PRODUCT DEMAND PREDICTION WITH MACHINE LEARNINGS

TEAM MEMBER

911721104069 : NELLIYAN B

Phase 3 Submission Document

Project: Product Demand Analysis

Phase 3: Development Part 1

INTRODUCTION

In order to provide intelligent and meaningful responses, an in-depthexamination and assessment of various factors such as consumption growthpatterns, income and price elasticity of demand, market composition, nature ofcompetition, availability of substitutes, and teach of distribution channels isrequired.

Because of the importance of demand analysis, it should be done in amethodical and orderly manner.

The following are the major steps in such ananalysis:

- Situational analysis and goal-setting
- Secondary data collection
- Market survey
- Market characterisation
- Demand forecasting
- Market planning

Data sources

Built upon statistical models, machine learning utilizes additional internal and external sources of information to make more accurate, data-driven predictions. ML engines can work with both structured and unstructured data including

- past financial and sales reports (historical data),

- marketing polls,
- macroeconomic indicators,
- social media signals (retweets, shares, spikes in followers),
- weather forecasts,
- news about local events,
- competitors activity, and more.

Artificial Intelligence (AI)

In essence, this is the concept of trying to get machines to match human behaviour. In recent times, this has gained importance due, in part, to the sheer volume and variety of data that companies are now able to collect and the speed at which they can process it.

Machine Learning

This is a discipline within AI that is dedicated to the study of algorithms which are designed to perform a task using data. More importantly, the machine uses this data to automatically learn and improve without human intervention.

In essence, Machine Learning combines applied statistics and computer science with speed and precision to predict future behaviour. To do this, they need to collect and store a large amount of data (Big Data), which can often be a handicap for many businesses.

Data Science and Advanced Analytics

Data science covers the exploration and interpretation of data to unlock meaningful insights for businesses. Combining principles and practices from the fields of mathematics, statistics, Artificial Intelligence and Machine Learning, this advanced analysis approach allows businesses to analyse vast amounts of data.

This in turn can be used to make a descriptive analysis (which is based on explaining what has happened through statistics, graphs and tables). However, where Machine Learning techniques really add value is by performing two more sophisticated types of analysis: predictive analytics (making predictions based on past situations for future use); and prescriptive analysis (simulating different scenarios and evaluating what actions will attain the best results in the future).

Forecasting demand for new products

New products are notoriously difficult to plan for. However, by applying Machine Learning algorithms with advanced configuration, Machine-Learning-enabled systems can autonomously cluster demand history from multiple products to identify and anticipate trends in demand. This, in turn, enables the system to predict the potential future volume of demand.

The result: supply chain teams can build robust forecasts for new products far quicker, while simultaneously removing uncertainty and risk from new product launches.

Forecasting anomaly detection

Using similar techniques to those relied upon in fraud detection, Machine Learning techniques enable supply chain teams to identify outliers in demand history and exclude this from any analysis. Furthermore, by utilising advanced neural networks to cluster SKUs that are highly sensitive to anomalies, these products can be managed more proactively.

By detecting anomalies in customer transactions, availability and inventory status, the robustness of other supporting algorithms only improves.

Product Demand Prediction (Case Study)

A product company plans to offer discounts on its product during the upcoming holiday season. The company wants to find the price at which its product can be a better deal compared to its competitors. For this task, the company provided a dataset of past changes in sales based on price changes. We need to train a model that can predict the demand for the product in the market with different price segments.

The **dataset** that we have for this task contains data about:

1. the product id;

2. store id;
3. total price at which product was sold;
4. base price at which product was sold;
5. Units sold (quantity demanded);

Importance of loading and processing dataset:

Loading and preprocessing the dataset is an important first step in building any machine learning model. However, it is especially important for house price prediction models, as house price datasets are often complex and noisy.

By loading and preprocessing the dataset, we can ensure that the machine learning algorithm is able to learn from the data effectively and accurately.

