EDICTINGHOUSE PRICE USING MACHINE LEARNING

BATCH MEMBER

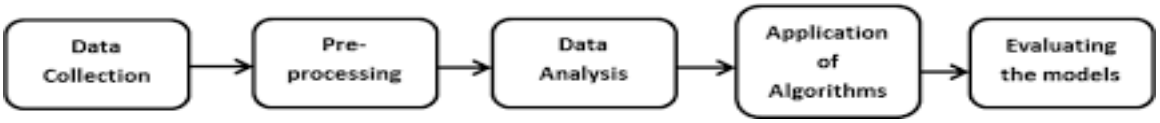
961621205009 :Anjana K
Phase 3 submission document

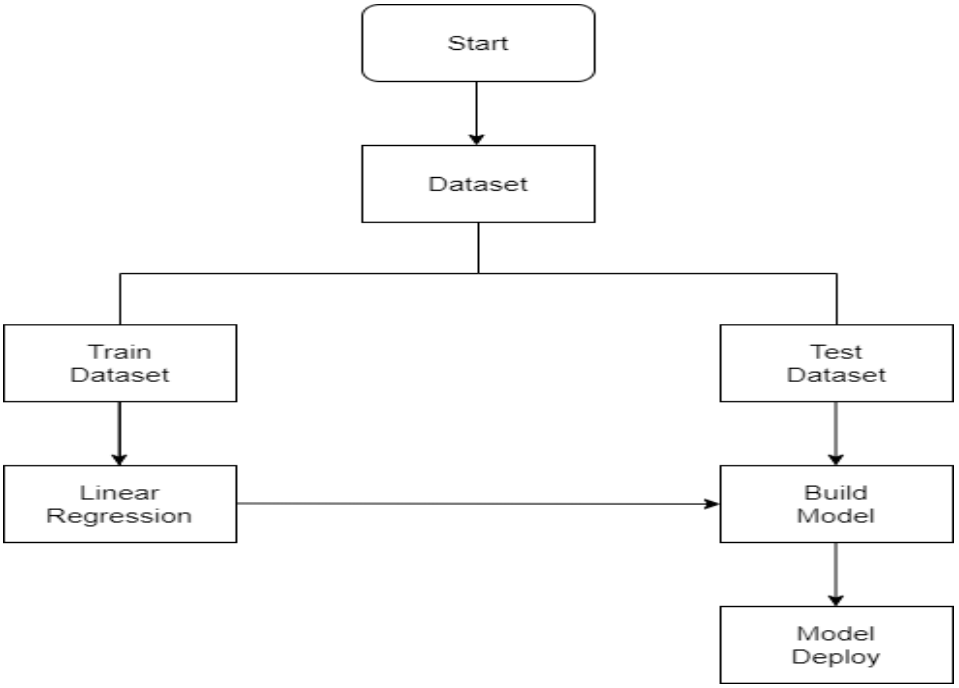
Project Title: House Price Prediction

Phase 3: Development Part 1

Topic: Start building the house price prediction model by loading and pre-processing the dataset

BLOCK DIAGRAM





PROPOSED SYSTEM PHASES

Phase 1: Collection of data

Data processing techniques and processes are numerous. We collecteddata for USA/Mumbai real estate properties from various real estate websites. The data would be having attributes such as Location, carpet area, built-up area, age of the property, zip code, price, no of bedroomsetc. We must collect the quantitative data which is structured and categorized. Data collection is needed before any kind of machine learning research is carried out. Dataset validity is a must otherwise there is no point in analyzing the data.

Phase 2: Data preprocessing

Data preprocessing is the process of cleaning our data set. There mightbe missing values or outliers in the dataset. These can be handled by data cleaning. If there are many missing values in a variable we will drop those values or substitute it with the average value.

Phase 3: Training the model

Since the data is broken down into two modules: a Training set and Testset, we must initially train the model. The training set includes the target variable. The decision tree regressor algorithm is applied to the training data set. The Decision tree builds a regression model in the form of a tree structure.

