INFOSYS SPRINGBOARD INTERNSHIP REPORT – FIRST REVIEW

Title: **On-Time Shipment Prediction using AI/ML**

Intern Name: **Gladwin C Bino**

Institution: **LBS College of Engineering, Kasaragod**

Mentor: **Springboard Mentor**

Duration: 24th September 2025 – November 2025

Domain: **Artificial Intelligence / Machine Learning**

Abstract

The project 'On-Time Shipment Prediction using AI/ML' focuses on predicting whether a shipment will be delivered on time or delayed using machine learning techniques. In the modern logistics sector, delivery timeliness is crucial for maintaining customer satisfaction and optimizing supply chain efficiency. This project involves data preprocessing, feature analysis, visualization, and machine learning model development to identify patterns influencing delivery performance. By leveraging predictive analytics, the system aims to help businesses proactively manage logistics, minimize delays, and improve overall operational efficiency.

Introduction

In today's fast-paced supply chain industry, timely delivery of shipments plays a vital role in customer satisfaction and brand reliability. However, due to multiple factors such as distance, traffic, weather, and warehouse handling delays, shipments may not always reach on time. With the advancement of Artificial Intelligence and Machine Learning, predictive modeling can now assist in forecasting shipment delays based on historical data. This project leverages machine learning algorithms to predict whether a shipment will arrive on time or not, providing insights for better decision-making and resource planning.

Objectives of the Project

- To develop a machine learning model to predict on-time or delayed shipments.

- To analyze key factors affecting shipment delivery time.

- To visualize and interpret data patterns influencing delivery delays.

- To improve logistics efficiency through predictive insights.

Methodology

The methodology of this project includes several stages of AI/ML pipeline development. The steps followed so far are summarized below:

- Dataset Collection – Downloaded dataset from Kaggle containing over 10,000 shipment records.

- Data Cleaning – Handled missing values using mean and mode imputation techniques.

- Descriptive Statistics – Computed measures such as mean, median, and standard deviation to understand data distribution.

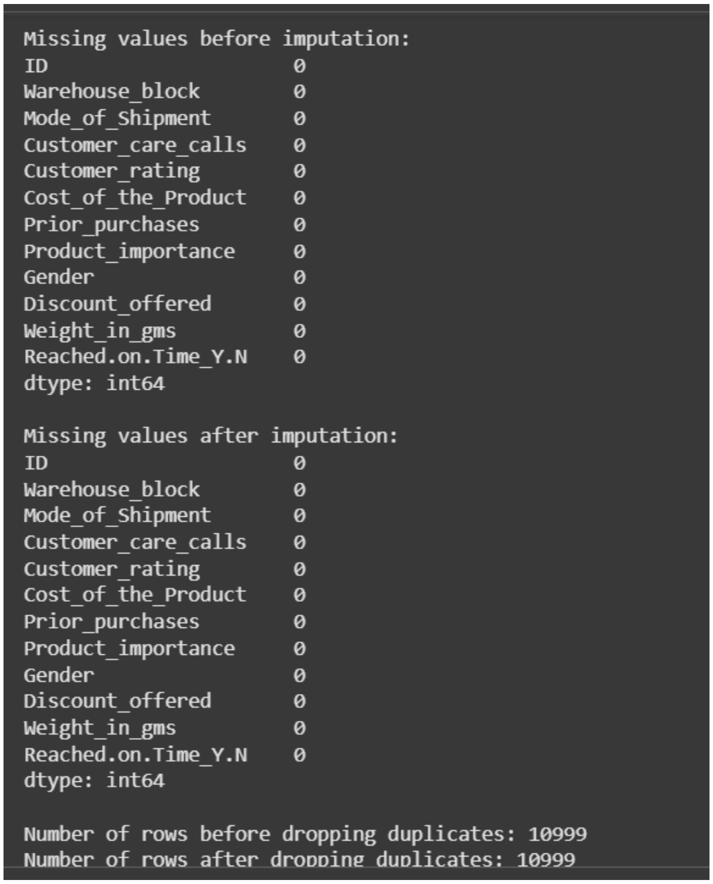
- Univariate & Bivariate Analysis – Analyzed features individually and in relation to the target variable.

- Data Visualization – Used Matplotlib and Seaborn to visualize relationships and detect outliers.

- Addressed Class Imbalance – Applied oversampling to balance the target classes (On Time vs. Not On Time).

Work Done So Far

Date	Work Description
24th September 2025 (Wednesday)	Browsed the internet to identify relevant datasets for shipment and delivery prediction. Selected and downloaded a dataset from Kaggle containing 10,000+ rows with multiple shipment-related features. Briefly reviewed the dataset structure (rows, columns, feature types).
25th September 2025 (Thursday)	Began data cleaning and preprocessing using Pandas in Google Colab. Identified missing values and replaced them systematically using mean and mode for numerical and categorical columns respectively.