Loading the dataset:

- ✓ Loading the dataset using machine learning is the process of bringing the data into the machine learning environment so that it can be used to train and evaluate a model.
- ✓ The specific steps involved in loading the dataset will vary depending on the machine learning library or framework that is being used. However, there are some general steps that are common to most machine learning frameworks:

Identify the dataset:

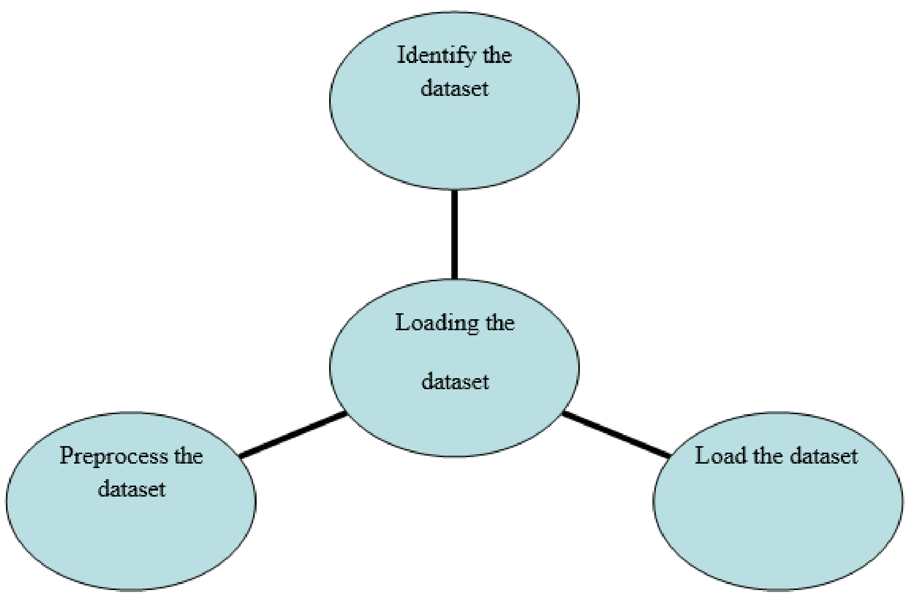
The first step is to identify the dataset that you want to load. This dataset may be stored in a local file, in a database, or in a cloud storage service.

Load the dataset:

Once you have identified the dataset, you need to load it into the machine learning environment. This may involve using a built-in function in the machine learning library, or it may involve writing your own code.

Preprocess the dataset:

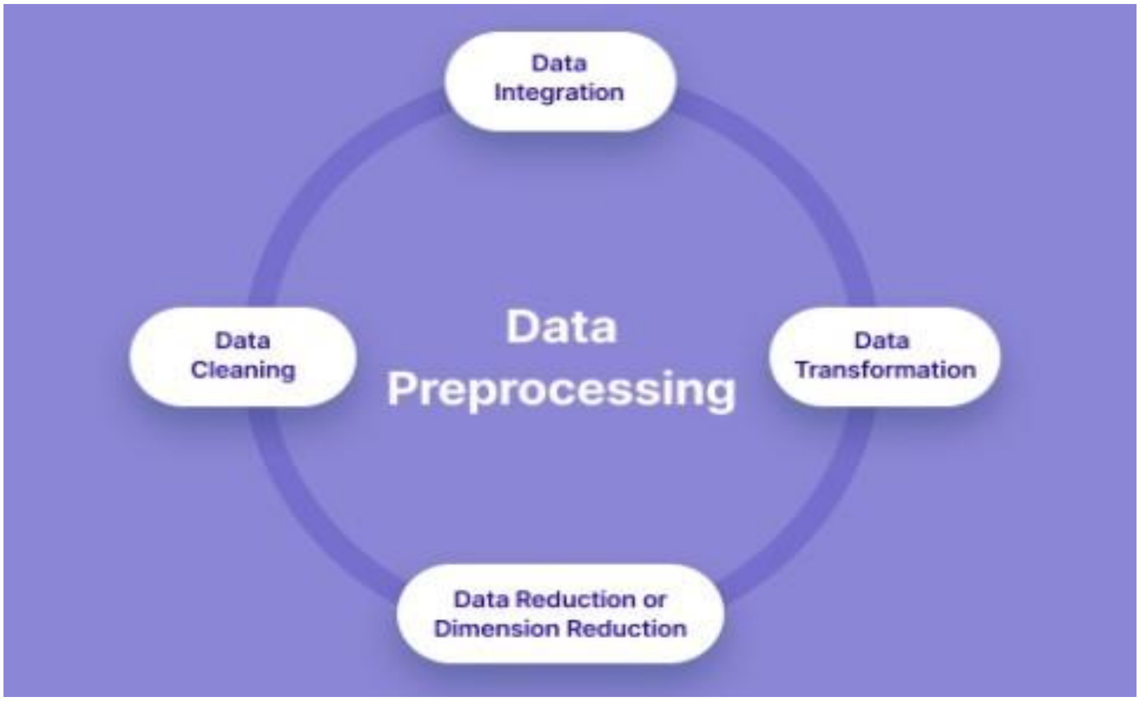
Once the dataset is loaded into the machine learning environment, you may need to preprocess it before you can start training and evaluating your model. This may involve cleaning the data, transforming the data into a suitable format, and splitting the data into training and test sets.



