House Value Prediction using Machine Learning

Project Overview

This project aims to predict house values using machine learning techniques. The dataset used for this project contains housing-related features from California. We apply multiple machine learning models to analyze and predict median house values based on various factors such as median income, house age, average rooms, population, and location.

Dataset

The dataset used for this project is `california_housing.csv`, which contains the following columns:

- `MedInc`: Median Income of households
- `HouseAge`: Median age of houses
- `AveRooms`: Average number of rooms per household
- `AveBedrms`: Average number of bedrooms per household
- `Population`: Total population of the area
- `AveOccup`: Average household occupancy
- `Latitude`: Geographic latitude
- `Longitude`: Geographic longitude
- `MedHouseVal`: Median house value (Target variable)

Data Preprocessing

1. **Loading Data:**
    - The dataset is loaded using pandas.
2. **Feature and Target Separation:**
    - `X` (features) consists of all columns except `MedHouseVal`.
    - `y` (target variable) consists of `MedHouseVal`.
3. **Data Splitting:**
    - The dataset is split into 80% training and 20% testing sets using `train_test_split` from `sklearn.model_selection`.

Model Building

Linear Regression

- A linear regression model is trained using `LinearRegression` from `sklearn.linear_model`.
- Predictions are made on training and testing sets.
- Model performance is evaluated using:
    - **Mean Squared Error (MSE)**
    - **R-squared Score (R^2)**

Random Forest Regressor

- A random forest model is built using `RandomForestRegressor` from `sklearn.ensemble`.
- The model is trained and evaluated similarly to linear regression.

Model Evaluation

The models are compared based on the following performance metrics:

- **Linear Regression:**
    - Training MSE: `0.5283`
    - Training R^2: `0.6021`
    - Testing MSE: `0.5089`
    - Testing R^2: `0.6223`
- **Random Forest:**
    - Training MSE: `0.7158`
    - Training R^2: `0.4608`
    - Testing MSE: `0.7068`
    - Testing R^2: `0.4754`

Data Visualization

- Scatter plots are used to visualize the relationship between actual and predicted values.
- A trend line is plotted to observe model fit.

Conclusion

- **Linear Regression** performs better than **Random Forest** in this case, achieving a higher R^2 score and lower MSE.
- Further improvements can be made by feature engineering, hyperparameter tuning, and trying different regression models.

Dependencies

Ensure you have the following Python libraries installed:

pip install pandas scikit-learn matplotlib numpy

Usage

1. Load the dataset.
2. Preprocess and split data.
3. Train and evaluate models.
4. Compare results and visualize predictions.