Phase 4: Testing and Integrating with UI

The trained model is applied to test dataset and house prices are predicted. The trained model is then integrated with the front end using Flask in python

ALTERNATIVE REGRESSOR (XG BOOST REGRESSOR)

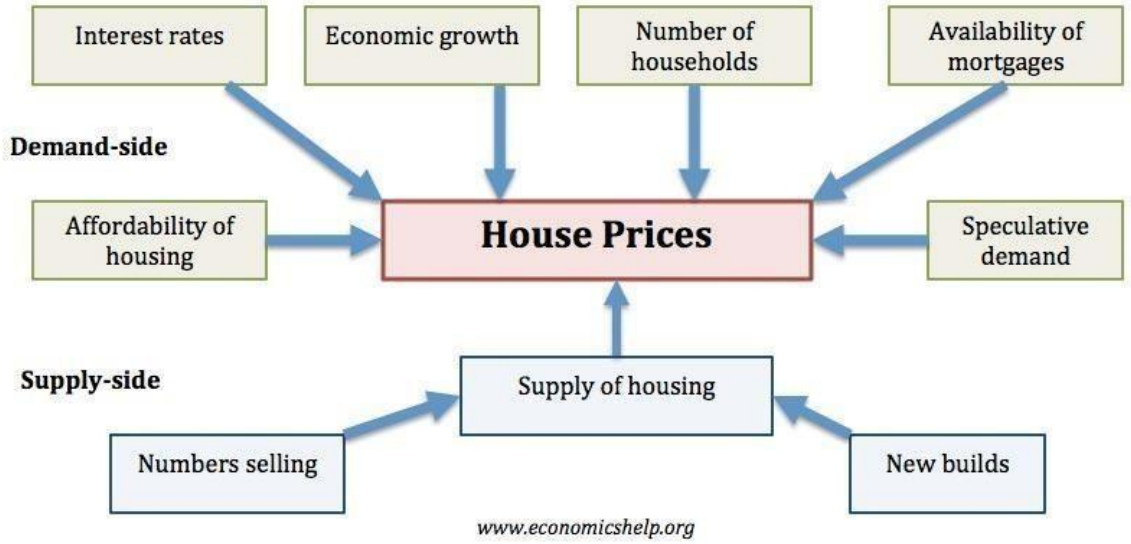
The results of the regression problems are continuous or real values. Some commonly used regression algorithms are Linear Regression and Decision Trees. There are several metrics involved in regression like root-mean-squared error (RMSE) and mean-squared-error (MAE). These are some key members of XGBoost models, each plays an important role.

- **RMSE:** It is the square root of mean squared error (MSE).
- **MAE:** It is an absolute sum of actual and predicted differences, but it lacks mathematically, that's why it is rarely used, as compared to other metrics.

XGBoost is a powerful approach for building supervised regression models. The validity of this statement can be inferred by knowing about its (XGBoost) objective function and base learners.

FACTORS THAT AFFECT HOUSE PRICING

In order to predict house prices, first we have to understand the factors that affect house pricing.



- **Economic growth.** Demand for housing is dependent upon income. With higher economic growth and rising incomes, people will be able to spend more on houses; this will increase demand and push up prices. In fact, demand for housing is often noted to be income elastic (luxury good); rising incomes leading to a bigger % of income being spent on houses. Similarly, in a recession, falling incomes will mean people can't afford to buy and those who lose their job may fall behind on their mortgage payments and end up with their home repossessed.

- **Unemployment.** Related to economic growth is unemployment. When unemployment is rising,fewer people will be able to afford a house. But, even the fear of unemployment may discouragepeople from entering the property market.

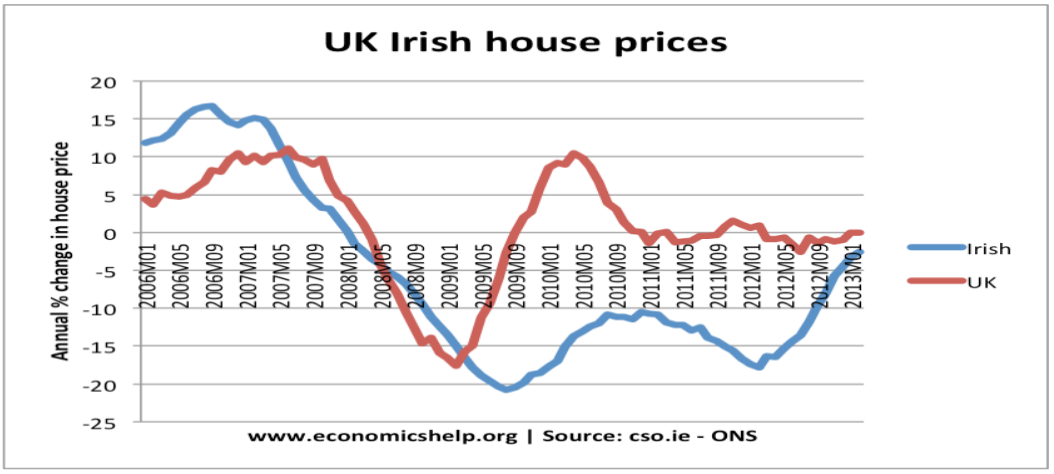
- **Interest rates.** Interest rates affect the cost of monthly mortgage payments. A period of high- interest rates will increase cost of mortgage payments and will cause lower demand for buying a house. High-interest rates make renting relatively

more attractive compared to buying. Interest rates have a bigger effect if homeowners have large variable mortgages. For example, in 1990-92, the sharp rise in interest rates caused a very steep fall in UK house prices because many homeowners couldn't afford the rise in interest rates.

