

TASK NO:1

Introduction

This project aims to predict house prices based on various features using machine learning techniques. The dataset is preprocessed, important features are extracted, and a machine learning model is trained to provide accurate price predictions.

Code Explanation

Initialization:

- Required libraries are imported, including pandas, numpy, matplotlib, seaborn, and scikit-learn.
- The dataset is loaded using pandas and examined with `head()`, `tail()`, and `describe()` functions.
- The shape of the dataset (number of rows and columns) is displayed.
- All attributes (columns) are printed to understand available features.

Handling Missing Values:

- Identifies missing values using `isnull().sum()` and sorts them in descending order.
- Missing numerical values are filled with the mean of their respective columns.
- Data types of certain columns are converted from float to integer.

Data Preprocessing:

- Categorical variables are encoded into numerical values.
- Feature scaling is applied using `StandardScaler` to normalize numerical features.
- The dataset is split into training and testing sets using `train_test_split()`.

Model Training:

- A Random Forest Regressor is initialized and trained on the dataset.
- The model learns patterns from the training data to predict house prices.

Model Evaluation:

- The model's performance is evaluated using mean squared error (MSE).
- The lower the MSE, the better the model's prediction capability.

Predictions:

- The trained model is used to predict house prices on the test data.
- Predicted values are compared with actual prices for accuracy assessment.

Visualization:

- Data is visualized using seaborn and matplotlib to understand feature relationships.
- Plots such as scatter plots, histograms, and boxplots are used for analysis.

Conclusion:

This project successfully implements a house price prediction model using machine learning. Data preprocessing, feature selection, and model training steps ensure accurate price predictions. Further improvements can be made by fine-tuning hyperparameters and exploring advanced ML models.