AI_PHASE4

Domain:Artificial Intelligence

Project 4:Predicting House Prices Using Machine Learning

INTRODUCTION:

The house price prediction project aims to create a predictive model that estimates house prices based on various property attributes. This phase involves fundamental tasks, including data preprocessing, model selection and training, and model evaluation.

DATASET:

https://www.kaggle.com/datasets/vedavyasv/usa-housing

OVERVIEW:

- This document presents a Python code implementation for predicting house prices based on a dataset containing attributes such as 'Avg. Area Income', 'Avg. Area House Age', 'Avg. Area Number of Rooms', 'Avg. Area Number of Bedrooms', 'Area Population', and the target variable 'Price'.
- The code utilizes Linear Regression and Random Forest models for prediction and evaluates their performance using Mean Squared Error (MSE) and R-squared metrics.

Steps involved:

1. **Data Loading and Overview**
2. **Data Preprocessing**
3. **Model Selection and Training**
4. **Model Evaluation**
5. **Results**

1.Date loading and overview:

```
# Load the dataset
data = pd.read_csv('house_data.csv')

# Display the first few rows to understand the structure of the data
print(data.head())
```

Explanation:

- Loading the dataset using Pandas.
- Printing the first few rows helps understand the columns and data types.

2. Data Preprocessing:

```
# Selecting Features and Target Variable
X = data[['Avg. Area Income', 'Avg. Area House Age', 'Avg. Area Number of Rooms', 'Avg. Area Number of Bedrooms', 'Area Population']]
y = data['Price']

# Splitting the dataset into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Feature Scaling
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)
```

- Identified the independent features and the dependent target variable.
- Split the dataset into training and testing sets for model evaluation.
- Scaled the features to have a standard mean and variance.

3. Model Selection and Training:

```
# Linear Regression model
linear_model = LinearRegression()
linear_model.fit(X_train, y_train)

# Random Forest model
forest_model = RandomForestRegressor(n_estimators=100, random_state=42)
forest_model.fit(X_train, y_train)
```

- Description of the chosen machine learning algorithm (e.g., Linear Regression, Random Forest, etc.).
- Training the model on the dataset.
- Hyperparameter tuning (if performed) and reasoning behind it.

4. Model Evaluation:

```
# Linear Regression predictions and evaluation
linear_predictions = linear_model.predict(X_test)
linear_mse = mean_squared_error(y_test, linear_predictions)
```

```
linear_r2 = r2_score(y_test, linear_predictions)

# Random Forest predictions and evaluation
forest_predictions = forest_model.predict(X_test)
forest_mse = mean_squared_error(y_test, forest_predictions)
forest_r2 = r2_score(y_test, forest_predictions)
```

- Made predictions using both models on the test set.
- Calculated Mean Squared Error (MSE) and R-squared to evaluate model performance.

5. Results:

```
print("Linear Regression Model:")
print(f"Mean Squared Error: {linear_mse}")
print(f"R-squared: {linear_r2}")
print("\nRandom Forest Model:")
print(f"Mean Squared Error: {forest_mse}")
print(f"R-squared: {forest_r2}")
```

- Displayed the performance metrics (MSE and R-squared) of both models.

FINAL OUTPUT:

```
Linear Regression Model:
```

Mean Squared Error: 10089009300.89399

R-squared: 0.9179971706834331

Random Forest Model:

Mean Squared Error: 14462012668.455772

R-squared: 0.882453675969917

- Displayed the performance metrics (MSE and R-squared) of both models.

CONCLUSION:

- In the final document, you can summarize the outcomes, compare model performances, and discuss possible improvements or next steps.
- Each section should include clear explanations to ensure the reader understands the purpose and workflow of the code.
- This expanded content provides a comprehensive understanding of the project's different phases. Adjust it based on the project's specific requirements and goals.


THANK YOU!