	Ready To Move	Rajaji Nagar	4 BHK	Brway G	3300	4		600
10	Super built-up Area	Ready To Move	Marathahalli	3 BHK		1310	3	1	63.25
11	Plot Area	Ready To Move	Gandhi Bazar	6 Bedroom		1020	6		370

After pre-processing the data, the main features which left for model training and testing is

[BHK, Total_Sqft, Bathroom, Balcony]



CONCLUSION:

We can therefore say that our goal is achievable as we have successfully identified all our parameters as mentioned earlier. At first, we collected the data and then processed the data by cleaning and training it. Then we do a variety data analysis to get all the logical information from the data. We also tried to visualize the data we use a variety of visual aids such as maps, graphs, charts, etc. After visualizing our data, we split it up two parts of the test and the purpose of the training. After this, we started training our model using one of the most popular ones known algorithms linear regression.