**PRODUCT DEMAND PREDICTION**

**PHASE – 4**

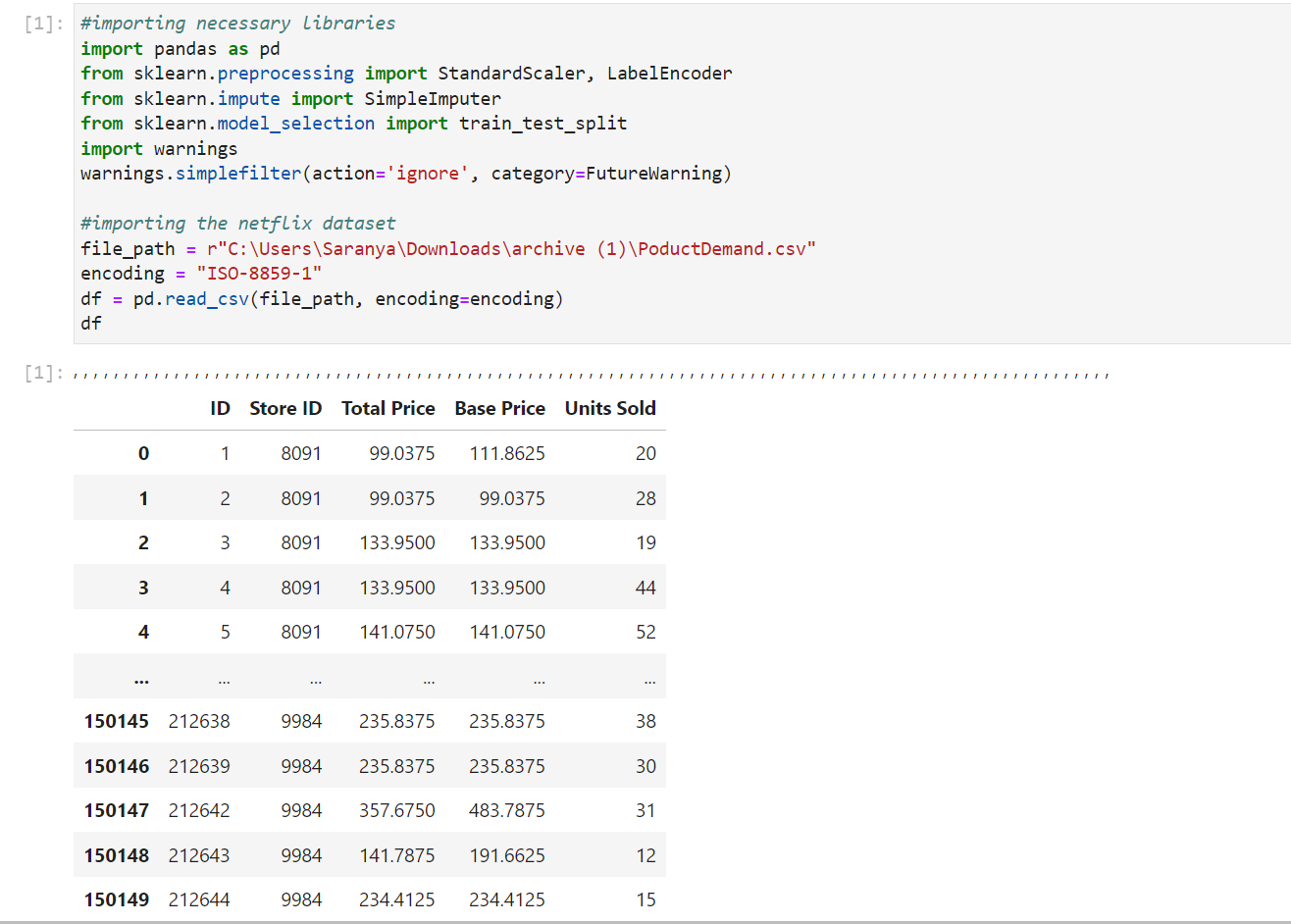
**Development Phase Part – 2**

**-**

(Team Member)