- **Consumer confidence**. Confidence is important for determining whether people want to take therisk of taking out a mortgage. In particular expectations towards the housing market is important; if people fear house prices could fall, people will defer buying.

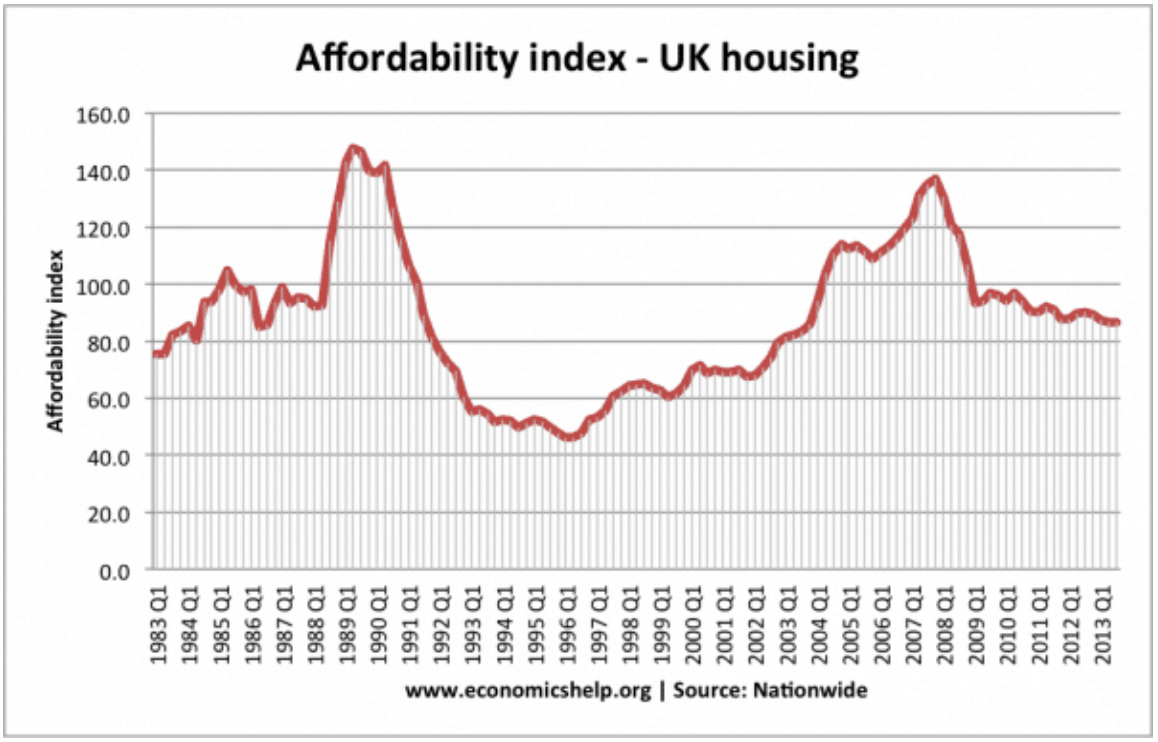
- **Mortgage availability**. In the boom years of 1996-2006, many banks were very keen to lend mortgages. They allowed people to borrow large income multiples (e.g. five times income). Also, banks required very low deposits (e.g. 100% mortgages). This ease of getting a mortgage meant that demand for housing increased as more people were now able to buy. However, since the credit crunch of 2007, banks and building societies struggled to raise funds for lending on the money markets. Therefore, they have tightened their lending criteria requiring a bigger deposit to buy a house. This has reduced the availability of mortgages and demand fell.

- **Supply**. A shortage of supply pushes up prices. Excess supply will cause prices to fall. For example, inthe Irish property boom of 1996-2006, an estimated 700,000 new houses were built. When the property market collapsed, the market was left with a fundamental oversupply. Vacancy rates reached 15%, and with supply greater than demand, prices fell.



By contrast, in the UK, housing supply fell behind demand. With a shortage, UK house prices didn't fall as much as in Ireland and soon recovered – despite the ongoing credit crunch. The supply of housing depends on existing stock and new house builds. Supply of housing tends to be quite inelastic because to get planning permission and build houses is a time-consuming process. Periods of rising house prices may not cause an equivalent rise in supply, especially in countries likethe UK, with limited land for home-building.

