**PRODUCT DEMAND PREDICTION WITH MACHINE LEARNING**

**INTRODUCTION:**

* In demand forecasting, machine learning algorithms can analyze historical sales patterns and predict future trends.
* The first step is collecting data about past sales, such as
* product type
* quantity purchase frequency
* seasonality sold
* discounts, and more.
* Demand forecasting is the process of using predictive analysis of historical data to estimate and predict customers' future demand for a product or service.
* Demand forecasting helps the business make better-informed supply decisions that estimate the total sales and revenue for a future period of time.

The limitations of traditional linear regression models in capturing complex

relationships.

 Emphasize the need for advanced regression techniques like Gradient Boosting and

XG Boost to enhance prediction accuracy.

Content for Project Phase 2 :

Consider exploring advanced regression techniques like Gradient Boosting or XG Boost for

improved Prediction accuracy.

Data Source

A good data source for product demand prediction using machine learning should be

Accurate, Complete, Covering the ID, STORE ID,TOTAL PRICE and unit sold.

Dataset Link:

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

In above Dataset link given about

ID

Store Id

Total price

Base price

Unit sold

Based on product demand prediction with machine learning.

Data Collection:

Data collection is the process of gathering data for use in business decision-making, strategic planning, research and other purposes.

Collect historic sales data and external factors that influence demand, such as marketing campaigns, holidays, economic indicators, etc.

**Data collection is the process of gathering and measuring information from countless different sources. In order to use the data we collect to develop practical artificial intelligence (AI) and machine learning solutions, it must be collected and stored in a way that makes sense for the business problem at hand.**

**Data Preprocessing:**

* **Clean and preprocess the data, handle missing values, and convert categorical features into numerical representations**
* **Data preprocessing transforms the data into a format that is more easily and effectively processed in data mining, machine learning and other data science tasks.**
* **Data Processing is the task of converting data from a given form to a much more usable and desired form i.e. making it more meaningful and informative. Using Machine Learning algorithms, mathematical modeling, and statistical knowledge, this entire process can be automated.**

**Feature Engineering:**

* Feature engineering is the pre-processing step of machine learning, which is used to transform raw data into features that can be used for creating a predictive model using Machine learning or statistical Modelling. Feature engineering in machine learning aims to improve the performance of models.
* Create additional features that capture seasonal patterns, trends, and external influences on product demand.

Advanced Regression Techniques:

 Ridge Regression: Introduce L2 regularization to mitigate multicollinearity and

overfitting.

 Lasso Regression: Employ L1 regularization to perform feature selection and

simplify the model.

 Elastic Net Regression: Combine both L1 and L2 regularization to benefit from their  Ridge Regression: Introduce L2 regularization to mitigate multicollinearity and

overfitting.

 Lasso Regression: Employ L1 regularization to perform feature selection and

simplify the model.

 Elastic Net Regression: Combine both L1 and L2 regularization to benefit from their respective advantages.

 Random Forest Regression: Implement an ensemble technique to handle nonlinearity and capture complex relationships in the data.

 Random Forest Regression: Implement an ensemble technique to handle nonlinearity

and capture complex relationships in the data.

 Gradient Boosting Regressors (e.g., XG Boost, Light GBM): Utilize gradient

boosting algorithms

**Model Selection:**

**Choose suitable regression algorithms (e.g., Linear Regression, Random Forest, XGBoost) for demand forecasting.**

**Model selection in machine learning is the process of selecting the best algorithm and model architecture for a specific job or dataset. It entails assessing and contrasting various models to identify the one that best fits the data & produces the best results.**

**LIBRARIES :**

 Pandas is a Python library used for working with data sets. It has functions for analyzing, cleaning, exploring, and manipulating data.

The name "Pandas" has a reference to both "Panel Data", and "Python Data Analysis" and was created by Wes McKinney in 2008.

NUMPY :

NumPy is a Python library used for working with arrays. It also has functions for working in domain of linear algebra, fourier transform, and matrices. NumPy was created in 2005 by Travis Oliphant.

MATPLOTLIB:

Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python.

Matplotlib makes easy things easy and hard things possible. Create publication quality plots.

Make interactive figures that can zoom, pan, update.

SciPy:

The name “SciPy” stands for “Scientific Python”. It is an open-source library used for high-level scientific computations. This library is built over an extension of Numpy. It works with Numpy to handle complex computations. While Numpy allows sorting and indexing of array data, the numerical data code is stored in SciPy. It is also widely used by application developers and engineers.

**Model Training**

**Model training is the phase in the data science development lifecycle where practitioners try to fit the best combination of weights and bias to a machine learning algorithm to minimize a loss function over the prediction range.**

PROGRAM:

**from s k learn import datasets  
from s k learn . model \_ selection import train \_ test \_ split  
from s k learn. metrics import accuracy \_ score  
from s k learn .tree import Decision Tree Classifier**

**data = datasets .load \_ wine(as \_ frame = True)  
X = data . data  
y = data . target**

**X \_ train, X \_ test, y \_ train, y \_ test = train \_ test \_ split(X, y, test \_ size = 0.25, random \_ state = 22)**

**D tree = Decision Tree Classifier (random \_ state= 22)  
d tree . fit (X \_ train , y \_ train)**

OUTPUT:

**Decision Tree Classifier (random \_ state=22)**