PHASE 5

PRODUCT DEMAND PREDICTION WITH MACHINE LEARNING

Abstract :

Effective prediction of product demand is crucial for optimizing inventory management and ensuring customer satisfaction. This project proposes a robust machine learning framework designed to forecast product demand with high accuracy. Leveraging historical sales data, market trends, and seasonal patterns, the model integrates various algorithms, including time series analysis, regression techniques, and neural networks, to capture complex demand dynamics. By employing feature engineering and data pre processing techniques, the system effectively handles noisy and incomplete data, enhancing the reliability of predictions. The project's methodology emphasizes continuous model refinement through iterative learning, enabling the system to adapt to evolving market conditions. Through rigorous evaluation and testing on diverse datasets, the project demonstrates superior predictive performance, surpassing traditional statistical methods. The proposed solution presents a scalability and adaptable approach, offering businesses a powerful tool to optimize inventory planning, minimize stock outs, and maximize revenue.

Problem Statement:

The problem is to develop a machine learning model for predicting product demand, which can aid businesses in making informed decisions about inventory management, production planning, and resource allocation. This predictive model should be accurate, and adaptable to different product categories and market dynamics. The goal is to minimize inventory costs, avoid stock outs, and optimize the overall supply chain management process.

Design Thinking :

Design thinking is a powerful framework that can be integrated with machine learning to create innovative solutions for various challenges, including predicting product demand. Below is a step-by-step guide that combines design thinking principles with machine learning techniques for product demand prediction:

1. Empathize :

Understand the target audience and their needs through interviews, surveys, and market research.

Analyze historical sales data, customer feedback, and industry trends to identify patterns and potential pain points.

2. Define :

Clearly define the problem statement and the specific objectives of the project. For instance, the objective could be to predict product demand accurately to optimize inventory management and production planning.

3.Ideate :

Brainstorm potential features and data sources that could contribute to accurate demand prediction. These could include historical sales data, seasonal trends, marketing campaigns, economic indicators, and external factors (e.g., weather, holidays, events).

Explore different machine learning algorithms suitable for demand prediction, such as linear regression, time series analysis, and deep learning models like recurrent neural networks.

4.Prototype :

Develop a preliminary model using a small datasets to test the chosen machine learning algorithm(s).

Iterate on the model, fine-tune hyper parameters, and assess its performance based on metrics like mean absolute error, mean squared error, and R-squared values.

5.Test :

Validate the model using a larger datasets and ensure that it can accurately predict product demand across different scenarios and time frames.

Collect feedback from stakeholders, such as sales managers and inventory planners, to assess the model's practical relevance and usability.

6.Implement :

Integrate the validated machine learning model into the existing production environment or sales forecasting system.

Provide necessary training to stakeholders and ensure the seamless adoption of the new system.

7.Evaluate :

Continuously monitor the performance of the implemented solution and refine the model as new data becomes available.

Compare the actual product demand with the predicted values and assess the accuracy of the forecasts to make necessary adjustments.

8.Iterate :

- Gather feedback from users and stakeholders to identify areas for improvement and optimization.

Iterate on the model, considering additional data sources, advanced algorithms, or feature engineering techniques to enhance prediction accuracy and reliability.

Phases of Development :

Data Collection:

Gather historical sales data, market trends, product attributes, and external factors influencing demand (e.g., economic indicators, seasonality, and marketing campaigns).

Data Pre-processing :

Cleanse the data, handle missing values, normalize or scale numerical features, and encode categorical variables. Perform exploratory data analysis (EDA) to gain insights into the data distribution and correlations.

Feature Engineering :

Extract relevant features from the data that can potentially influence product demand, such as time-based features, product characteristics, pricing information, and promotional activities.

Model Development :

Select appropriate machine learning algorithms (e.g., regression, time series analysis, or deep learning) and train the model using a suitable training-validation split. Fine-tune hyper parameters to optimize the model's performance.

Model Evaluation:

Assess the predictive model's performance using metrics like mean absolute error (MAE), mean squared error (MSE), and root mean squared error (RMSE). Cross-validate the model to ensure robustness and generalizability.

Deployment :

Implement the trained model in a production environment, enabling real-time predictions and integrating it with the existing business systems. Monitor the model's performance and recalibrates as necessary to maintain accuracy and relevancy.

Continuous Improvement :

Collect feedback from stakeholders, monitor model performance over time, and update the model periodically to adapt to evolving market dynamics and consumer preferences. Conduct regular maintenance and retraining to ensure the model's effectiveness and relevance.