- **Affordability/house prices to earnings.** The ratio of house prices to earnings influences the demand. As house prices rise relative to income, you would expect fewer people to be able to afford. For example, in the 2007 boom, the ratio of house prices to income rose to 5. At this level, house prices were relatively expensive, and we saw a correction with house prices falling.



Another way of looking at the affordability of housing is to look at the percentage of take-home paythat is spent on mortgages. This takes into account both house prices, but mainly interest rates and the cost of monthly mortgage payments. In late 1989, we see housing become very unaffordable because of rising interest rates. This caused a sharp fall in prices in 1990-92.

- **Geographical factors.** Many housing markets are highly geographical. For example, national house prices may be falling, but some areas (e.g. London, Oxford) may still see rising prices. Desirable areas can buck market trends as demand is high, and supply limited. For example, houses near goodschools or a good rail link may have a significant premium to other areas. This graph shows that first time buyers in London face much more expensive house prices – over 9.0 times earnings compared to the north, where house prices are only 3.3 times earnings.

SAMPLE CODE

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt

%matplotlib inline

HouseDF = pd.read_csv('USA_Housing.csv')
HouseDF.head()
HouseDF=HouseDF.reset_index()
HouseDF.head()
HouseDF.info()
HouseDF.describe()
HouseDF.columns
sns.pairplot(HouseDF)
sns.distplot(HouseDF['Price’])
sns.heatmap(HouseDF.corr(), annot=True)

X = HouseDF[['Avg. Area Income', 'Avg. Area House Age', 'Avg. Area Number of Rooms','Avg. Area Number of Bedrooms', 'Area Population']]

y = HouseDF['Price’]

from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.4, random_state=101)

from sklearn.linear_model import minmaxscaler

lm = minmaxscaler(feature_range=(0,1))

lm.fit_transform(X_train,y_train)

print(lm.intercept_)

coeff_df = pd.DataFrame(lm.coef_,X.columns,columns=['Coefficient’])
coeff_df
```

```
from keras.layers import Dense,Dropout,LSTM
from keras.models import Sequential
model = Sequential()
model.add(LSTM(units = 50,activation = 'relu',return_sequences = True,input_shape =
(x_train.shape[1], 1)))
model.add(Dropout(0.2))

model.add(LSTM(units = 60,activation = 'relu',return_sequences = True))
model.add(Dropout(0.3))

model.add(LSTM(units = 80,activation = 'relu',return_sequences = True))
model.add(Dropout(0.4))

model.add(LSTM(units = 120,activation = 'relu'))
model.add(Dropout(0.5))

model.add(Dense(units = 1))

model.compile(optimizer='adam', loss = 'mean_squared_error')
model.fit(x_train, y_train,epochs=50)

print(lm.intercept_)

coeff_df = pd.DataFrame(lm.coef_,X.columns,columns=['Coefficient'])
coeff_df

predictions = lm.predict(X_test)

scale_factor = 1/0.02099517
y_predicted = y_predicted * scale_factory
y_test = y_test * scale_factor

plt.scatter(y_test,predictions)

sns.distplot((y_test-predictions),bins=50);
```

```
plt.figure(figsize=(12,6))
plt.plot(y_test,'b',label = 'Original Price')
plt.plot(y_predicted,'r',label = 'Predicted Price')
plt.xlabel('Time')
plt.ylabel('Price')
plt.legend()
plt.show()