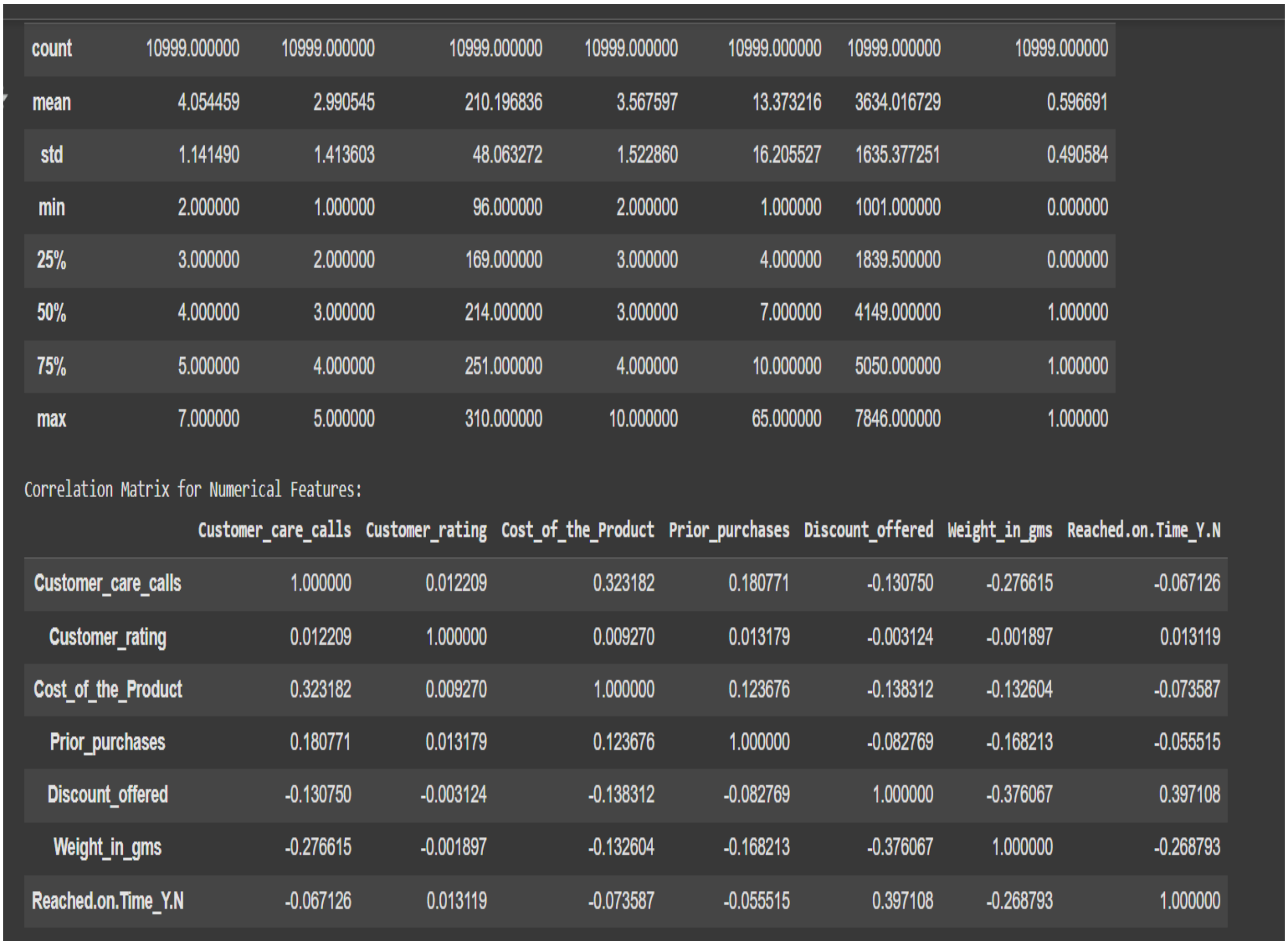


```
Missing values before imputation:
ID                     0
Warehouse_block        0
Mode_of_Shipment       0
Customer_care_calls    0
Customer_rating        0
Cost_of_the_Product    0
Prior_purchases        0
Product_importance     0
Gender                 0
Discount_offered       0
Weight_in_gms          0
Reached.on.Time_Y.N    0
dtype: int64

Missing values after imputation:
ID                     0
Warehouse_block        0
Mode_of_Shipment       0
Customer_care_calls    0
Customer_rating        0
Cost_of_the_Product    0
Prior_purchases        0
Product_importance     0
Gender                 0
Discount_offered       0
Weight_in_gms          0
Reached.on.Time_Y.N    0
dtype: int64

Number of rows before dropping duplicates: 10999
Number of rows after dropping duplicates: 10999
```

26th September 2025 (Friday)	Generated descriptive statistics of the dataset using .describe() and other summary functions to understand the distribution of numerical and categorical features.