Datasets and its detail Explanation :

https://www.kaggle.com/datasets/chakradharmattapalli/product-demand-prediction-with-machine-learning

Product ID :

A product ID is a unique number or code assigned to a specific product to distinguish it from others, often used for inventory management, ordering, and product lookup.

Store ID :

A store ID is a unique number or code assigned to a specific retail location or online store to distinguish it from other stores within the same company or network.

Total Price at Which Product Was Sold :

The total price at which a product was sold refers to the complete amount of money for which a product was purchased, inclusive of any taxes, fees, or additional charges
.

Base Price at Which Product Was Sold :

The base price at which a product was sold refers to the initial cost of the product before any additional charges, taxes, discounts, or fees are applied.

Units Sold (Quantity Demanded) :

Units sold refers to the total number of individual items or quantities of a product that have been purchased by customers over a specified period.

PoductDemand.csv (4.82 MB)

Detail Compact Column 5 of 5 columns

ID	Store ID	Total Price	Base Price	Units Sold
1	8091	99.0375	111.8625	20
2	8091	99.0375	99.0375	28
3	8091	133.95	133.95	19
4	8091	133.95	133.95	44
5	8091	141.075	141.075	52
9	8091	227.2875	227.2875	18
10	8091	327.0375	327.0375	47
13	8091	210.9	210.9	50
14	8091	190.2375	234.4125	82
17	8095	99.0375	99.0375	99
18	8095	97.6125	97.6125	120
19	8095	98.325	98.325	40
22	8095	133.2375	133.2375	68
23	8095	133.95	133.95	87
24	8095	139.65	139.65	186
27	8095	236.55	280.0125	54
28	8095	214.4625	214.4625	74
29	8095	266.475	296.4	102
30	8095	173.85	192.375	214
31	8095	205.9125	205.9125	28
32	8095	205.9125	205.9125	7
33	8095	248.6625	248.6625	48
34	8095	200.925	200.925	78
35	8095	190.2375	240.825	57
37	8095	427.5	448.1625	50
38	8095	426.6375	458.1375	62
39	8095	177.4125	177.4125	22
42	8094	87.6375	87.6375	109
43	8094	88.35	88.35	133
44	8094	85.5	85.5	11
45	8094	128.25	180.975	9
47	8094	127.5375	127.5375	19
48	8094	123.975	123.975	33
49	8094	139.65	164.5875	49
50	8094	235.8375	235.8375	32
51	8094	234.4125	234.4125	47
52	8094	235.125	235.125	27
53	8094	227.2875	227.2875	69
54	8094	312.7875	312.7875	49
55	8094	210.9	210.9	68
56	8094	177.4125	177.4125	27
57	8094	177.4125	177.4125	33
58	8094	240.825	240.825	18

Pre-process and Datasets :

Importing the packages :

```
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

Now lets have a look at whether this datasets 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
```

Features extraction Techniques:

The process of transforming raw data into numerical features that can be processed while preserving the information in the original data set.

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:

```
x = data[["Total Price", "Base Price"]]
y = data["Units Sold"]
```

