

# PRODUCT DEMAND PREDICTION WITH MACHINE LEARNINGS

## **Team Members :**

- 1.B.NITHYASRI
- 2.M.VEEBALAKSHMI
- 3.P.PRATHISHA

**Department of electronics and communication engineering – III year Students of NPR  
college of engineering and technology - Natham Dindigul**

**# 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
```

```
import matplotlib.pyplot as plt
```

**# Load a sample dataset (you should replace this with  
your dataset)**

```
data = pd.DataFrame({
```

```
'Feature1': [1, 2, 3, 4, 5],
```

```
'Feature2': [2, 3, 4, 5, 6],  
'Feature3': [3, 4, 5, 6, 7],  
'Demand': [10, 15, 20, 25, 30]  
})
```

```
# Assuming you have features and target variable
```

```
X = data[['Feature1', 'Feature2', 'Feature3']] # Features
```

```
y = data['Demand'] # Target variable
```

```
# Split the data into training and testing sets
```

```
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()
```

```
# Fit the model to the training data
```

```
model.fit(X_train, y_train)
```

```
# Make predictions on the test set
```

```
y_pred = model.predict(X_test)
```

```
# Calculate the Mean Squared Error to evaluate the  
model
```

```
mse = mean_squared_error(y_test, y_pred)
```

```
print(f"Mean Squared Error: {mse}")
```

```
# Visualize the predictions
```

```
plt.scatter(y_test, y_pred)
```

```
plt.xlabel("Actual Demand")
```

```
plt.ylabel("Predicted Demand")
```

```
plt.title("Demand Prediction")
```

```
plt.show()
```

***OUTPUT :***

***PRODUCT DEMAND PREDICTION OUTPUT***

***048709793414476E-29***