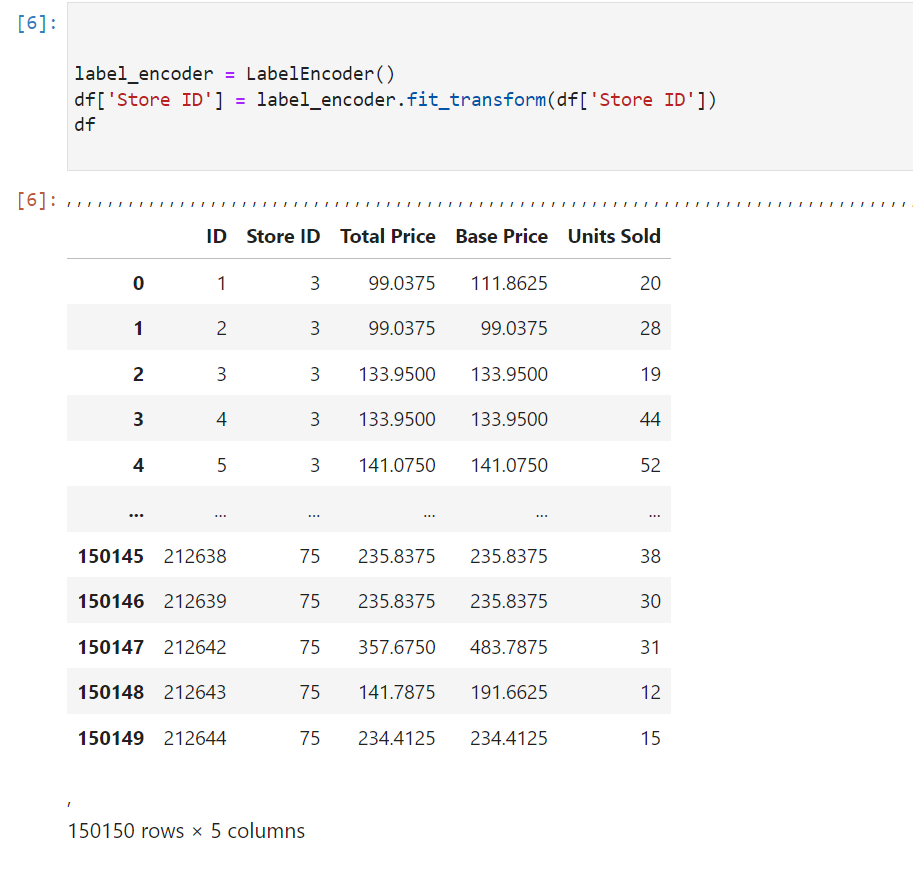
Explanation for Development part – 1

**Step 1:** Importing the required libraries and loading the dataset

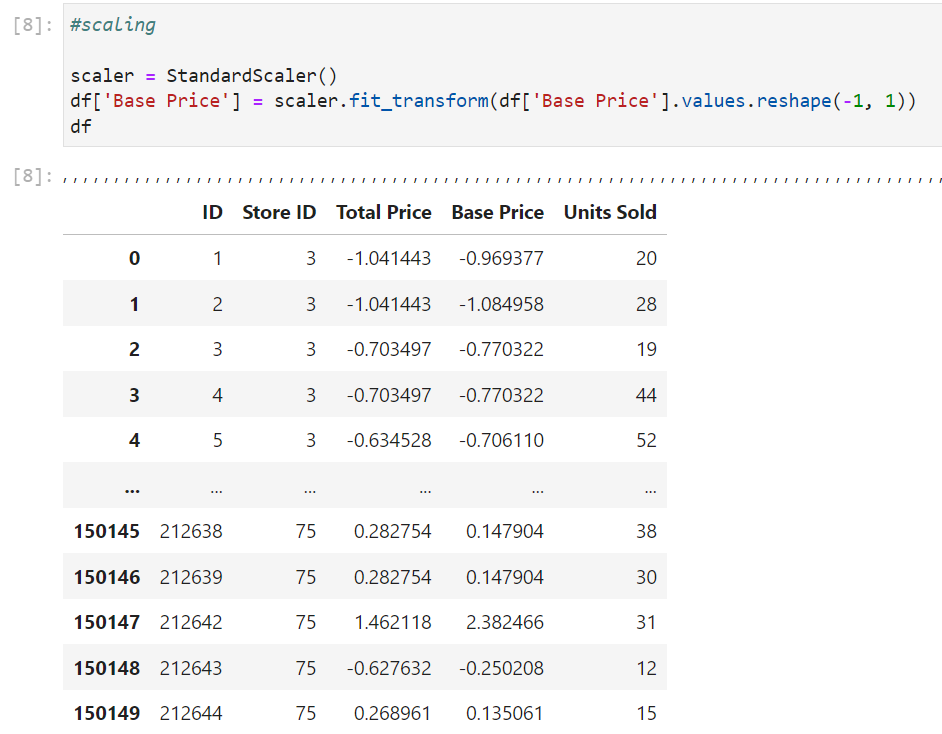


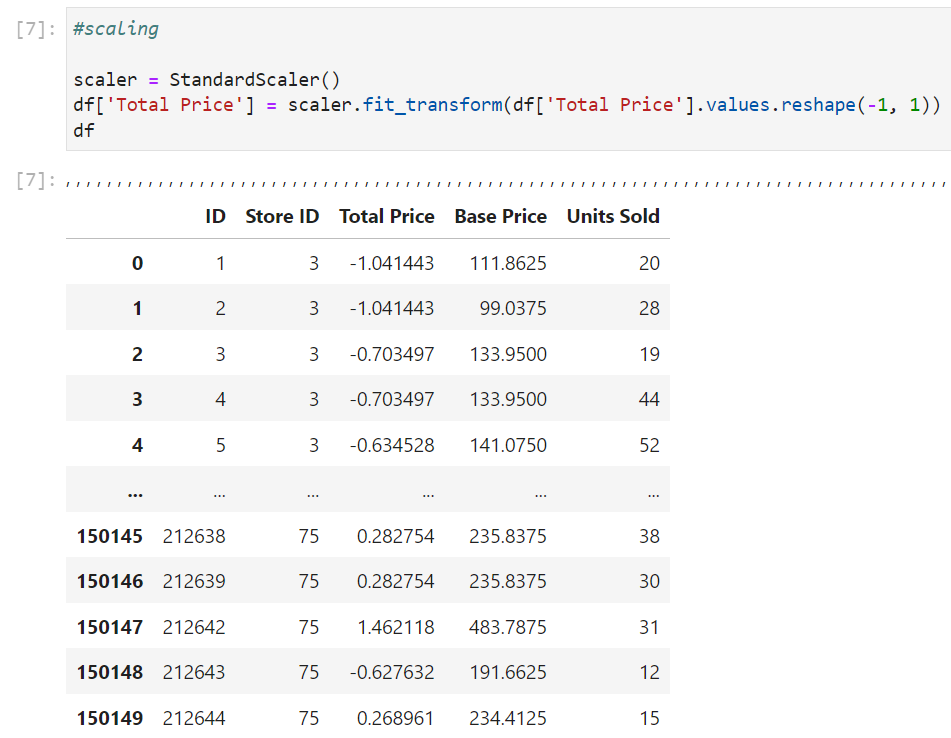
**Step 2**: Handling Missing Data



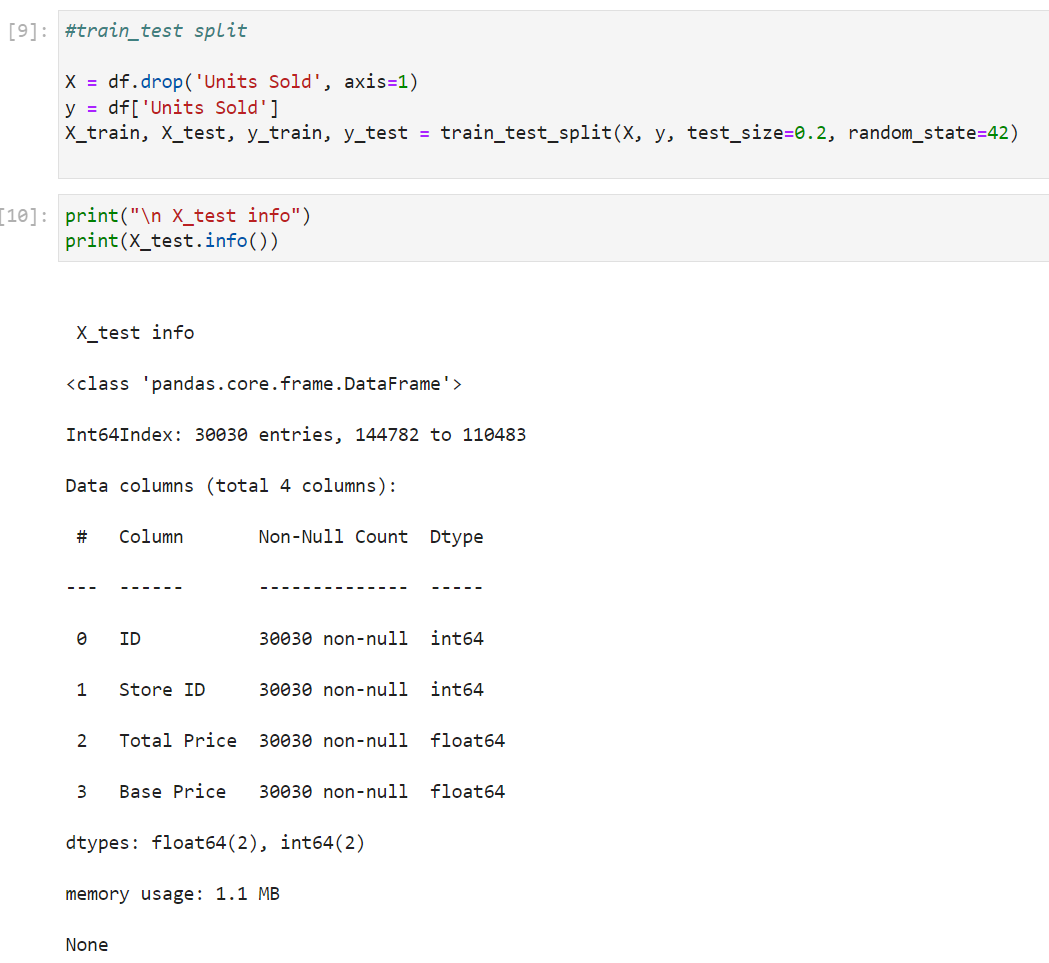
**Step 3:** Label encoder for Total Price Column 

**Step 4**: Feature Scaling using StandardScaler





**Step 5**: Splitting the data into a training set and a test



**Feature Scaling, Model Training, and Evaluation Algorithm for Product Demand Prediction**

**Objective:**

This algorithm aims to guide the development of a predictive model for product demand prediction using the provided dataset. It covers essential steps, including feature engineering, model training, and evaluation, to ensure accurate predictions.

**Steps:**

1. Load and Preprocess the Dataset:

* Load the dataset, which includes information on products such as ID, StoreID, Total Price, Base Price, Units Sold.
* Ensure that you understand the dataset's structure and contents.

2. Feature Engineering:

* Review the dataset to identify which features will be used for predicting the demand for products.
* Handle any missing data. It appears that the dataset does not have any missing values.
* Encode categorical data, using techniques like label encoding or one-hot encoding to convert them into a numerical format.

3. Feature Scaling

* Analyze the dataset and determine if feature scaling is required. Some machine learning algorithms benefit from scaled features.
* If needed, apply feature scaling to numerical features. For example, you can use standardization to scale the feature.

4. Split the Dataset:

* Split the dataset into training and testing sets to assess the model's performance.
* A common split ratio is 80% for training and 20% for testing. Ensure that the split is random to avoid any potential biases.

5. Select a Machine Learning Model:

* -Choose an appropriate machine learning model for regression tasks.

6. Train the Model:

* Initialize the chosen model.
* Fit the model to the training data, using the selected features as input and IMDb scores as the target variable.
* During training, the model will learn patterns in the data.