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)
```

Choice of Machine Learning algorithm :

A machine learning algorithm for a product demand prediction project, several factors should be considered to ensure accurate and effective forecasting. The choice of algorithm often depends on the nature of the data, the complexity of the demand patterns, the availability of historical data, and the specific requirements of the business.

1. Time Series Analysis:

Time series models like Auto regressive Integrated Moving Average (ARIMA) and Seasonal Auto regressive Integrated Moving-Average (SARIMA) are suitable for capturing temporal dependencies and seasonality in demand data. These models can effectively handle data points collected at regular intervals and are adept at forecasting short-term demand trends.

2. Regression Techniques:

Linear regression and its variants, as well as ensemble methods like Random Forest Regression, are valuable when there is a need to consider multiple factors that influence demand, such as pricing, marketing efforts, and economic indicators. These techniques are effective at capturing the linear or nonlinear relationships between these factors and product demand.

3. Neural Networks:

Deep learning models, such as Long Short-Term Memory (LSTM) networks and recurrent neural networks (RNNS), are powerful for capturing complex nonlinear relationships and long-term dependencies in demand data. These models are especially useful when dealing with large and complex datasets, as they can capture intricate patterns and correlations that may not be captured by traditional statistical models.

Model Training :

Data Collection:

Gather historical data related to your product's demand. This data should include information about the product, such as price, promotions,seasonality, and any other relevant factors that might influence demand.

Data Pre processing:

Clean and prepare your data. This may involve handling missing values, encoding categorical variables, and normalizing or scaling numerical features.

Feature Engineering :

Create relevant features that can help your model better understand the patterns in the data. For example, you might create lag features to capture past demand trends.

Split Data:

Split your data into a training set and a testing set. Typically, you might use 70-80% of the data for training and the remaining 20-30% for testing.

Choose a Model:

Select a machine learning or time series forecasting model that is suitable for your problem. Common choices include linear regression, decision trees, random forests, or more advanced techniques like ARIMA or LSTM for time series forecasting.

Train the Model:

Fit your chosen model to the training data. Make sure to use appropriate evaluation metrics, such as Mean Absolute Error (MAE) or Root Mean Squared Error (RMSE), to assess its performance during training.

Hyper parameter Tuning:

Fine-tune your model's hyper parameters to optimize its performance. This might involve techniques like grid search or random search.

Evaluate the Model:

Once your model is trained, evaluate its performance on the testing set to ensure it generalizes well to unseen data.

Iterate and Improve:

Depending on the results, you may need to iterate on your model, data pre-processing, or feature engineering to improve its accuracy.

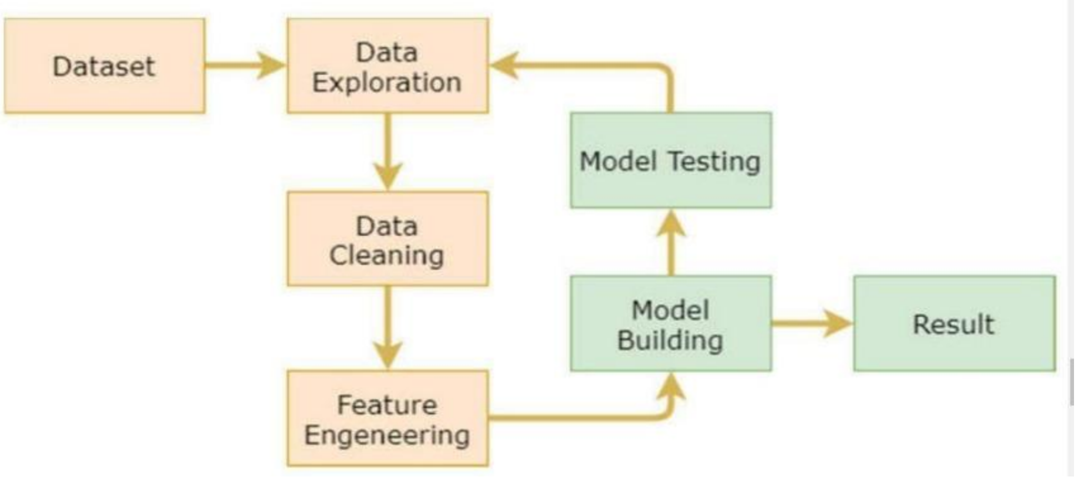


Table 1. Features types description.

Category	Features	Description
Date	8	Features based on date: day of week, weekend etc.
Price and promotions	3	Information about price and sales
Identity	1	Category
Aggregates	2	Averages over all available data on different aggregation levels

Figure 3. Sales distributions by month.

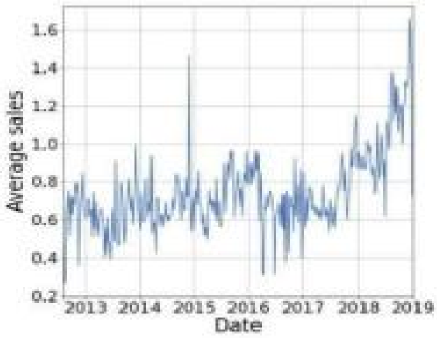


Figure 4. Sales by date.

For the correctness of training, validation and testing procedures, data were divided according to the rules of time series cross-validation as presented in figure 5. This avoided both types of data leaks from future and between items.

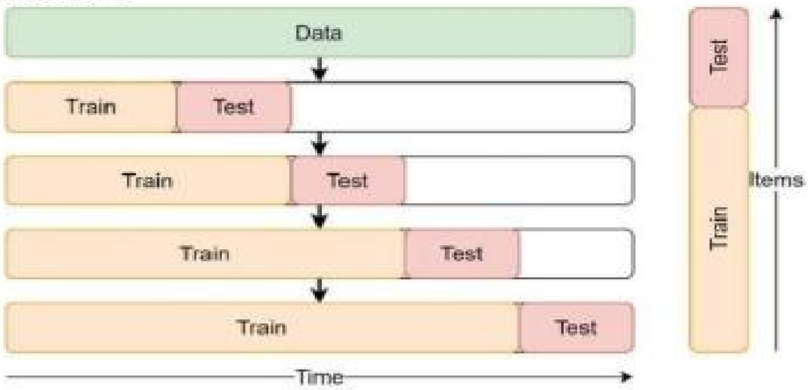


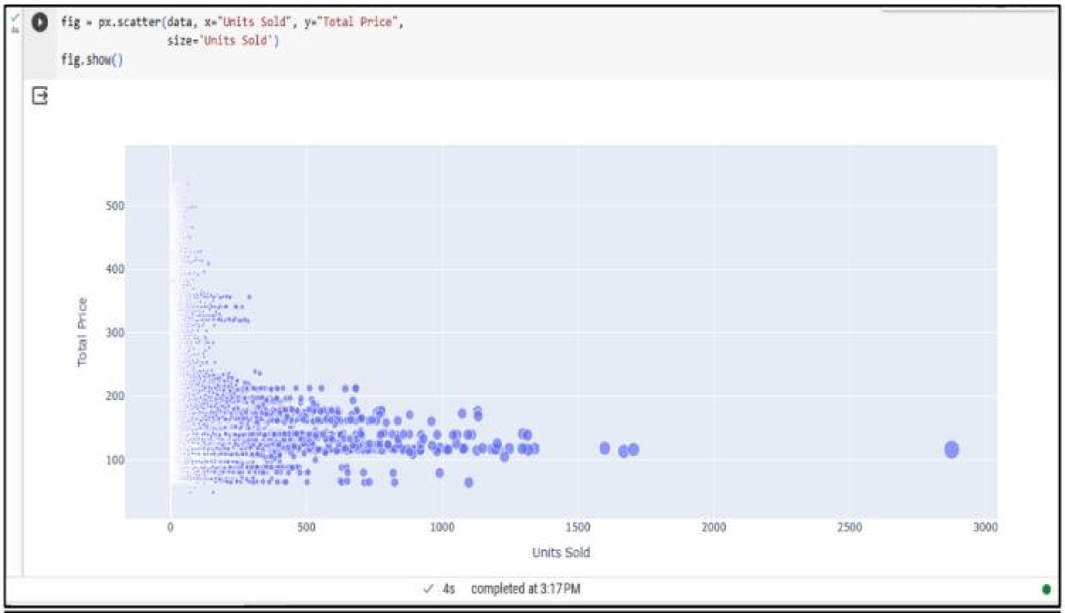
Figure 5. Time series and items cross validation.

