PRODUCT DEMAND PREDICTION WITH MACHINE LEARNING

OBJECTIVE:

To build a model that assigns a score to each given sales data and then based on a threshold value, classifies the 'Product in Demand' from the overall sale data of a store, which can then be used to increase the overall sale rate of the store.

ABSTRACT:

To improve a sales team's gains by defining a clear classification within a given set of demand so that the store focuses on the supply chain for the demanded product.

To help them focus on the demand with the most priority, thus increasing the overall sales rate and reducing the risk of the sales for non-demand products

PROBLEM DEFINITION AND DESIGN THINKING:

To understand the problem statement and create a document on what have we understood and how will we proceed ahead with solving the problem.

PROBLEM DEFINITION:

The problem is to create a machine learning model that forecasts product demand based on historical sales data and external factors.

The goal is to help businesses optimize inventory management and production planning to efficiently meet customer needs.

This project involves data collection, data pre-processing, feature engineering, model selection, training, and evaluation.

DESIGN THINKING:

- Data Collection: Collect historical sales data and external factors that influence demand, such as marketing campaigns, holidays, economic indicators, etc.
- Data Pre-processing: Clean and pre-process the data, handle missing values, and convert categorical features into numerical representations.
- Feature Engineering: Create additional features that capture seasonal patterns, trends, and external influences on product demand.
- Model Selection: Choose suitable regression algorithms (e.g., Linear Regression, Random Forest, XG Boost) for demand forecasting.
- Model Training: Train the selected model using the preprocessed data.
- Evaluation: Evaluate the model's performance using appropriate regression metrics (e.g., Mean Absolute Error, Root Mean Squared Error).

DATASET :

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

SALE DEMAND SCORE PREDICTION:

The model uses a Logistic regression library as its core to fit in the given dataset. This library is referred by the model to observe the underlying pattern within the given dataset. In a brief it tries to apply and fit o the logistic regression formula,

Y = b0 + b1x1 + b2x2 +b2x2 +...+bnxn Where,

b0...bn denotes coefficients of the input features and x1...xn denotes the input features.

But this approach uses a transformed version of this formula to specifically find a probabilistic value for a positive outcome such as,

P(Y = 1/x) = 1 / (1 + exp(-(b0 + b1x1 + b2x2 +b2x2 +...+bnxn)))

The above formula can be described as the probability of the outcome being 1 given the x features.

Where,

b0...bn denotes coefficients of the input features and x1...xn denotes the input features

DEVELOPMENT PROCESS:

CREATION:

Step 1: The '.csv' files are loaded into the model.

Step 2: The dataset is pre-processed by handling any missing values and removing the least relevant features

Step 3: The dataset is classified as variables for prediction and variables for the features.

Step 4: The dataset is then split into training and testing sections.

Step 5: The parameters for randomness are defined.

Step 6: The dataset is observed for underlying patterns.

Step 7: The dataset is then fit into the model.

TUNING:

Step 8: A test metric is used to evaluate the model's accuracy.

Step 9: Based on the accuracy the model's hyper-parameters are tuned.

Step 10: Step 9 is repeated until the model reaches the desired accuracy.

LIBRARIES/MODULES USED:

- PANDAS for dataset manipulation,
- NUMPY for encoded variables manipulation,
- SEABORN for visualization,
- SCI-KIT LEARN for the core
- MODEL SELECTION

  train_test_split for the division within the dataset PRE-PROCESSING

  StandardScaler for iteration tuning LINEAR MODEL

  Logistic_Regression for the core

METRICS:

- confusion_matrix,
- accuracy_score and
- classification_report for the evaluation.

FEATURES:

The features included in the model are,

- Total value,
- Base value,
- Units sold &
- Store Id

The y variable for prediction is set to 'Demand'

EVALUATED SCORE:

The current model is evaluated for an overall accuracy score of 98%

ACCURACY:

