# Predicting product demand using machine learning

**1. Data Collection:** Gather historical sales data, customer demographics, and other relevant information. This data can be collected from various sources such as sales records, online transactions, and customer surveys.

**2. Data Preprocessing:** Clean and preprocess the data to remove outliers, missing values, and inconsistencies. Convert data into a suitable format for machine learning.

**3. Feature Engineering:** Create relevant features from the data that can help in demand prediction. This might include variables like seasonality, location, and marketing campaigns.

**4. Model Selection:** Choose an appropriate machine learning model for demand prediction. Common models include regression, time series analysis, and neural networks.

**5. Training the Model:** Train the selected model on the preprocessed data. Use a portion of the data for training and another portion for testing to evaluate the model's performance.

**6. Evaluation:** Measure the model's performance using appropriate metrics (e.g., Mean Absolute Error, Root Mean Square Error). Refine the model as needed.

**7. Deployment:** Once the model is satisfactory, deploy it to make real-time predictions or generate demand forecasts.

**8. Continuous Monitoring:** Continuously monitor the model's performance and update it as new data becomes available or market conditions change.

Program

**# Import necessary libraries**

**import pandas as pd**

**from sklearn.model\_selection import train\_test\_split**

**from sklearn.linear\_model import LinearRegression**

**from sklearn.metrics import mean\_squared\_error**

**# Load your historical sales data (replace 'data.csv' with your dataset)**

**Data**

Payton program

# Import necessary libraries

import pandas as pd

from sklearn.model\_selection import train\_test\_split

from sklearn.linear\_model import LinearRegression

from sklearn.metrics import mean\_squared\_error

# Load your historical sales data (replace ‘data.csv’ with your dataset)

data = pd.read\_csv(‘data.csv’)

# Assuming your dataset contains columns like ‘date’, ‘product\_price’, ‘marketing\_budget’, and ‘demand’

# You might need to preprocess the ‘date’ column to extract useful features (e.g., month, day, year)

# Split the data into features (X) and target (y)

X = data[[‘product\_price’, ‘marketing\_budget’]]

y = data[‘demand’]

# Split the data into a training set and a testing set

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

# Create a linear regression model

model = LinearRegression()

# Train the model on the training data

model.fit(X\_train, y\_train)

# Make predictions on the test data

y\_pred = model.predict(X\_test)

# Evaluate the model’s performance using mean squared error

mse = mean\_squared\_error(y\_test, y\_pred)

print(f”Mean Squared Error: {mse}”)

# You can now use the trained model to predict demand for new product prices and marketing budgets

# For example:

new\_product\_price = 20

new\_marketing\_budget = 1000

predicted\_demand = model.predict([[new\_product\_price, new\_marketing\_budget]])

print(f”Predicted Demand: {predicted\_demand[0]}”)