# AI-BASED HOUSE PRICE PREDICTION

# WITH MACHINE LEARNING

***Objective:***

House price prediction refers to the process of using machine learning techniques to estimate or forecast the selling price of residential properties, such as houses or apartments. This predictive modelling typically involves considering various property attributes and characteristics, such as size, location, number of bedrooms, amenities, and more, to make accurate predictions about what a property is likely to sell for in the real estate market. House price prediction models are valuable tools for homebuyers, sellers, real estate professionals, and investors, helping them make informed decisions in the housing market.

Problem Statement:

The problem statement for this project is to predict real estate prices based on various features such as income, house age, number of rooms, number of bedrooms, and population. The dataset used for this task is 'USA\_Housing.csv,' which likely contains historical data on real estate listings.

Phases of Development:

**1.Data Preparation:**

•Import necessary libraries like Pandas, NumPy, Seaborn, and scikit-learn for data processing, visualization, and modeling.

•Load the dataset from '**USA\_Housing.csv'** and examine its structure using dataset.info().

•Visualize the data using Seaborn to gain insights, such as histograms, box plots, pair plots, and correlation heatmaps.

•Preprocess the data by selecting features (X) and the target variable (Y). Then, split the data into training and testing sets using train\_test\_split. Perform feature scaling using StandardScaler.

**2.Linear Regression Model:**

•Initialize a Linear Regression model using LinearRegression().

•Train the model on the training data with scaled features.

•Make predictions on the test data and visualize the actual vs. predicted results.

•Evaluate the model using metrics like R-squared (r2\_score), Mean Absolute Error (MAE), and Mean Squared Error (MSE).

**3.Support Vector Regression (SVR):**

•Initialize an SVR model using SVR().

•Train the SVR model on the scaled training data.

•Make predictions on the test data and visualize the results.

•Evaluate the SVR model using the same metrics as in the Linear Regression model.

**4.Lasso Regression:**

•Initialize a Lasso Regression model with a specified alpha value.

•Train the Lasso model, make predictions, and visualize the results.

•Evaluate the Lasso model using the same metrics as above.

**5.Random Forest Regression:**

•Initialize a Random Forest Regression model with a specified number of estimators (trees).

•Train the Random Forest model, make predictions, and visualize the results.

•Evaluate the Random Forest model using the same metrics as above.

**6.XGBoost Regression:**

•Initialize an XGBoost Regression model using xg.XGBRegressor().

•Train the XGBoost model, make predictions, and visualize the results.

•Evaluate the XGBoost model using the same metrics as above.

**Dataset Description:**

The dataset used in this project is named 'USA\_Housing.csv.' While the code does not provide a specific description, we can make some assumptions about the dataset based on the column names used in the code. Here are some likely assumptions:

1.**'Price':** This is the target variable, representing the price of real estate properties.

2.**'Avg. Area Income':** This feature likely represents the average income of residents in the area where the property is located.

3.**'Avg. Area House Age':** This feature likely represents the average age of houses in the area.

4.**'Avg. Area Number of Rooms':** This feature likely represents the average number of rooms in houses in the area.

5.**'Avg. Area Number of Bedrooms':** This feature likely represents the average number of bedrooms in houses in the area.

6.**'Area Population':** This feature likely represents the population of the area where the property is located.

Data Preprocessing Steps:

**1.Data Loading:** The dataset is loaded using Pandas' read\_csv function from the file 'USA\_Housing.csv.'

**2.Data Exploration and Visualization:** Several Seaborn and Matplotlib functions are used to explore and visualize the data. This includes creating histograms, box plots, pair plots, and correlation heatmaps. These visualizations provide insights into the distribution and relationships between different features in the dataset.

**3.Feature Selection and Target Variable**:Features and the target variable are selected for the modeling phase. The features selected include '*Avg. Area Income'* , '*Avg. Area HouseAge'* ,'*Avg. Area Number of Rooms'* , *'Avg. Area Number of Bedrooms*' and '*Area Population*.' The target variable is '*Price.*'

**4.Data Splitting:** The data is split into training and testing sets using train\_test\_split from scikit-learn. The training set is used for model training, and the testing set is used to evaluate the models.

