College code: 9133

Course: Artificial Intelligence

Phase 2: Innovation

Project title: House Price Predictor

TEAM MEMBERS:

maalinivcse2021@gmail.com *Maalini V*
divyadharshiniscse2021@gmail.com *DivyadharshiniS*
manoranjithampcse2021@gmail.com *ManoranjithamP*
pavithrancse2021@gmail.com *Pavithra N*
priyadharshiniscse2021@gmail.com *Priyadharshini S*

ENCRYPTION CODE:

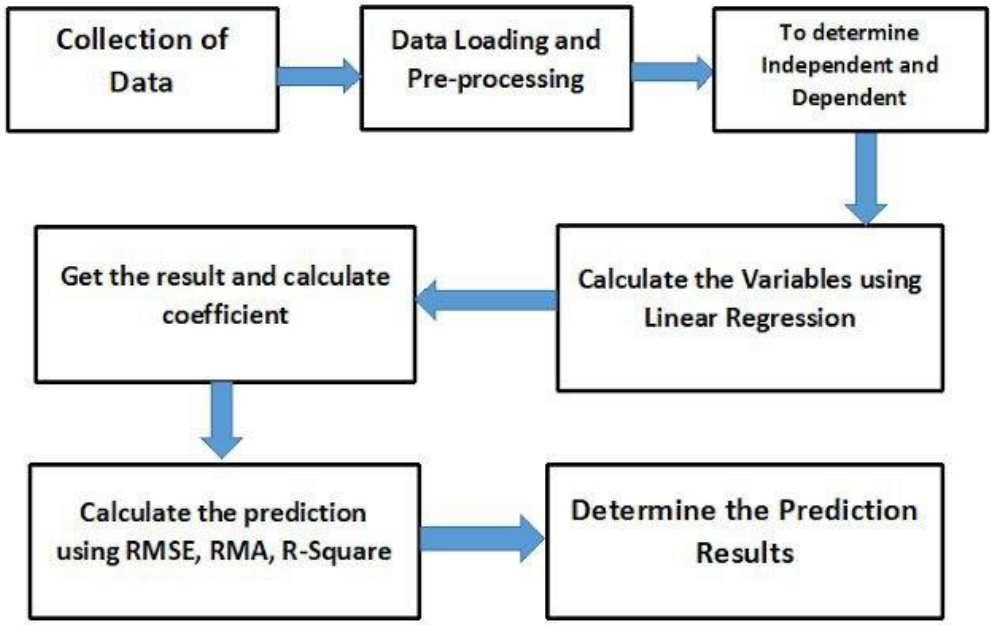
```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error
data = pd.read_csv('house_prices.csv')
X = data[['feature1', 'feature2', 'feature3']]
y = data['house_price']
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=42)
model = LinearRegression()
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
mse = mean_squared_error(y_test, y_pred)
```

```
print(f"Mean Squared Error: {mse}")
```

DECRYPTION CODE:

```
import pandas as pd
import joblib
model = joblib.load('house_price_model.pkl')
new_data = pd.read_csv('new_house_data.csv')
predictions = model.predict(new_data)
print(predictions)
```

FLOWCHART :



STEPS:

Data Collection:

Gather a dataset containing historical information on houses. This dataset should include features like square footage, number of bedrooms and bathrooms, location, and any other relevant information. It should also include the actual selling prices of these houses.

Data Preprocessing:

Clean the data by handling missing values and outliers.
Encode categorical features if necessary (e.g., converting locations into numerical values using one-hot encoding).
Split the dataset into training and testing sets to evaluate the model later.

Feature Selection/Engineering:
Analyze feature importance and select the most relevant features.
Create new features if they can provide valuable information (e.g., a "price per square foot" feature).

Model Selection:

Choose a regression algorithm. Linear Regression is a good starting point, but you can explore more complex models like Decision Trees, Random Forests, or Gradient Boosting as well.

Model Training:

Train the chosen machine learning model on the training dataset, where features (e.g., square footage, number of bedrooms) are used to predict the house prices.

Model Evaluation:

Use the trained model to make predictions on the testing dataset.
Evaluate the model's performance using metrics like Randomness Measuring Algorithm (RMA) or Root Mean Squared Error (RMSE) to measure how well it predicts house prices compared to the actual prices.

Hyperparameter Tuning:

Fine-tune the model's hyperparameters to optimize its performance.

Deployment:

If the model meets your accuracy requirements, you can deploy it to make real time predictions on new unseen data.