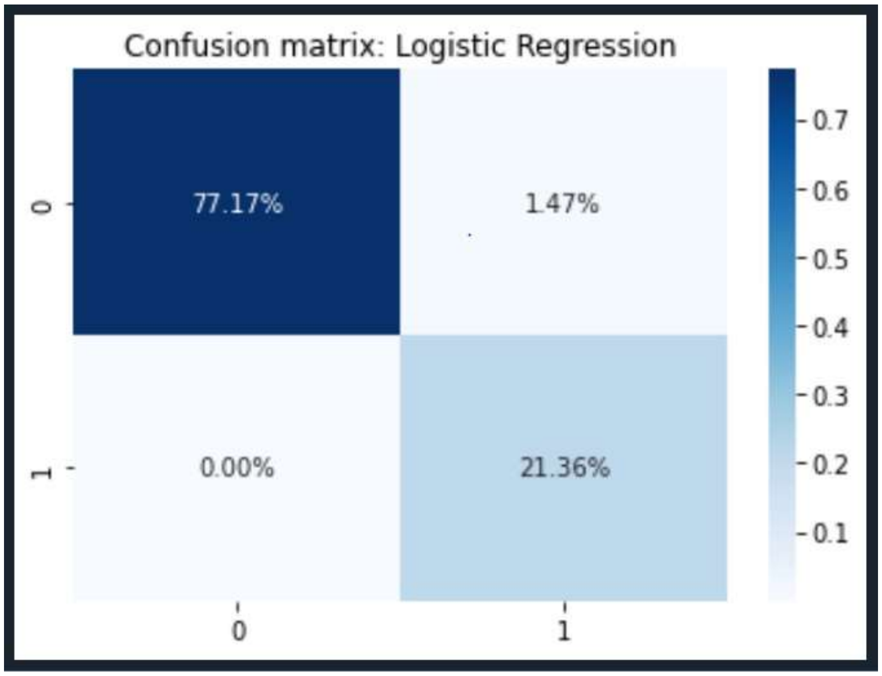
TRUE POSITIVES : 77.1%

TRUE NEGATIVES: 21.36%

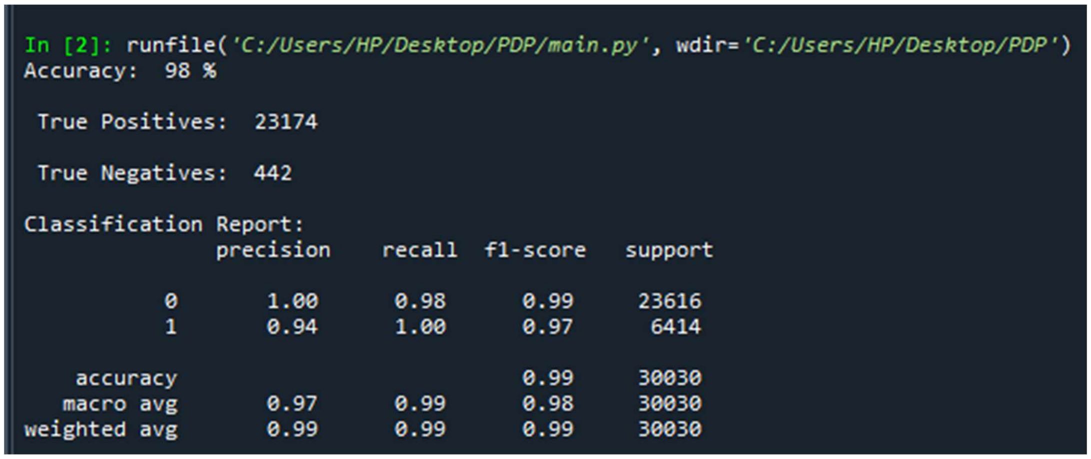
FALSE POSITIVES: 1.47%

FALSE NEGATIVES: nil

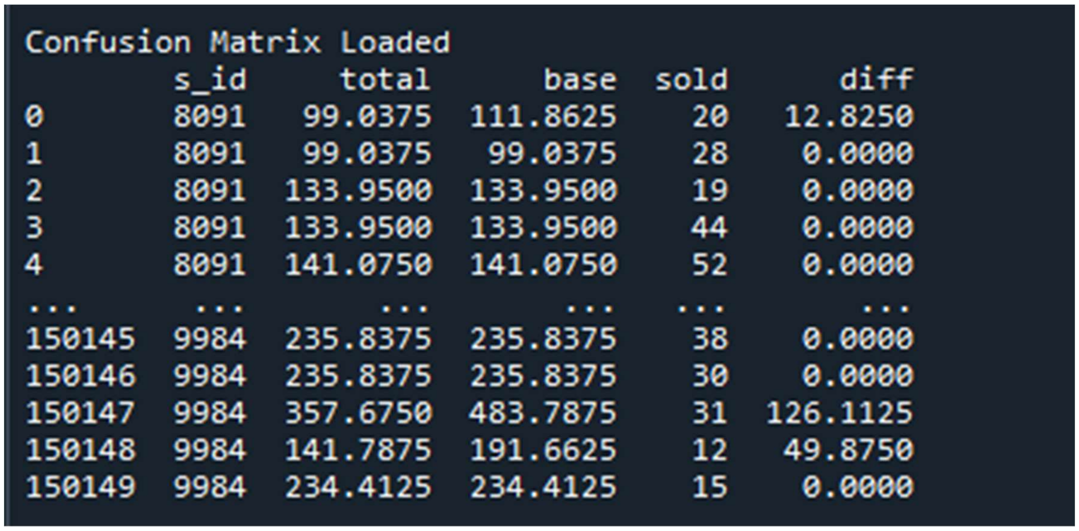
CONFUSION MATRIX SCREENSHOT (EVALUATION):



PERFORMANCE:

