Product Demand Prediction

Explanation about the statement:-

The question at hand is to develop a machine learning model that can predict product demand. This prediction is based on historical sales data and external factors like holidays, promotions, or economic indicators. The primary goal is to help businesses manage their inventory efficiently and plan production to meet customer needs effectively. This project involves collecting and preprocessing data, selecting the right machine learning algorithms, and training and evaluating models to accurately forecast product demand.

About Dataset :-

The DataSet is downloaded from kaggle website . Kaggle is like the Olympics of data science. It's a platform where data scientists and machine learning engineers from all over the world come together to collaborate and compete. Whether you're a seasoned pro or just dipping your toes into the world of data, Kaggle has something for everyone.

The dataset has totally 212644 records . The columns that are present in it are

1. Store ID: This column likely contains unique identifiers for different stores. Each row in the dataset corresponds to a specific store.
2. Total Price: This column probably represents the total revenue generated from the sales of the product in a particular store. It's likely calculated by multiplying the number of units sold by the price per unit (Total Price = Units Sold \* Base Price).
3. Base Price: This is likely the original or base price of the product per unit before any discounts or promotions. It's a key factor in calculating the total revenue.
4. Units Sold: This column represents the quantity of the product sold in a given store. It's a crucial indicator of the demand for the product in that particular location.

This dataset is to capture information about product sales in different stores, including details about pricing, quantity sold, and total revenue. Analyzing such data can provide insights into the performance of products across various locations and help optimize pricing and inventory strategies.

Libraries to be used :-

1. Pandas:

Pandas is essential for data manipulation and analysis. It helps you load, clean, and preprocess your dataset, making it suitable for machine-learning tasks.One of the ways to use pandas is to download Anaconda. You can download Anaconda From https://www.anaconda.com/download/

1. NumPy:

NumPy is a fundamental library for numerical operations. It's often used for array manipulation and mathematical functions in machine learning applications. You can download numpy from <https://numpy.org/install/>

1. Matplotlib and Seaborn:

These libraries are used for data visualization. You can create various types of plots and graphs to gain insights from your data. This library can be installed by using the command “pip install matplotlib”

1. Scikit-Learn:

Scikit-Learn is a powerful machine-learning library that provides tools for building predictive models. It includes various algorithms for regression and classification tasks. You can use it for demand prediction models. “pip install -U scikit-learn” to install scikit-learn.

Training the Model :-

* Data Collection: Gather historical data related to your products, including sales, pricing, promotions, and other relevant variables.
* Data Preprocessing: Clean the data by handling missing values and outliers. Encode categorical variables and scale or normalize numerical features. Split the data into features (independent variables) and the target variable (demand).
* Data Splitting: Split the dataset into a training set and a testing set. Common splits are 70-30 or 80-20, mostly used for training.
* Model Selection: Choose a machine learning algorithm suitable for demand prediction. Options include linear regression, decision trees, random forests, or more advanced methods like XGBoost or deep learning for time series forecasting.
* Model Training: Train the selected model using the training data. The model will learn the patterns and relationships in the data.

Testing the Model:

* Model Evaluation:

Use the testing set to assess the model's performance.

Calculate evaluation metrics like Mean Absolute Error (MAE), Mean Squared Error (MSE), and R-squared (R2) to measure how well the model predicts demand.

* Hyperparameter Tuning: Fine-tune the model's hyperparameters to optimize its performance. You can use techniques like cross-validation for this.
* Model Deployment: Once satisfied with the model's performance, deploy it to make predictions on new data.
* Monitoring and Maintenance: Continuously monitor the model in a real-world environment and retrain it as needed to keep it accurate.

Accuracy Check metrics :-

The choice of metrics for accuracy evaluation depends on the type of machine learning problem you are working on. In the context of product demand prediction, which is typically a regression problem (predicting numerical values), the following metrics are commonly used for accuracy evaluation:

1. Mean Absolute Error (MAE)

MAE calculates the average absolute differences between predicted and actual values. It's easy to understand because it represents an average error.

Formula:

MAE = (∑i= 1 to n |yi - hat{yi})/n

where (yi) is the actual value and hat{yi} is the predicted value for the ith sample.

1. Mean Squared Error (MSE)

MSE calculates the average of the squared differences between predicted and actual values.Squaring the errors gives higher weight to larger errors, making them sensitive to outliers.

Formula:

MSE = (∑i= 1 to n|(yi- hat{yi})^2)/n

1. Root Mean Squared Error (RMSE):

RMSE is the square root of the MSE, providing an interpretable metric in the same unit as the target variable. It gives an idea of how much your predictions deviate, on average, from the actual value .

Formula:

RMSE = sqrt{(∑i= 1 to n|(yi- hat{yi})^2)/n}

1. R-squared (R²) Score:

R-squared represents the proportion of the variance in the dependent variable (target) that is predictable from the independent variables (features).It ranges from 0 to 1, where 1 indicates a perfect fit.

Formula:

R^2 = 1 - 1/n{sumtion of i= 1 to n (y\_i - hat{y}\_i)^2}{sumtion of i= 1 to n(y\_i - \bar{y})^2}

where (bar{y}) is the mean of the actual values (y\_i).

For instance, MAE might be suitable if we want to understand the average prediction error in the same unit as your target variable. If we want to give more weight to larger errors, consider MSE or RMSE. R-squared provides an overall measure of how well the model fits the data.