**5. Feature Scaling:** The selected features are scaled using StandardScaler from scikit-learn. This is important for ensuring that features are on a similar scale, which can improve the performance of some machine learning models.

Feature Extraction Techniques:

The code provided does not include specific feature extraction techniques. Instead, it focuses on feature selection and data preprocessing. Feature extraction typically involves creating new features from existing ones, but in this case, the code uses the original features as they are for modeling. The selection of relevant features and the standardization of these features are the key preprocessing steps in this project.

Choice of Machine Learning Algorithms:

**Linear Regression:** Linear regression is a common choice for regression tasks, such as predicting real estate prices, when there is a linear relationship between the features and the target variable.

**Support Vector Regression (SVR):** SVR is a regression technique that can capture non-linear relationships and is useful when the data does not have a clear linear pattern.

**Lasso Regression:** Lasso regression is used for feature selection and can help in situations where some features are not very informative or can be eliminated.

**Random Forest Regression:**  Random Forest is an ensemble method that combines multiple decision trees to improve predictive accuracy and can capture non-linear relationships in the data.

**XGBoost Regression:**  XGBoost is another ensemble method that is known for its strong predictive capabilities and is often used in regression tasks.

Model Training:

For each of the chosen machine learning algorithms, the code follows a common process for model training:

* Initialize the model (e.g., LinearRegression(), SVR(), Lasso(), RandomForestRegressor(), or XGBRegressor()).
* Fit the model to the training data using the fit method, where the model learns the relationships between the features and the target variable.

Evaluation Metrics:

The code uses several evaluation metrics to assess the performance of the models:

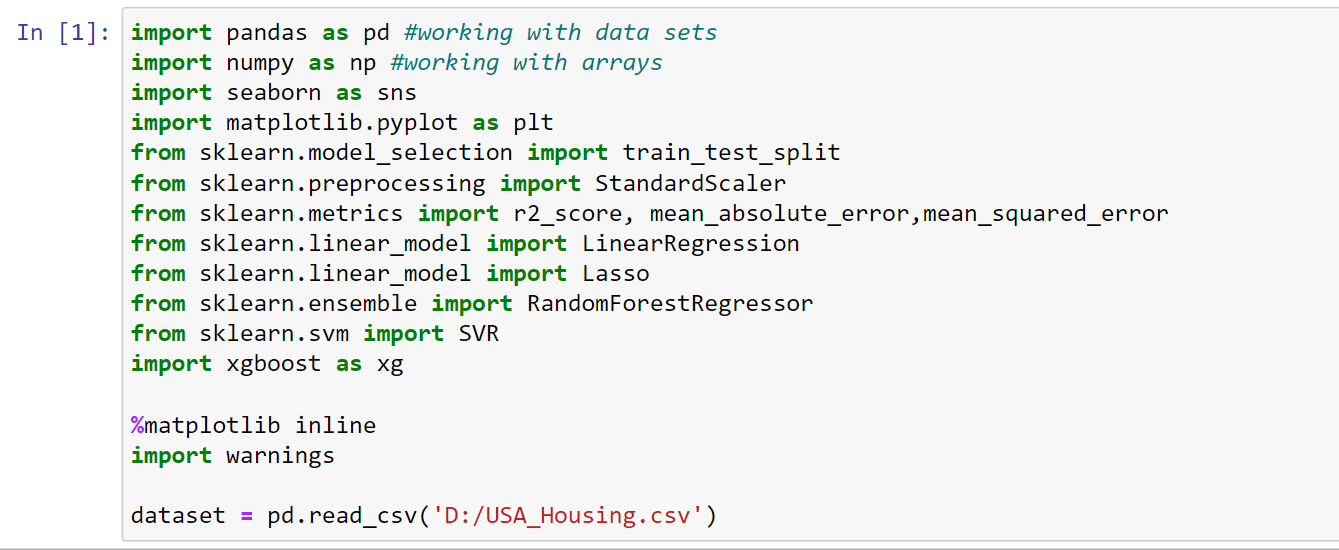
1**.R-squared (r2\_score):** R-squared measures how well the independent variables explain the variance in the dependent variable. A higher R-squared value indicates a better fit. It ranges from 0 to 1, with 1 being a perfect fit.