Some common data preprocessing tasks include:

- **Data cleaning:** This involves identifying and correcting errors and inconsistencies in the data. For example, this may involve removing duplicate records, correcting typos, and filling in missing values.
- **Data transformation:** This involves converting the data into a format that is suitable for the analysis task. For example, this may involve converting categorical data to numerical data, or scaling the data to a suitable range.
- **Feature engineering:** This involves creating new features from the existing data. For example, this may involve creating features that represent interactions between variables, or features that represent summary statistics of the data.
- **Data integration:** This involves combining data from multiple sources into a single dataset. This may involve resolving inconsistencies in the data, such as different data formats or different variable names.

Data preprocessing is an essential step in many data science projects. By carefully preprocessing the data, data scientists can improve the accuracy and



Preprocessing the dataset:

- Data preprocessing is the process of cleaning, transforming, and integrating data in order to make it ready for analysis.
- This may involve removing errors and inconsistencies, handling missing values, transforming the data into a consistent format, and scaling the data to a suitable range.

Product Demand Prediction using Python

Let's start by importing the necessary Python libraries and the dataset we need for the task of product demand prediction:

```
import pandas as pd

import numpy as np

import plotly.express as px

import seaborn as sns

import matplotlib.pyplot as plt

from sklearn.model_selection import train_test_split

from sklearn.tree import DecisionTreeRegressor


data=pd.read_csv("https://raw.githubusercontent.com/amankharwal/Website-data/master/demand.csv")

data.head()
```

	ID	Store ID	Total Price	Base Price	Units Sold
0	1	8091	99.0375	111.8625	20
1	2	8091	99.0375	99.0375	28
2	3	8091	133.9500	133.9500	19
3	4	8091	133.9500	133.9500	44
4	5	8091	141.0750	141.0750	52

Look at whether this dataset contains any null values or not:

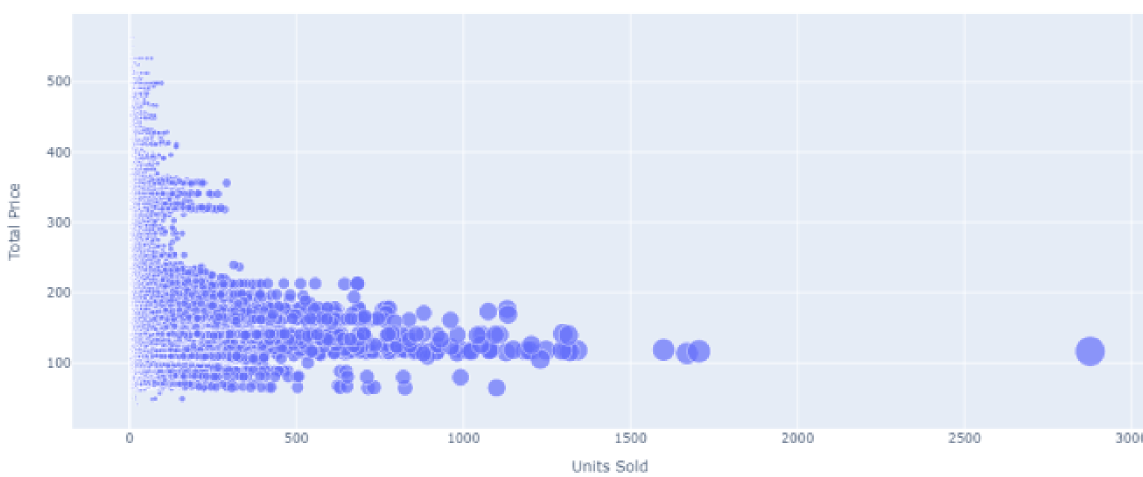
data.isnull().sum()

```
ID             0
Store ID       0
Total Price    1
Base Price     0
Units Sold     0
dtype: int64
```