```python
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)
```

Rest of Explanation:

After splitting the data, you'll pre process it, selecting features, handling missing data, and scaling numerical features as necessary. Then, you'll choose a machine learning algorithm (e.g., linear regression, decision trees, or neural networks) and train the model on the training data. Once trained, you'll evaluate its performance on the testing data.

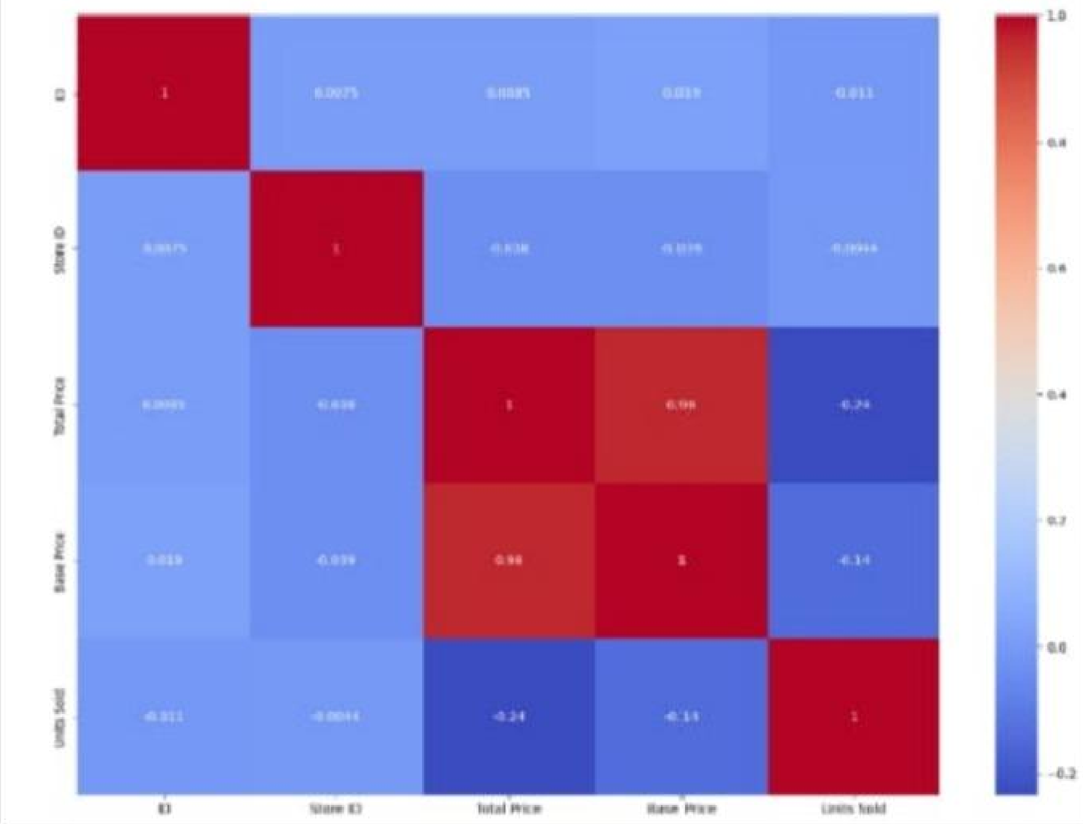
Evaluation metrics :