2.**Mean Absolute Error (MAE):** MAE measures the average absolute errors between the actual and predicted values. It provides an idea of the average prediction error.

3.**Mean Squared Error (MSE):** MSE measures the average of the squared differences between actual and predicted values. It penalizes larger errors more strongly than MAE.

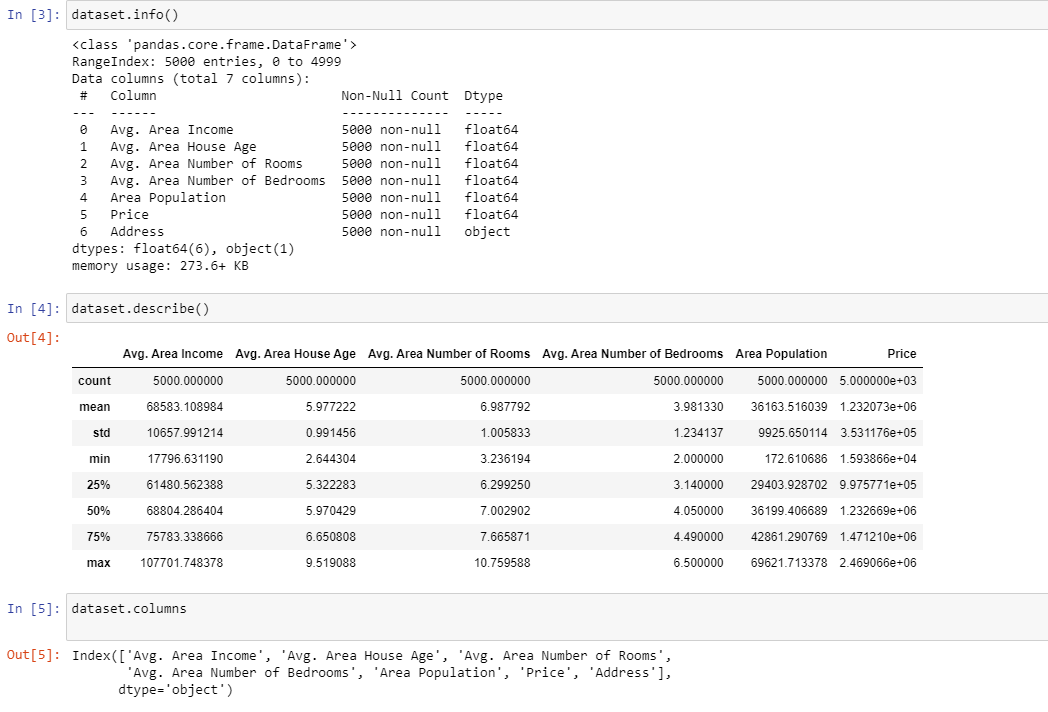
**Data Loading:**

Load the given dataset into our program. Use libraries like Pandas and Numpy for loading and mathematical operations in dataset.



**Data Exploration:**

Explore the dataset to understand its structure and features. This step helps to gain insights into the data and decide which features are relevant for our prediction model.