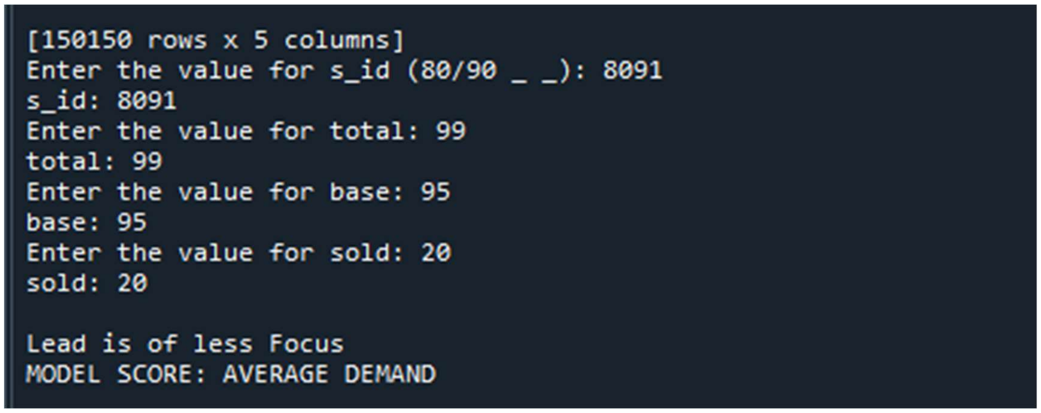


```
In [2]: runfile('C:/Users/HP/Desktop/PDP/main.py', wdir='C:/Users/HP/Desktop/PDP')
Accuracy:  98 %

 True Positives:  23174

 True Negatives:  442

Classification Report:
              precision    recall  f1-score   support

           0       1.00      0.98      0.99     23616
           1       0.94      1.00      0.97      6414

    accuracy                           0.99     30030
   macro avg       0.97      0.99      0.98     30030
weighted avg       0.99      0.99      0.99     30030
```

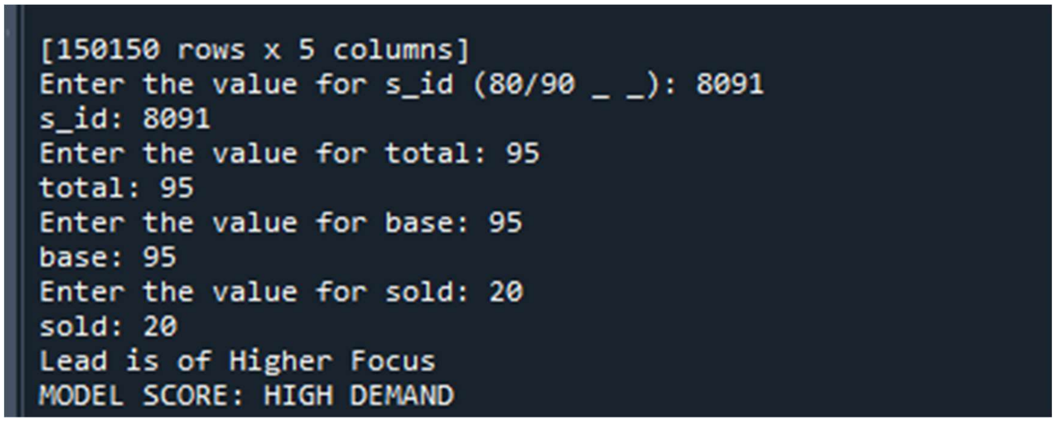


```
Confusion Matrix Loaded
        s_id     total      base  sold      diff
0       8091   99.0375  111.8625    20   12.8250
1       8091   99.0375   99.0375    28    0.0000
2       8091  133.9500  133.9500    19    0.0000
3       8091  133.9500  133.9500    44    0.0000
4       8091  141.0750  141.0750    52    0.0000
...      ...       ...       ...   ...       ...
150145  9984  235.8375  235.8375    38    0.0000
150146  9984  235.8375  235.8375    30    0.0000
150147  9984  357.6750  483.7875    31  126.1125
150148  9984  141.7875  191.6625    12   49.8750
150149  9984  234.4125  234.4125    15    0.0000
```

TEST PREDICTIONS:



```
[150150 rows x 5 columns]
Enter the value for s_id (80/90 _ _): 8091
s_id: 8091
Enter the value for total: 99
total: 99
Enter the value for base: 95
base: 95
Enter the value for sold: 20
sold: 20

Lead is of less Focus
MODEL SCORE: AVERAGE DEMAND
```



```
[150150 rows x 5 columns]
Enter the value for s_id (80/90 _ _): 8091
s_id: 8091
Enter the value for total: 95
total: 95
Enter the value for base: 95
base: 95
Enter the value for sold: 20
sold: 20
Lead is of Higher Focus
MODEL SCORE: HIGH DEMAND
```