from sklearn import metrics

print('MAE:', metrics.mean_absolute_error(y_test, predictions))
print('MSE:', metrics.mean_squared_error(y_test, predictions))
print('RMSE:', np.sqrt(metrics.mean_squared_error(y_test, predictions)))
```

ADVANTAGE OF LSTM OVER OTHER MODELS

The LSTM model can be tuned for various parameters such as changing the number of LSTM layers, adding dropoutvalue or increasing the number of epochs.

Long Short Term Memory (LSTM)

LSTMs are widely used for sequence prediction problems and have proven to be extremely effective. The reason they work so well is because LSTM is able to store past information that is important, and forget the information that is not. LSTM has three gates:

The input gate: The input gate adds information to the cell state

The forget gate: It removes the information that is no longer required by the model. The output gate: Output Gate at LSTM selects the information to be shown as output.

EXPLANATION OF THE OUTPUT RESULTS AND THE DATASET

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	price
0	0.00632	18.0	2.31	0.0	0.538	6.575	65.2	4.0900	1.0	296.0	15.3	396.90	4.98	24.0
1	0.02731	0.0	7.07	0.0	0.469	6.421	78.9	4.9671	2.0	242.0	17.8	396.90	9.14	21.6
2	0.02729	0.0	7.07	0.0	0.469	7.185	61.1	4.9671	2.0	242.0	17.8	392.83	4.03	34.7
3	0.03237	0.0	2.18	0.0	0.458	6.998	45.8	6.0622	3.0	222.0	18.7	394.63	2.94	33.4
4	0.06905	0.0	2.18	0.0	0.458	7.147	54.2	6.0622	3.0	222.0	18.7	396.90	5.33	36.2

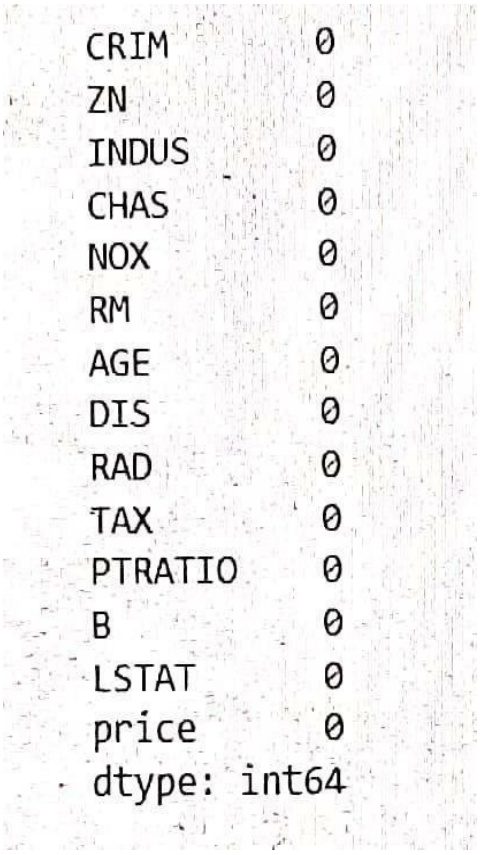
First we import a sample data from sklearn library , you can get different types of sample data from Kaggle. The data taken here is the data of various parameters and the house prices in a given city called boston in the year between 1970 to 2020.

Here the data parameters are explained as follows:

```
1. CRIM      per capita crime rate by town
2. ZN        proportion of residential land zoned for lots over
             25,000 sq.ft.
3. INDUS     proportion of non-retail business acres per town
4. CHAS      Charles River dummy variable (= 1 if tract bounds
             river; 0 otherwise)
5. NOX       nitric oxides concentration (parts per 10 million)
6. RM        average number of rooms per dwelling
7. AGE       proportion of owner-occupied units built prior to 1940
8. DIS       weighted distances to five Boston employment centres
9. RAD       index of accessibility to radial highways
10. TAX      full-value property-tax rate per $10,000
11. PTRATIO  pupil-teacher ratio by town
12. B        1000(Bk - 0.63)^2 where Bk is the proportion of blacks
             by town
13. LSTAT    % lower status of the population
14. MEDV     Median value of owner-occupied homes in $1000's
```

Here for understanding purpose we have taken first 5 index/instance of data and printed them. In total there are 506 rows ofdata from the dataset , of which we have printed first 5 rows using head() function. There are 14 columns in total, i.e, 13 colums containing data of the place, and the 14^{th} column is the target column which contains the house prices.

Then we check if our data has some null values i.e missing values. Since if the data is incomplete , then there will be error during processing state which may lead to loss of accuracy in predicting model. Here in our given data , there is nomissing value as we can see.