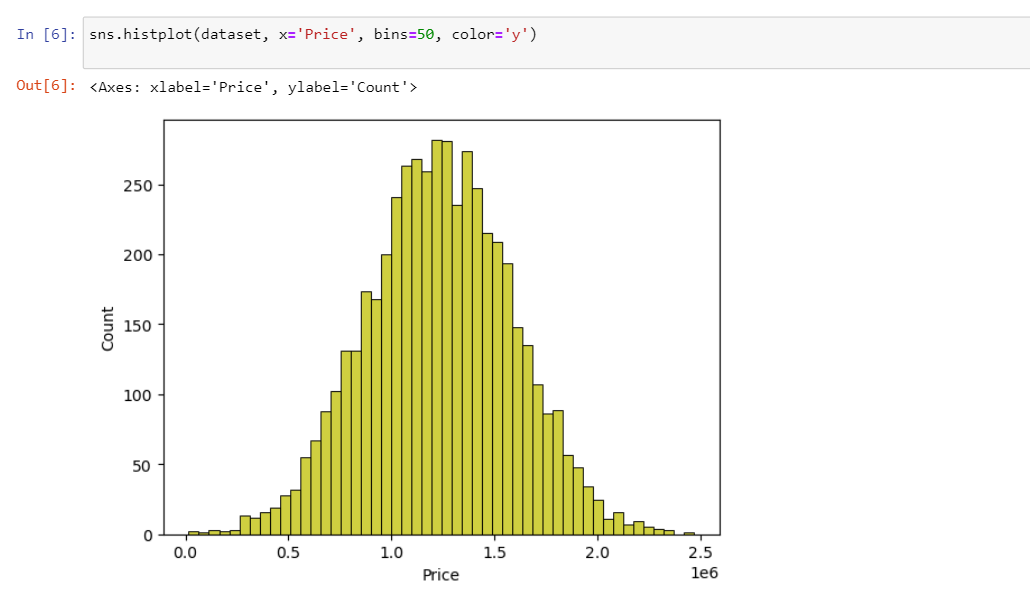


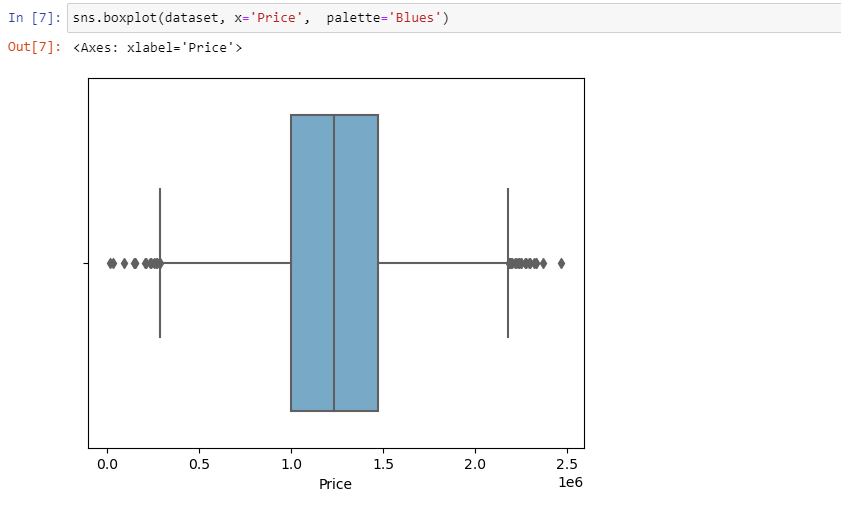
**Data preprocessing:**

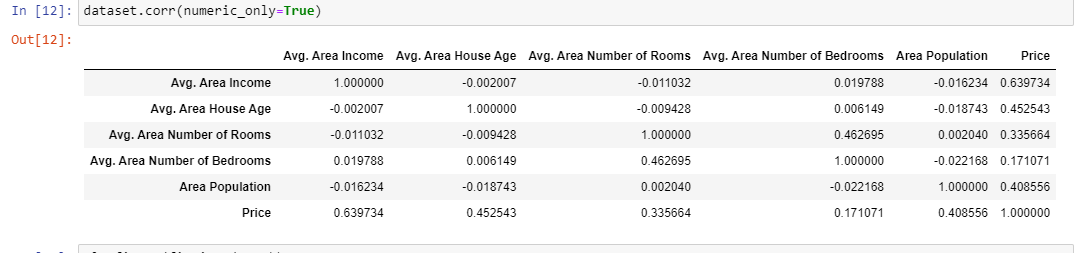
In this given dataset there is no non-null values or any missing values .So, we no need to preprocess it.

