PREDICTING HOUSE PRICE USING MACHINE LEARNING

PHASE-2

INNOVATION

INTRODUCTION:

Predicting house prices using machine learning is a common and practical application of data science in the real estate industry. It involves using historical data and various machine learning algorithms to create models that can predict the selling price of houses based on their features. This application is valuable for homebuyers, sellers, and real estate professionals, as it provides insights into property values, market trends, and investment decisions.

GRADIENT BOOSTING:

Gradient Boosting is a machine learning technique for regression and classification problems, which builds a predictive model in the form of an ensemble of weak learners, typically decision trees. It is an extension of boosting, a machine learning ensemble method that combines the predictions of several base estimators (often decision trees) to improve accuracy and robustness over a single estimator.

The "gradient" in Gradient Boosting refers to the use of gradient descent optimization technique to minimize the loss function, which measures the difference between the predicted values and the actual target values.

XGBOOST:

XGBoost, short for eXtreme Gradient Boosting, is an optimized and scalable machine learning algorithm designed to efficiently handle large datasets and solve various supervised learning problems, including regression, classification, ranking, and user-defined prediction tasks**.**

STEPS INVOLVED IN PHASE 2

STEP 1: Understanding the Problem

- Define the objective: Predict house prices based on various features.

- Understand the context: Familiarize yourself with the real estate market and factors influencing house prices.

STEP 2 : Gathering Data

- Collect relevant data: Gather a dataset containing historical house prices and relevant features.

STEP 3 : Data Cleaning

- Handle missing values: Fill missing values using mean, median, or advanced imputation techniques.
- Outlier detection and removal: Identify outliers in numerical features and consider removing or transforming them.
- Data type conversion: Ensure data types of features are appropriate (e.g., convert categorical variables to numerical using encoding methods).
- Remove duplicates: Check for and remove duplicate records if present.

STEP 4 : Data Exploration and Analysis

- Load the data into a pandas DataFrame.
- Explore the data: Analyze summary statistics, check for data distributions, and identify patterns.
- Visualize the data: Utilize histograms, box plots, and correlation matrices to understand feature relationships.
- Explore target variable: Analyze the distribution of house prices and check for skewness.

STEP 5 : Data Splitting

- Split the dataset into training and testing sets. Common split ratios are 80/20 or 70/30, with the larger portion allocated to training.

STEP 6 : Choosing a Model

- Select a regression algorithm suitable for the problem. Options include Linear Regression, Decision Trees, Random Forest, Gradient Boosting, or XGBoost.

- Consider ensemble methods for improved accuracy and generalization.

STEP 7 : Training the Model

- Train the chosen model using the training data.
- Fine-tune hyperparameters using techniques like cross-validation or grid search.

STEP 8 : Model Evaluation

- Evaluate the model's performance using appropriate metrics (Mean Squared Error, R-squared, etc.) on the test dataset.
- Visualize predictions vs. actual values for a qualitative assessment.

STEP 9 : Model Optimization

- If the model performance is not satisfactory, consider feature selection, feature engineering, or trying different algorithms.
- Optimize hyperparameters further to achieve better results.

STEP 10 : Prediction and Deployment

- Use the trained model to make predictions on new data (features of houses for which you want to predict the price).
- Deploy the model for real-time predictions using appropriate tools or frameworks.

CODE LINK :

https://colab.research.google.com/drive/1oC1UUZweoDr0i-_XVixeuGm4xHXqVl4R#scrollTo=Q7vB3W69_8Qr

CONCLUSION :

Machine learning techniques, such as regression models and ensemble methods like Gradient Boosting and XGBoost, offer powerful tools for making accurate predictions. However, the effectiveness of the prediction models heavily depends on the quality of the data and the features selected for analysis.