```
CRIM       0
ZN         0
INDUS      0
CHAS       0
NOX        0
RM         0
AGE        0
DIS        0
RAD        0
TAX        0
PTRATIO    0
B          0
LSTAT      0
price      0
dtype: int64
```

Since our data contains no missing value, the program will skip the dropping phase in data processing, where data isdropped to increase accuracy and fit missing values in a way so that it is suitable for modelling.

Next we try to describe the data in such a way so that both people and machine find it easy to understand the given data . In order to do thiswe use the describe() function.

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	price
count	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000	506.000000
mean	3.613524	11.363636	11.136779	0.069170	0.554695	6.284634	68.574901	3.795043	9.549407	408.237154	18.455534	356.674032	12.653063	22.532806
std	8.601545	23.322453	6.860353	0.253994	0.115878	0.702617	28.148861	2.105710	8.707259	168.537116	2.164946	91.294864	7.141062	9.197104
min	0.006320	0.000000	0.460000	0.000000	0.385000	3.561000	2.900000	1.129600	1.000000	187.000000	12.600000	0.320000	1.730000	5.000000
25%	0.082045	0.000000	5.190000	0.000000	0.449000	5.885500	45.025000	2.100175	4.000000	279.000000	17.400000	375.377500	6.950000	17.025000
50%	0.256510	0.000000	9.690000	0.000000	0.538000	6.208500	77.500000	3.207450	5.000000	330.000000	19.050000	391.440000	11.360000	21.200000
75%	3.677083	12.500000	18.100000	0.000000	0.624000	6.623500	94.075000	5.188425	24.000000	666.000000	20.200000	396.225000	16.955000	25.000000
max	88.976200	100.000000	27.740000	1.000000	0.871000	8.780000	100.000000	12.126500	24.000000	711.000000	22.000000	396.900000	37.970000	50.000000

Counts refers to the number of instances of data in each column i.e 506 since there are 506 rows of data for each columnMean refers to mean value of data in given colum.

Std means the standard value i.e the most common value in given set of data for a particular column.

Min refers the least data value in each column.

Max refers to the maximum data value in each column.

25% refers that 25 percentile of the data in that column is equal to or below that value.

Next we try to understand the correlation between the different values, in order to do that, the best way is by using heat map. Heat map is a representation of data in the form of a map or diagram in which data values are represented as colours.

Correlation is a statistical measure that expresses the extent to which two variables are linearly related (meaning they change together at a constant rate)

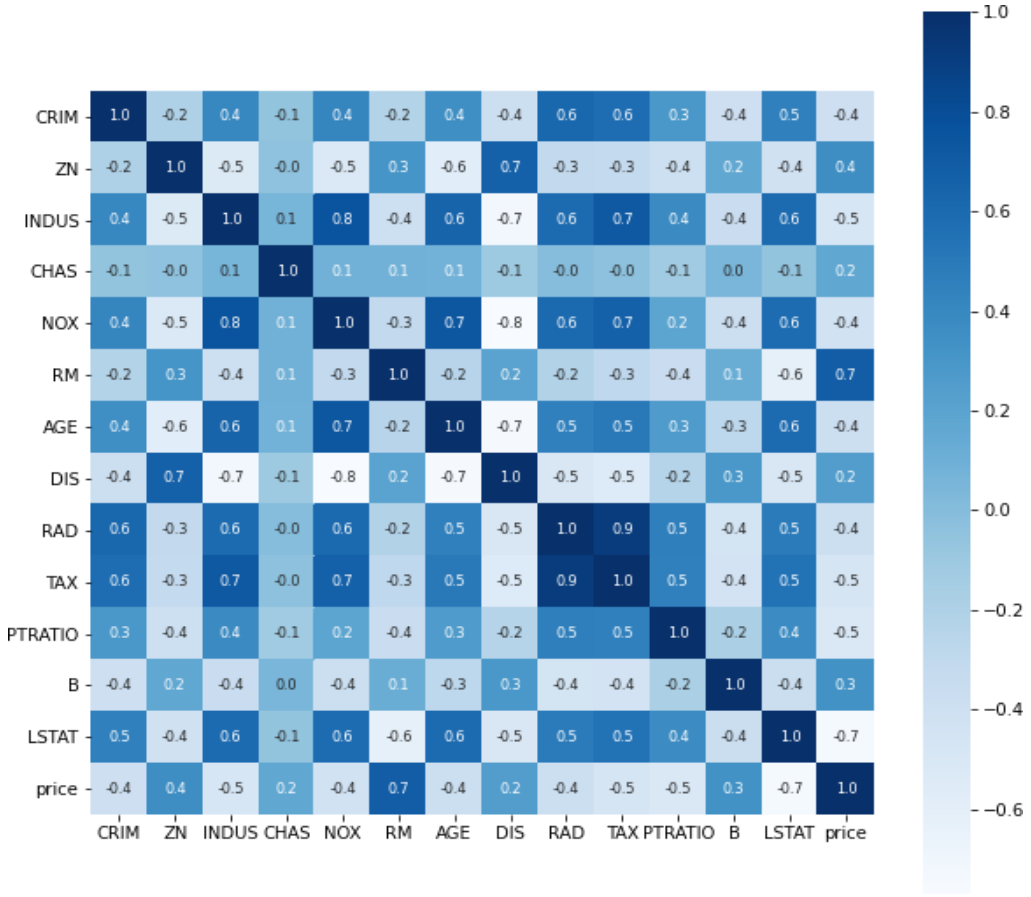
There are two types of correlation, they are:

1. Positive correlation: A positive correlation is a relationship between two variables that move in tandem—that is, inthe same direction. A positive correlation exists when one variable decreases as the other variable decreases, or onevariable increases while the other increases.

2. Negative correlation: Negative correlation is a relationship between two variables in which one variable increasesas the other decreases, and vice versa.

   In statistics, a perfect negative correlation is represented by the value -1.0, while a 0 indicates no correlation, and +1.0 indicates a perfect positive correlation. A perfect negative correlation means the relationship that existsbetween two variables is exactly opposite all of the time. These are two types of correlation are representednumerically and as well as by shade of colour in the heat map.

HEATMAP – for better understanding of which place is best suited for individual personal preference based on given dataset. This uses correlation concept



Next we split our data into variables x and y , in order to train our model to predict data.

