HOUSE PRICE PREDICTOR

SYNOPSIS

TITLE – HOUSE PRICE PREDICTOR

KUSHAL SG – AF0378117

ABOUT THE TOPIC (DATASET):

A house price predictor is a machine learning model designed to estimate the value of a property based on various features such as location, size, number of bedrooms, age, and other relevant attributes. The process begins with data collection, where information on key features affecting house prices is gathered. This data is then preprocessed to handle missing values, remove duplicates, normalize numerical features, and encode categorical variables. Next, appropriate machine learning algorithms like Linear Regression, Decision Trees, Random Forests, Gradient Boosting Machines, or Neural Networks are selected for modeling. The dataset is split into training and testing sets, with the model trained on the training set and validated using techniques like cross-validation to fine-tune hyperparameters and prevent overfitting. Finally, the model's performance is evaluated using metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), or R-squared to ensure accurate and reliable house price predictions.

Methodology for House Price Prediction

1. Data Collection:
   - Data Sources: Gather data from reliable sources such as real estate websites, government databases, or property listings.
   - Features: Collect relevant features like location (city, neighborhood), size (square footage, lot size), number of rooms (bedrooms, bathrooms), age of the property, and additional amenities (garage, pool, garden).

2. Data Preprocessing:
   - Cleaning: Handle missing values through imputation, remove duplicates, and correct any inconsistencies in the data.
   - Normalization: Scale numerical features to a standard range to ensure uniformity and improve model performance.
   - Encoding: Convert categorical variables (e.g., neighborhood, type of house) into numerical format using one-hot encoding or label encoding.

3. Exploratory Data Analysis (EDA

- Visualization: Use plots and graphs to visualize the distribution of features and their relationships with house prices.
- Correlation Analysis: Identify which features are most strongly correlated with house prices to inform feature selection.

4. Feature Engineering:

- Feature Selection: Choose the most relevant features based on EDA and domain knowledge.
- New Features: Create new features if necessary, such as price per square foot or proximity to amenities.

5. Model Selection:

- Algorithms: Choose suitable machine learning algorithms, such as:
  - Linear Regression
  - Decision Trees
  - Random Forests
  - Gradient Boosting Machines
  - Neural Networks

6. Model Training:

- Train-Test Split: Split the dataset into training and testing sets (e.g., 80-20 or 70-30 split).
- Training: Train the model on the training set using selected algorithms.
- Hyperparameter Tuning: Use techniques like grid search or randomized search with cross-validation to optimize hyperparameters.

7. Model Evaluation:

- Metrics: Evaluate the model's performance using metrics such as:
  - Mean Absolute Error (MAE)
  - Mean Squared Error (MSE)
  - Root Mean Squared Error (RMSE)
  - R-squared (R^2)
- Validation: Use cross-validation to assess the model's generalizability and avoid overfitting.

8. Model Deployment:

- Prediction: Use the trained model to predict house prices on new, unseen data.
- API Integration: Deploy the model as an API for real-time predictions in web or mobile

applications.

- User Interface: Create a user-friendly interface to allow users to input property details and receive price estimates.

9. Continuous Improvement:

- Monitoring: Continuously monitor the model's performance and update it with new data to maintain accuracy.
- Retraining: Periodically retrain the model as market conditions and factors affecting house prices evolve.

10. Documentation and Reporting:

- Documentation: Document the entire process, from data collection to model deployment, for reproducibility and transparency.
- Reporting: Create reports and dashboards to present model performance and insights to stakeholders..

Technologies:

Pandas, matplotlib, Microsoft, PowerBi, numpy, scikit-learn.

Software Requirements:

Operating System – Windows, Linux and mac

IDLE – Jupyter Notebook

Hardware Requirements:

RAM – Minimum 6GB

Processor – Minimum intel i5

THANK YOU