PREDICTING HOUSE PRICES USING MACHINE LEARNING

IBM GROUP-1

ARTIFICIAL INTELLIGENCE-PHASE 4

Department of ECE:
Sai Preethi K P (113321106078)
Papita Biswas (113321106066)
S Sharada (113321106096)
Sandhiya B (113321106080)
Srinidhi M (113321106093)

APPLICATION AND ALGORITHM OF PREDICTING HOUSE PRICES USING MACHINE LEARNING

APPLICATION:

1. *Data Collection*: Gather a dataset with features such as square footage, number of bedrooms, location, and other relevant information. Historical sale prices should be included.

2. *Data Preprocessing*: Clean the data by handling missing values and outliers, and convert categorical data into numerical formats using techniques like one-hot encoding.

3. *Feature Selection/Engineering*: Choose relevant features or create new ones that might improve prediction accuracy. For instance, you can calculate the price per square foot.

4. *Split Data*: Divide the data into training and testing sets to evaluate your model's performance.

5. *Model Selection*: Choose a regression algorithm, such as Linear Regression, Decision Trees, Random Forest, or Gradient Boosting, based on your data and task requirements.

6. *Model Training*: Train the selected model on the training data.

7. *Model Evaluation*: Evaluate the model's performance using metrics like Mean Absolute Error (MAE), Mean Squared Error (MSE), or Root Mean Squared Error (RMSE).Adjust hyperparameters to improve results.

8. *Cross-Validation*: Perform cross-validation to assess the model's generalization ability.

9. *Predict House Prices*: Use the trained model to predict house prices for new data.

10. *Model Deployment*: Deploy the model, typically as part of a web application or service where users can input property details to get price predictions.

11. *Regular Updates*: Periodically update the model with new data to maintain accuracy.

ALGORITHM:

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 line arregression 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 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 mill 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 reduce the gap between prices so that its easy and less time consuming for comparing and values range usually specified is between 0 to 1 using fit transform.

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 network layer for output unit.

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

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.