```
        CRIM    ZN  INDUS  CHAS    NOX    RM   AGE     DIS  RAD  TAX  \
0    0.00632  18.0   2.31   0.0  0.538  6.575  65.2  4.0900  1.0  296.0
1    0.02731   0.0   7.07   0.0  0.469  6.421  78.9  4.9671  2.0  242.0
2    0.02729   0.0   7.07   0.0  0.469  7.185  61.1  4.9671  2.0  242.0
3    0.03237   0.0   2.18   0.0  0.458  6.998  45.8  6.0622  3.0  222.0
4    0.06905   0.0   2.18   0.0  0.458  7.147  54.2  6.0622  3.0  222.0
..       ...   ...    ...   ...    ...    ...   ...     ...  ...    ...
501  0.06263   0.0  11.93   0.0  0.573  6.593  69.1  2.4786  1.0  273.0
502  0.04527   0.0  11.93   0.0  0.573  6.120  76.7  2.2875  1.0  273.0
503  0.06076   0.0  11.93   0.0  0.573  6.976  91.0  2.1675  1.0  273.0
504  0.10959   0.0  11.93   0.0  0.573  6.794  89.3  2.3889  1.0  273.0
505  0.04741   0.0  11.93   0.0  0.573  6.030  80.8  2.5050  1.0  273.0

     PTRATIO       B  LSTAT
0       15.3  396.90   4.98
1       17.8  396.90   9.14
2       17.8  392.83   4.03
3       18.7  394.63   2.94
4       18.7  396.90   5.33
..       ...     ...    ...
501     21.0  391.99   9.67
502     21.0  396.90   9.08
503     21.0  396.90   5.64
504     21.0  393.45   6.48
505     21.0  396.90   7.88