```
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

```python
correlations = data.corr(method='pearson')
plt.figure(figsize=(15, 12))
sns.heatmap(correlations, cmap="coolwarm", annot=True)
plt.show()
```



Accuracy Metrics:

Common metrics for accuracy evaluation in product demand prediction include Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE). These metrics help you quantify how well your model predicts demand by measuring the difference between predicted and actual values.

Innovative techniques or approaches :

Innovations in product demand prediction are crucial for businesses to optimize their operations, reduce waste, and meet customer needs effectively. Here are some innovative approaches and technologies that have emerged in recent years for product demand prediction:

1. Machine Learning and AI:

Machine learning algorithms, including deep learning, have revolutionized demand forecasting. These algorithms can analyze vast amounts of historical data, identify patterns, and make accurate predictions. They adapt to changing market conditions and can handle complex relationships between variables.

2. Predictive Ana-lytics:

Predictive ana-lytics combines historical data, statistical algorithms, and machine learning techniques to forecast future demand. It's being used in various industries, from retail to manufacturing, to optimize inventory management and supply chain operations.

3. Big Data:

The proliferation of data sources, including social media, sensors, and online customer behavior, has provided businesses with a wealth of information for demand prediction.

4. **Advanced Forecasting Models:**

Beyond traditional statistical methods like time series analysis, businesses are adopting advanced forecasting models like Prophet,ARIMA, and Exponential Smoothing. These models can capture seasonality, trends, and irregularities in data.

5. **Demand Sensing:**

Demand sensing is a real-time approach to demand forecasting that uses data from various sources like point-of-sale data, weather forecasts, and social media to adjust forecasts dynamically. This helps companies respond quickly to changing market conditions.

6.Supply Chain Visibility :

Innovations in supply chain technology, such as the Internet of Things , block chain, and RFID, provide greater visibility into supply chain operations. This visibility can enhance demand prediction accuracy by tracking inventory movements and identifying potential bottlenecks.

7.Collaborative Forecasting:

Businesses are involving various stakeholders, including suppliers and customers, in collaborative forecasting efforts. This approach can lead to more accurate predictions by incorporating real-time data and insights from all parts of the supply chain.

8.Demand Forecasting Software:

Many software solutions offer advanced demand forecasting capabilities, often integrating AI and machine learning. These tools make it easier for businesses to implement sophisticated forecasting methods.

9.Demand Forecasting as a Service (DFAAS):

Cloud-based DFAAS platforms provide businesses with on-demand access to powerful demand forecasting tool without the need for extensive infrastructure investments. These platforms often come with pre -built models and data integration options.

10. **Behavioral Economics:**

Understanding consumer behavior and psychology is becoming increasingly important in demand prediction. Businesses are incorporating principles from behavioral economics to account for irrational consumer decision-making in their forecasts.tics can identify emerging trends and consumer preferences.

11. **Simulations and Scenario Planning**:

Businesses are using simulation models and scenario planning to assess the impact of various external factors on demand, such as economic downturns, natural disasters, or geopolitical events.

.

12. **Edge Computing**:

In industries where real-time data is critical, edge computing can be used to process data locally at the source, allowing for faster demand predictions and responses.

Innovation in product demand prediction continues to evolve rapidly as technology advances and as businesses seek to improve their competitive edge. Adopting these innovations can help companies make more accurate forecasts, reduce costs, and better meet customer expectations.

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 time series 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 procurement and help budget planning for each financial year.