**Visualization:**

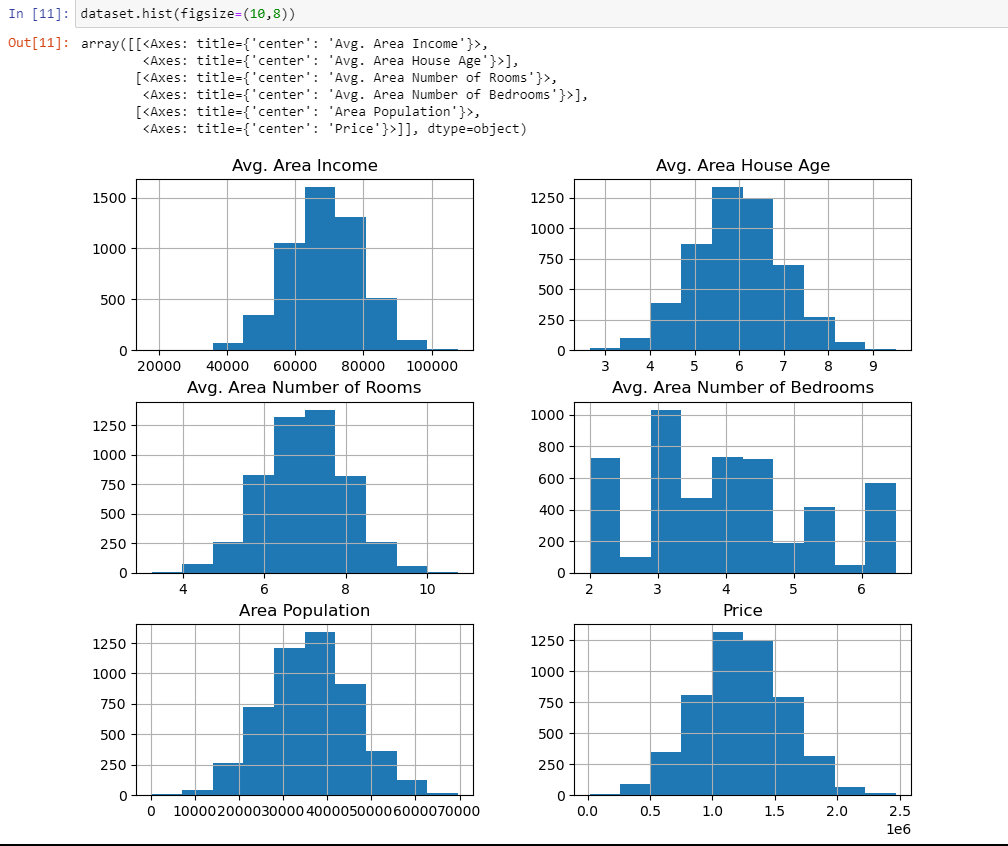
Visualization is the representation of data or information in a graphical or pictorial format. Here ,we used some visualization to make the predictions understandable to people. Here we used Hist plot that helps to visualize dataset distributions (Count VS Price)







Here Histogram is used to show frequency distributions. It uses array of data as parameters.



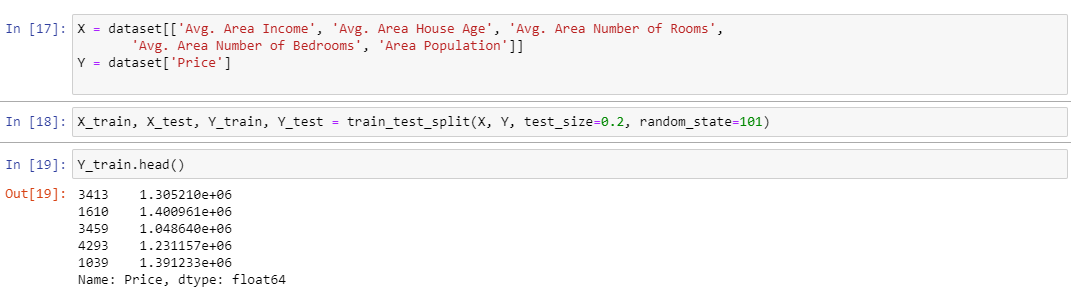
**TRAIN TEST SPLIT**

● The train-test split is a technique for evaluating the performance of a machine learning algorithm.

