# Product demand prediction machine learning

***Project Description***

The goal of this project is to develop a machine learning model that can accurately predict the demand for a specific product or a group of products. This prediction will be based on historical sales data and potentially other relevant features. The project involves several key components:

**1. Data Collection**

- Gather historical data on product sales, including timestamps, quantities sold, and any other relevant features such as pricing, promotions, and seasonal factors.

- Data sources may include point-of-sale records, online sales data, and inventory levels.

**2. Data Preprocessing**

- Clean and preprocess the collected data. This involves handling missing values, outliers, and data normalization.

- Create a suitable dataset for training and testing the machine learning model.

**3. Feature Engineering**

- Extract and engineer relevant features from the data that can help improve the accuracy of demand prediction.

- These features could include seasonality, product category, and historical sales trends.

**4. Model Selection**

* Choose a machine learning model for demand prediction. Common choices include time series forecasting models like ARIMA, regression models, or more advanced models like XGBoost or neural networks.

**5. Model Training**

- Split the dataset into training and testing sets.

- Train the selected model using the training data and tune hyperparameters to optimize its performance.

**6. Evaluation**

* Assess the model’s performance using appropriate evaluation metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), or Root Mean Squared Error (RMSE).

**7. Deployment**

* Deploy the trained model to make real-time or batch predictions for future product demand.

**8. Continuous Improvement**

* Implement monitoring and feedback loops to continuously improve the model’s accuracy as new data becomes available.

**9. Reporting and Visualization**

* Present the project results through clear visualizations and reports to stakeholders.

**10. Documentation**

* Document the entire project, including data sources, preprocessing steps, model selection, and deployment instructions.

**11. Conclusion**

* Summarize the project’s outcomes and discuss potential business impacts, such as optimizing inventory management and improving customer satisfaction.

**12. Future Enhancements**

* Suggest potential enhancements or extensions to the project, such as integrating external data sources or exploring advanced forecasting methods.

***Payton program***

# Import necessary libraries

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, r2\_score

# Load your dataset

Data = pd.read\_csv(‘your\_dataset.csv’)

# Prepare your data – You might need to preprocess and feature engineer your data

# Split the data into training and testing sets

X = data[[‘Feature1’, ‘Feature2’, …]] # Your relevant features

Y = data[‘Demand’] # Target variable

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

# Create and train the model

Model = LinearRegression()

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

# Make predictions

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

# Evaluate the model

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

R2 = r2\_score(y\_test, y\_pred)

# Print the evaluation metrics

Print(f”Mean Squared Error: {mse}”)

Print(f”R-squared: {r2}”)

# Now, you can use the trained model to predict demand for new data