PRODUCT DEMAND PREDICTION

**AIM:**

To predict the product demand using machine learning. It involves several steps, including feature engineering, model training, and evaluation.

**CODE:**

import pandas as pd

import numpy as np

import seaborn as sns

import matplotlib.pyplot as plt

data = pd.read\_csv('C:\\Users\\Mugun\\Desktop\\Dataset\\PoductDemand.csv')

print("Description of columns : \n")

print(data.describe()) #decription of each column

print("\nNo.of Null Columns :\n",data.isnull().sum()) #count of null values in columns

data = data.dropna() #to remove null data

**To analyze the difference between price and demand we have used a scatter plot:**

fig = px.scatter(data, x="Units Sold", y="Total Price",size='Units Sold')

fig.show()

print(data.corr())

Output:

ID Store ID Total Price Base Price Units Sold

ID 1.000000 0.007464 0.008473 0.018932 -0.010616

Store ID 0.007464 1.000000 -0.038315 -0.038848 -0.004372

Total Price 0.008473 -0.038315 1.000000 0.958885 -0.235625

Base Price 0.018932 -0.038848 0.958885 1.000000 -0.140032

Units Sold -0.010616 -0.004372 -0.235625 -0.140032 1.000000

correlations = data.corr(method='pearson')

plt.figure(figsize=(15, 12))

sns.heatmap(correlations, cmap="coolwarm", annot=True)

plt.show()

We have chosen the Total Price and the Base Price column as the features to train the model, and the Units Sold column as labels for the model:

x = data[["Total Price", "Base Price"]]

y = data["Units Sold"]

We split the data into training and test sets and use the decision tree regression algorithm to train our model:

xtrain, xtest, ytrain, ytest = train\_test\_split(x, y,test\_size=0.2,

random\_state=42)

from sklearn.tree import DecisionTreeRegressor

model = DecisionTreeRegressor()

model.fit(xtrain, ytrain)

Now let’s input the features (Total Price, Base Price) into the model and predict how much quantity can be demanded based on those values:

#features = [["Total Price", "Base Price"]]

features = np.array([[133.00, 140.00]])

model.predict(features)

**Final Output:**

array([27.])

**Conclusion:**

So this is how you can train a machine learning model for the task of product demand prediction using Python. Price is one of the major factors that affect the demand for the product. If a product is not a necessity, only a few people buy the product even if the price increases.