● Train Dataset: Used to fit the machine learning model.

● Test Dataset: Used to evaluate the fit machine learning model.

● Common split percentages include:



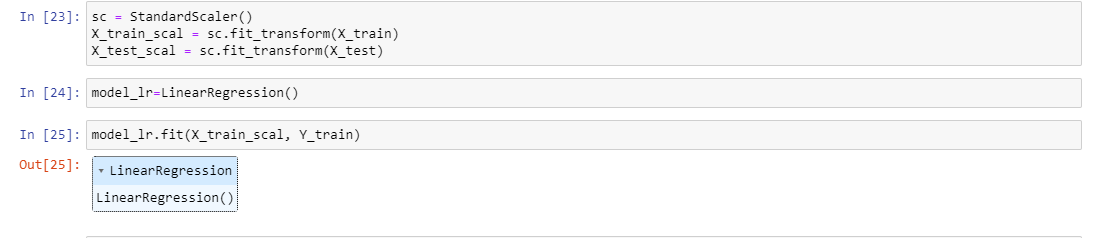


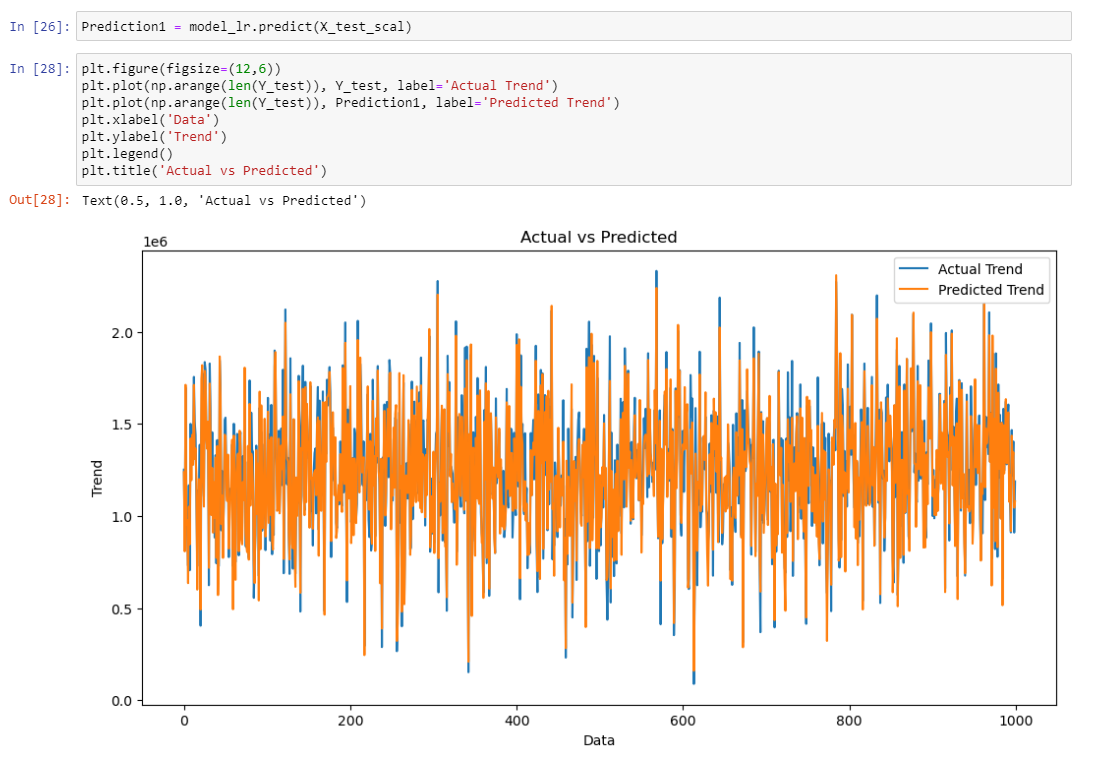


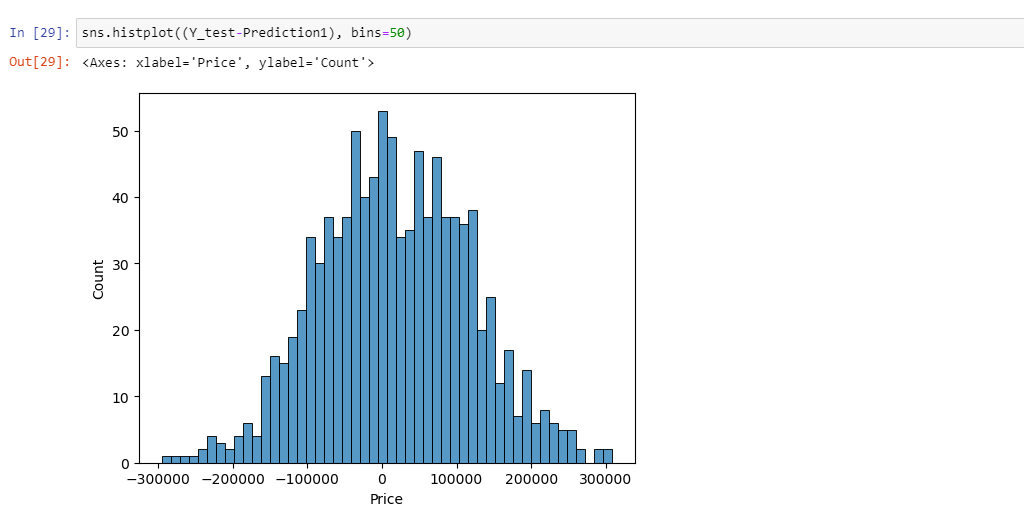


**LINEAR REGRESSION:**

Linear regression analysis is used to predict the value of a variable based on the value of another variable. The variable you want to predict is called the dependent variable. The variable you are using to predict the other variable's value is called the independent variable.