So the dataset has only one missing value in the Total Price column, I will remove that entire row for now:

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

fig.show()



We can see that most of the data points show the sales of the product is increasing as the price is decreasing with some exceptions. Now let's have a look at the correlation between the features of the dataset:

print(data.corr())

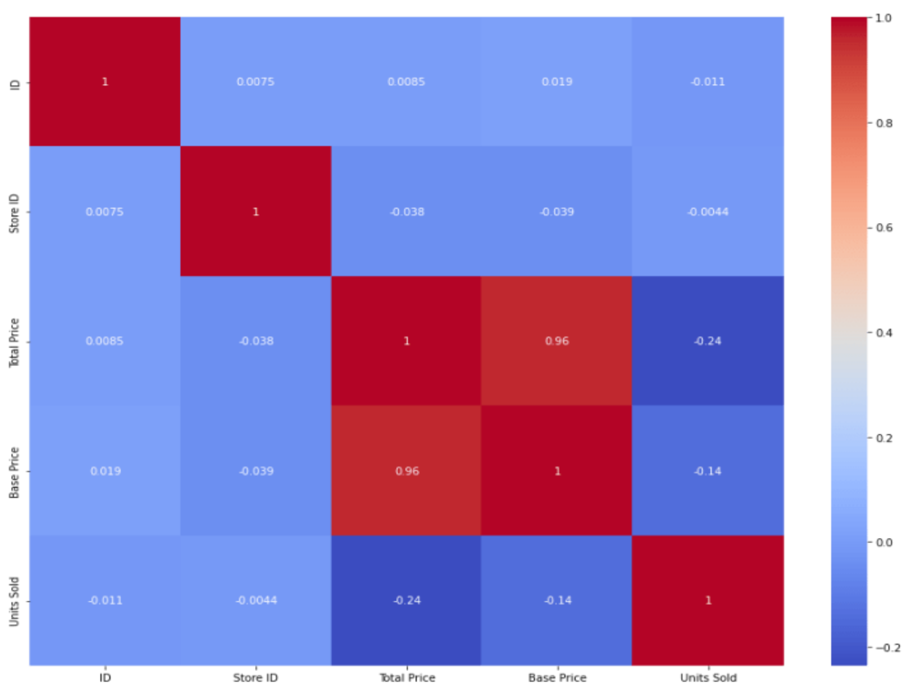
```
                   ID  Store ID  Total Price  Base Price  Units Sold
ID           1.000000  0.007464     0.008473    0.018932   -0.010616
Store ID     0.007464  1.000000    -0.038315   -0.038848   -0.004372
Total Price  0.008473 -0.038315     1.000000    0.958885   -0.235625
Base Price   0.018932 -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()



Product Demand Prediction Model

Now let's move to the task of training a machine learning model to predict the demand for the product at different prices. I will choose 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:**

```
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)
```

```
▾ DecisionTreeRegressor
DecisionTreeRegressor()
```

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)
```

```
array([27.])
```

CONCLUSION:

Customers today expect effective products and hassle free on-time services. These expectations could not be met without a strong supply-chain that involves strategic planning that includes demand forecasting.

The solution in this white paper is a statistical and ML-based solution that creates timeseries regarding each product and its entitlements based on geographic locations. The inputs of renewal rates and holidays based on each country or region helped generate accurate results by count and rate-based forecast on weekly basis. These forecasts assist the business in parts procurements and help budget planning for each financial year.

The Demand Forecasting project was originally used by services finance and part planning teams. But it has the potential to broaden its horizon by expanding the scope of the forecasting project and changing the granularity of forecast with expanded end users.