IBM NAAN MUDHALVAN

Artificial Intelligence Group – 3

HOUSE PRICE PREDICTION USING MACHINE LEARNING

Team members:

Name : Sanjai.J

Reg no: 721921243090

A Project Report on

HOUSE PRICE PREDICTION USING MACHINE LEARNING

Phase 1: Problem Definition and Design Thinking

In this part you will need to understand the problem statement and create a document on what have you understood and how will you proceed ahead with solving the problem. Please think on a design and present in form of a document.

Problem Definition: The problem is to predict house prices using machine learning techniques. The objective is to develop a model that accurately predicts the prices of houses based on a set of features such as location, square footage, number of bedrooms and bathrooms, and other relevant factors. This project involves data preprocessing, feature engineering, model selection, training, and evaluation.

Design Thinking:

1. Data Source: Choose a dataset containing information about houses, including features like location, square footage, bedrooms, bathrooms, and price.

1. Data Preprocessing: Clean and preprocess the data, handle missing values, and convert categorical features into numerical representations.

1. Feature Selection: Select the most relevant features for predicting house prices.

1. Model Selection: Choose a suitable regression algorithm (e.g., Linear Regression, Random Forest Regressor) for predicting house prices.

1. Model Training: Train the selected model using the preprocessed data.

1. Evaluation: Evaluate the model's performance using metrics like Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R-squared.

Dataset Link: https://www.kaggle.com/datasets/vedavyasv/usa-housing

```
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_factor y 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)))
```

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 linear regression model ,matplotlib for plotting the graph , pandas for reading the data from source and manipulation that data, etc.

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

CONCLUSION

Thus the machine learning model to predict the house price based on given dataset is executed successfully using xg regressor (a upgraded/ slighted boosted form of regular linear regression, this gives lesser error). This model further helps people understand whether this place is more suited for them based on heatmap correlation. It also helps people looking to sell a house at best time for greater profit. Any house price in any location can be predicted with minimum error by giving appropriate dataset.