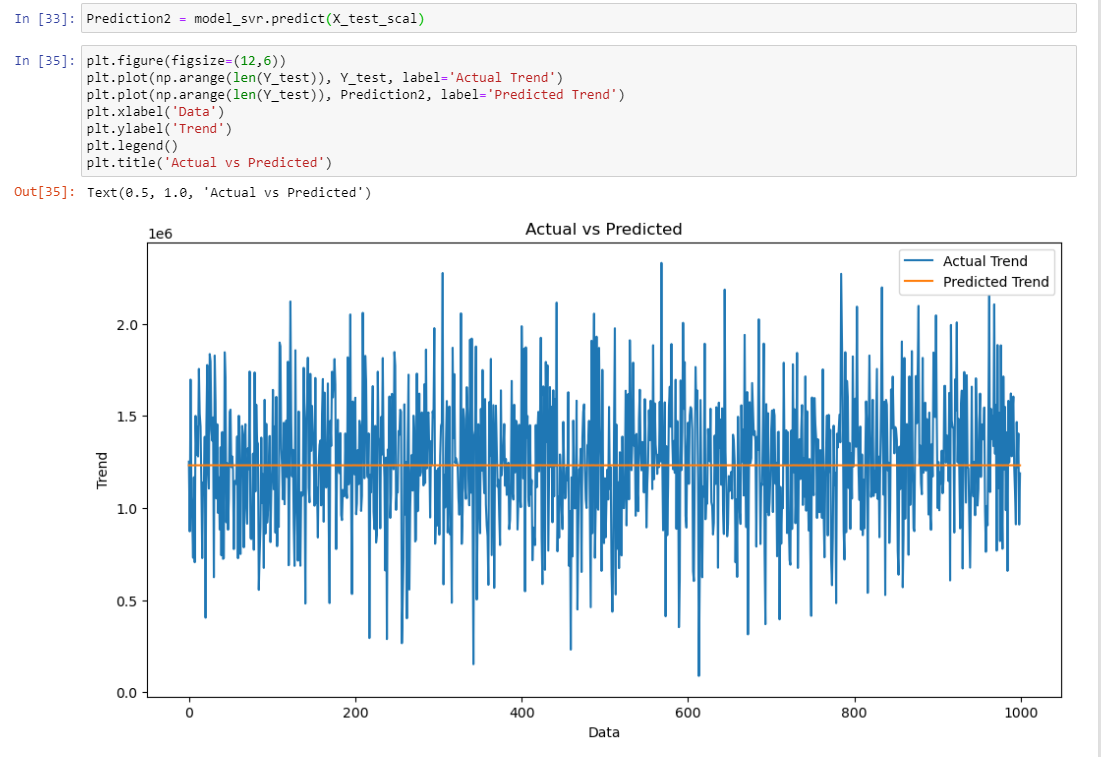
**EVALUATING ITS PERFORMANCE:**

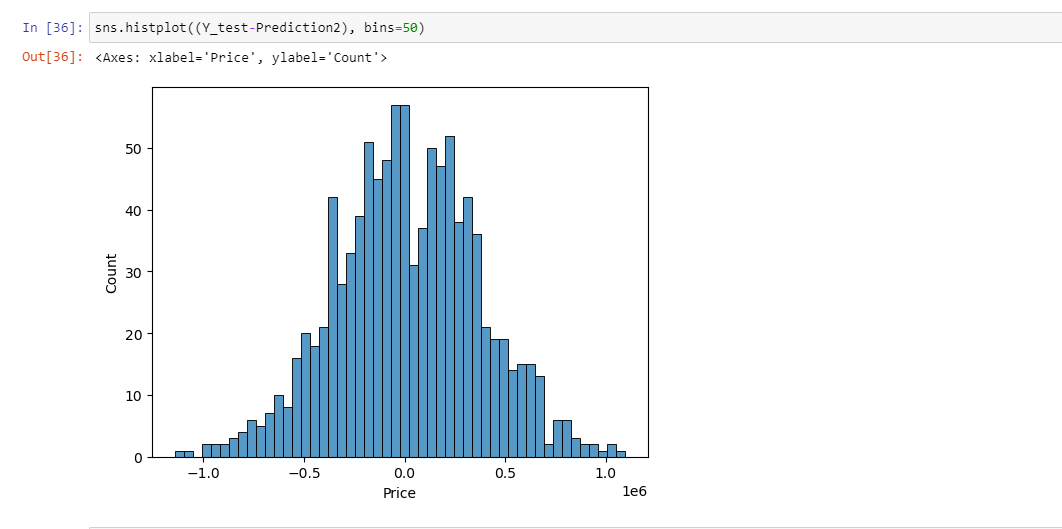


**SUPPORT VECTOR REGRESSION:**

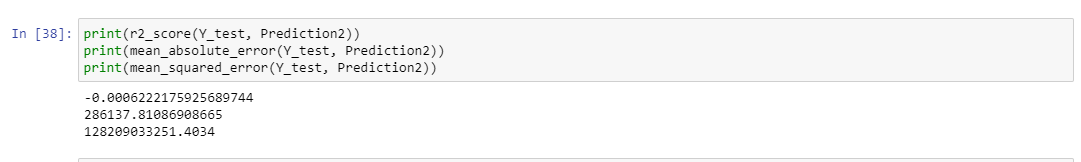
The goal of SVR is to find a function that approximates the relationship between the input variables and a continuous target variable, while minimizing the prediction error .







**EVALUATING ITS PERFORMANCE:**



**CONCLUSION:**

This project demonstrates fundamental practices for real estate price prediction, innovative techniques or approaches are not explicitly introduced. The code focuses on a solid foundation of data preprocessing, model selection, and evaluation. To achieve innovation, more advanced techniques like feature engineering, hyperparameter tuning, and cross-validation could be considered to improve model performance further.