[506 rows x 13 columns]
0      24.0
1      21.6
2      34.7
3      33.4
4      36.2
       ...
501    22.4
502    20.6
503    23.9
504    22.0
505    11.9
Name: price, Length: 506, dtype: float64
```

Here the varible x contains the value of the first 13 columns i.e the parameters that are required for calculating and predicting the house prices. The varible y contains the 14th column values which are the house prices.

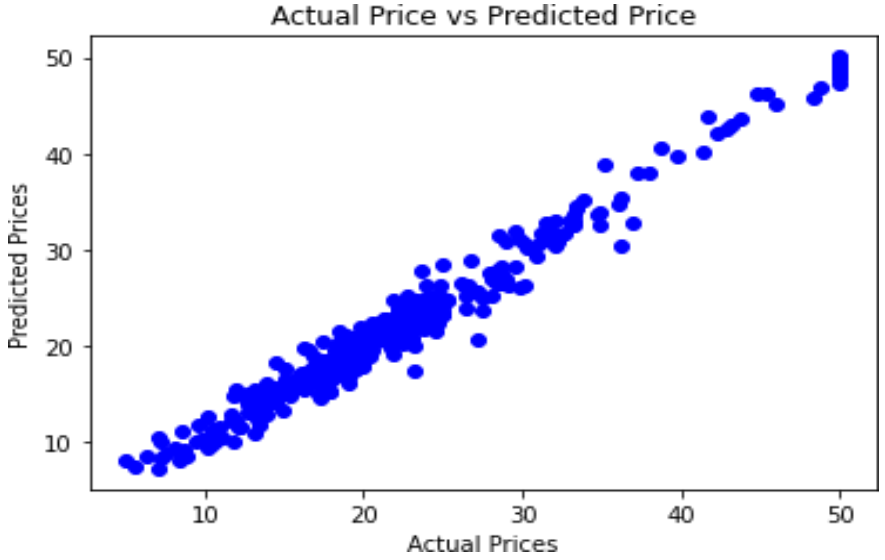
First we predict the values in y using the values in x . Then we compare the actual prices and predicted prices by using scatter plot. Then we find the r square error and mean square error between them . If the errors is less enough then we proceed for testing of the model since the training phase is over. If the error is large , thenwe use optimizers like adam, and repeat drop and fitting process for a set number of epochs to reduce the error.

The r square error or mean square error for good accuracy of the model in predicting the data is indicatednumerically also.

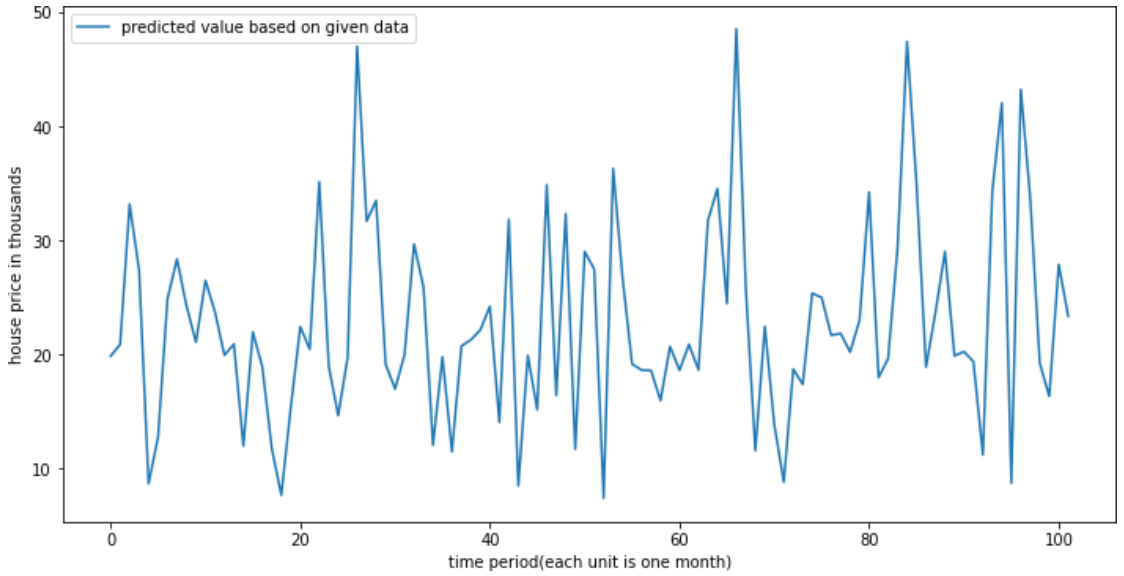
A model is good if these error values are less then 5.

Then during testing process we predict the future house prices using present and past data parameters ofhouses in an location. Then we plot this graphically as a house price over time graph.

For training the model , the error needs to be minimum for greater accuracy of model. The error between the actual and predicted price is plotted graphically using scatter plot. Here we can see that error is minimum sincethe data points of actual and predicted value are close to each other



PREDICTED VALUE OF HOUSE PRICE BASED ON TEST SAMPLE DATA



ALGORITHM BRIEF OUTLINE

1. Import the python libraries that are required for house price prediction using linear regression. Example: numpy is used for convention of data to 2d or 3d array format which is required for linearregression model ,matplotlib for plotting the graph , pandas for readingthe data from source and manipulation that data, etc.

2. First Get the value from source and give it to a data frame and thenmanipulate this data to required form using head(),indexing, drop().

3. Next we have to train a model, its always best to spilt the data intotraining data and test data for modelling.

4. Its always good to use shape() to avoid null spaces which will cause error during modelling process.

5. Its good to normalize the value since the values are in very large quantity for house prices , for this we may use minmaxscaler to reducethe gap between prices so that its easy and less time consuming for comparing and values.range usually specified is between 0 to 1 using fittransform.

6. Then we have to make few imports from keras: like sequential for initializing the network,lstm to add lstm layer, dropout to prevent overfitting of lstm layers, dense to add a densely connected networklayer for output unit.

7. In lstm layer declaration its best to declare the unit, activiation,returnsequence.

8. To compile this model its always best to use adam optimizer and set the loss as required for the specific data.

9. We can fit the model to run for a number of epochs. Epochs are the number of times the learning algorithm will work through the entire training set.

10. Then we convert the values back to normal form by using inverse minimal scale by scale factor.
11. Then we give a test data(present data)to the trained model to get the predicted value(future data).

12. Then we can use matplotlib to plot a graph comparing the test andpredicted value to see the increase/decrease rate of values in each time of the year in a particular place. Based on this people will know when its best time to sell or buy a place in a given location.