7. Make Predictions:

* Utilize the trained model to make IMDb score predictions on the testing data.
* The model predicts IMDb scores based on the test feature data.

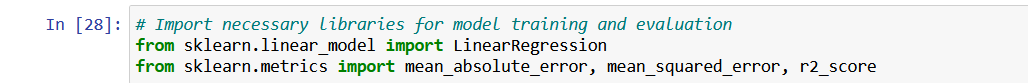
8. Evaluate the Model:

* + Assess the model's performance using regression evaluation metrics. Common metrics include:
  + Mean Absolute Error (MAE): Measures the average absolute difference between predicted and actual price.
  + Mean Squared Error (MSE): Measures the average of the squared differences between predicted and actual score
  + Root Mean Squared Error (RMSE): The square root of MSE, providing error in the original score units.
  + R-squared (R2): Measures the proportion of the variance in IMDb scores explained by the model.
  + Visualize the results, such as scatter plots comparing actual IMDb scores vs. predicted scores or distribution plots.

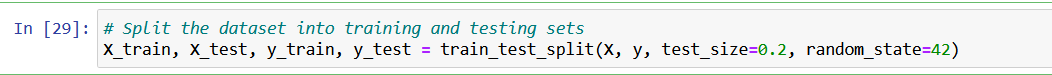
This algorithm provides a structured approach to developing a product demand prediction model specifically tailored to the dataset.

**Execution of the model:**

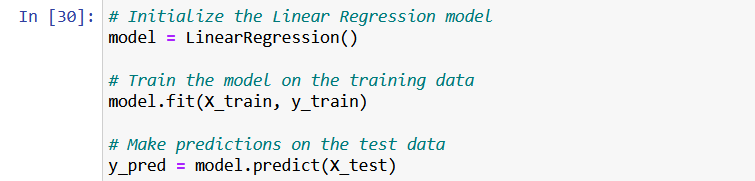
Importing the necessary libraries:



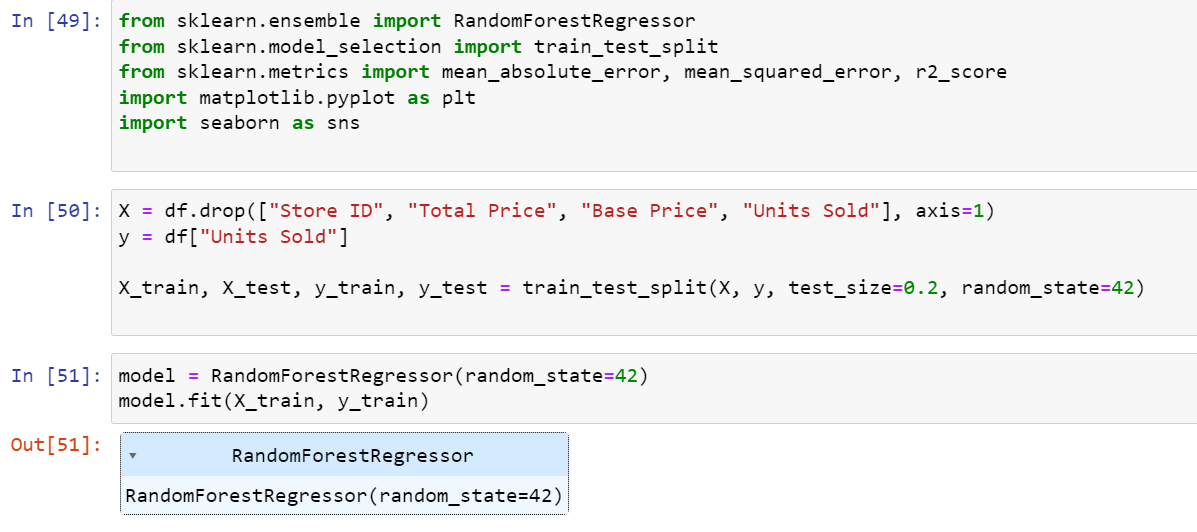
Train test split:



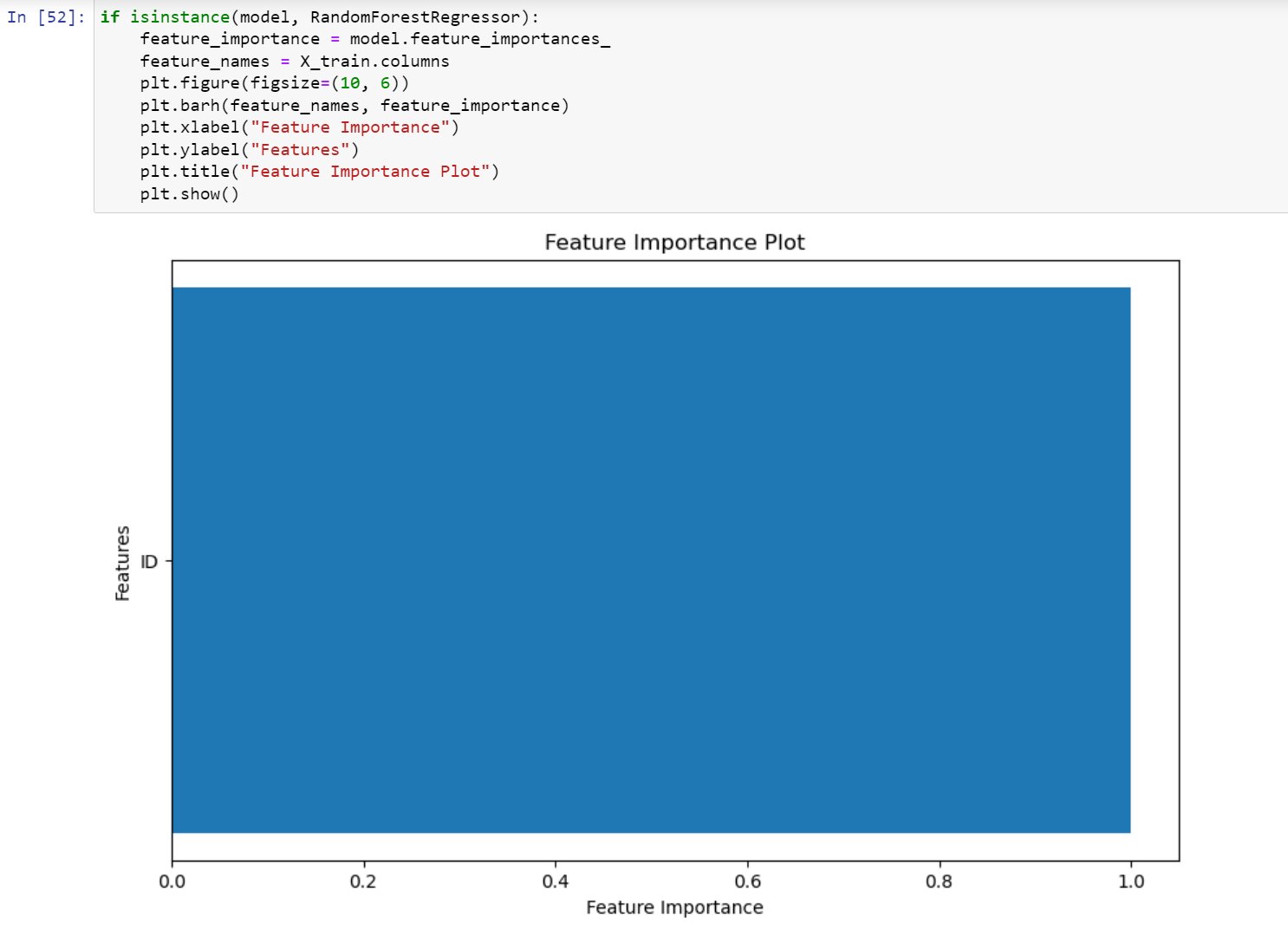
Linear Model:



Random Forest:

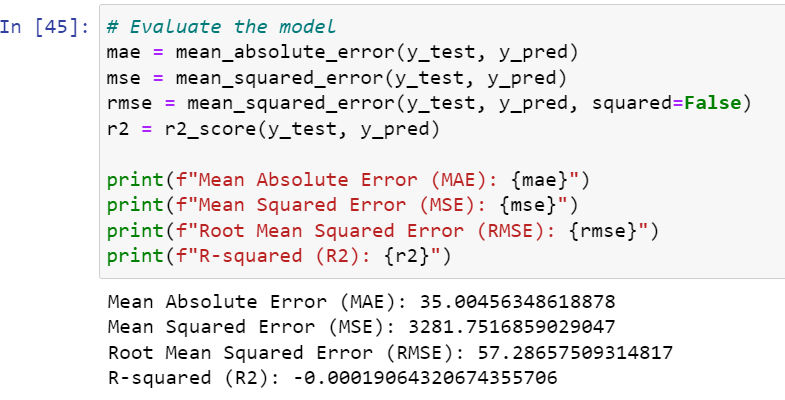


Feature importance plot for Random Forest



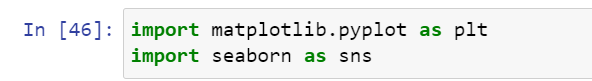
Evaluating the Model:

Using MAE, MSE, RMSE and R2

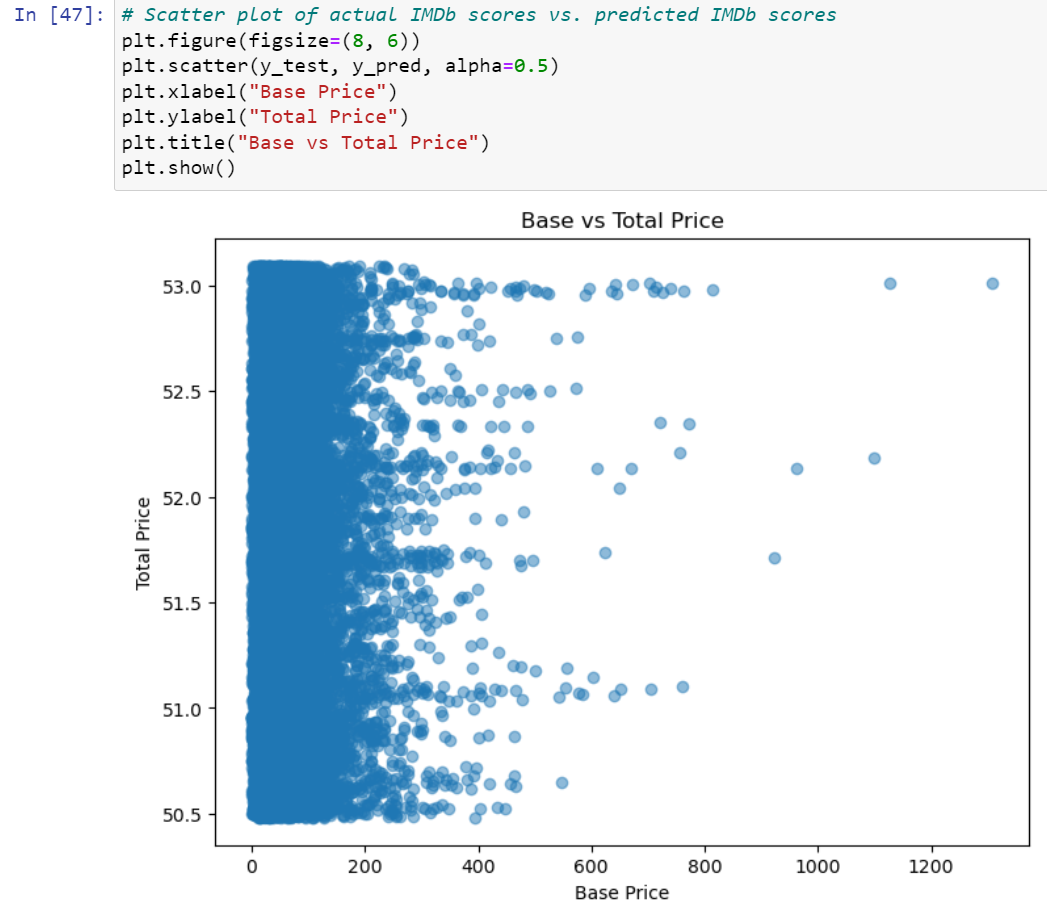


Visualization of the result:

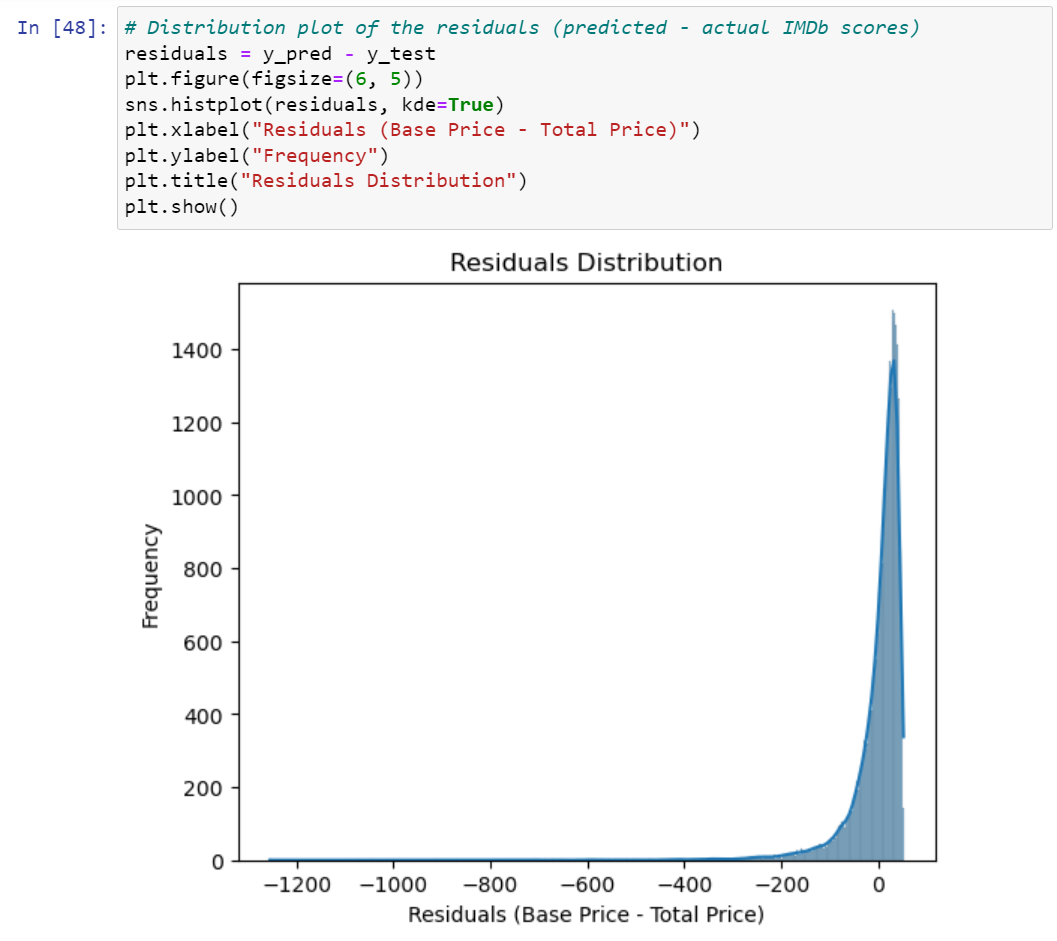
Importing the libraries:



Price vs Total Price:



Residual plot:



In this phase, we embarked on the journey of building a product demand prediction model for the dataset of units sold. We began by loading and preprocessing the dataset, which included handling missing data, encoding categorical features, and scaling numerical attributes.

Our model selection led us to a Random Forest Regressor, which has the advantage of capturing complex relationships within the data. After training the model, we evaluated its performance using metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and the R-squared (R2) coefficient. These metrics allowed us to assess the accuracy of our predictions.

Visualizations, including feature importance plots, residual plot and scatter plot provided additional insights into the model's performance. This comprehensive process equipped us with a powerful tool for predicting product demand, which can be invaluable for retailers, wholesalers and other businesses in assessing the potential success of the company.