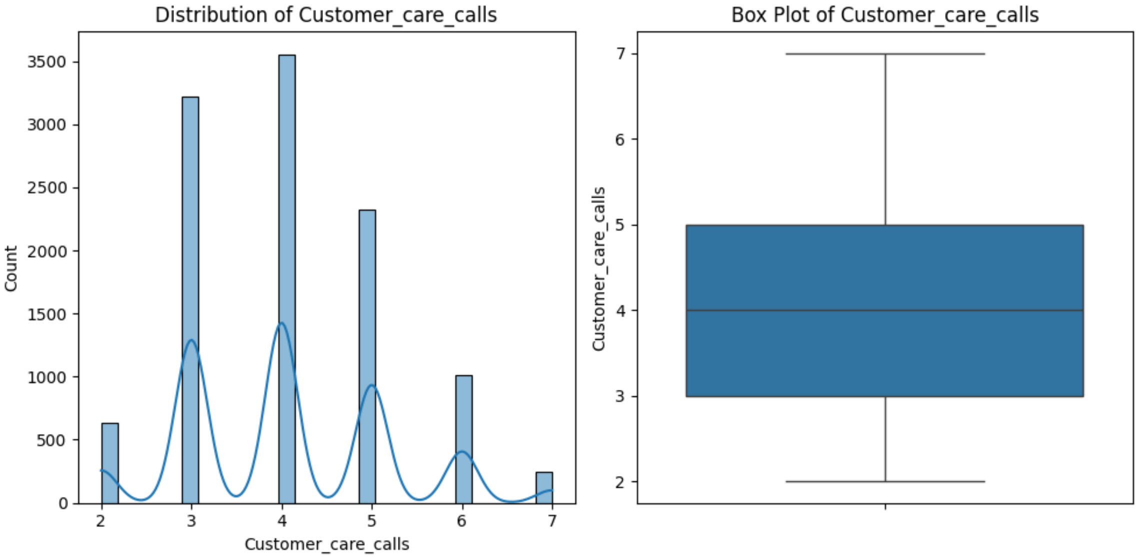


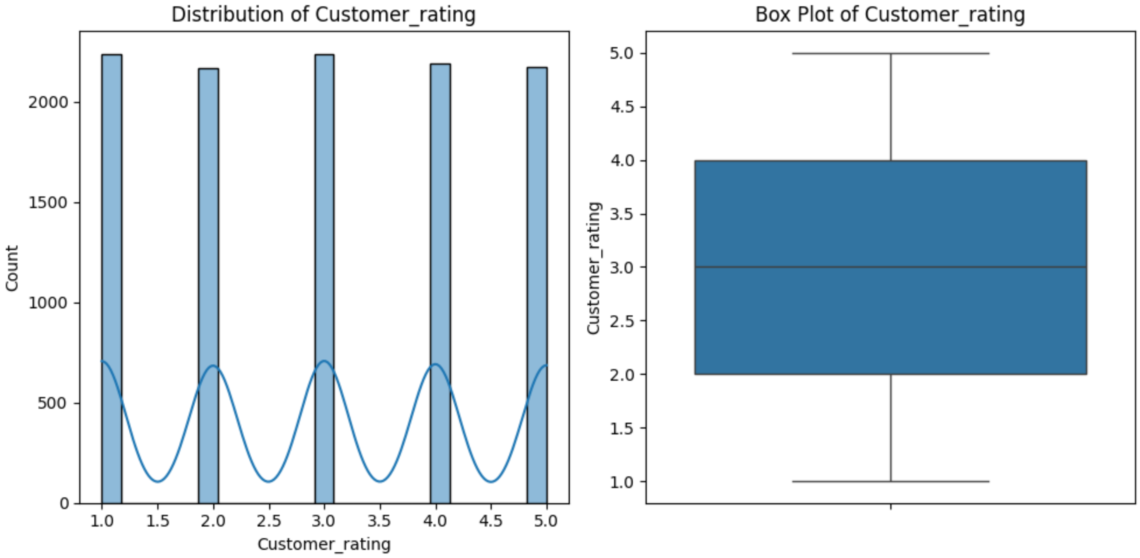
count	10999.000000	10999.000000	10999.000000	10999.000000	10999.000000	10999.000000	10999.000000
mean	4.054459	2.990545	210.196836	3.567597	13.373216	3634.016729	0.596691
std	1.141490	1.413603	48.063272	1.522860	16.205527	1635.377251	0.490584
min	2.000000	1.000000	96.000000	2.000000	1.000000	1001.000000	0.000000
25%	3.000000	2.000000	169.000000	3.000000	4.000000	1839.500000	0.000000
50%	4.000000	3.000000	214.000000	3.000000	7.000000	4149.000000	1.000000
75%	5.000000	4.000000	251.000000	4.000000	10.000000	5050.000000	1.000000
max	7.000000	5.000000	310.000000	10.000000	65.000000	7846.000000	1.000000

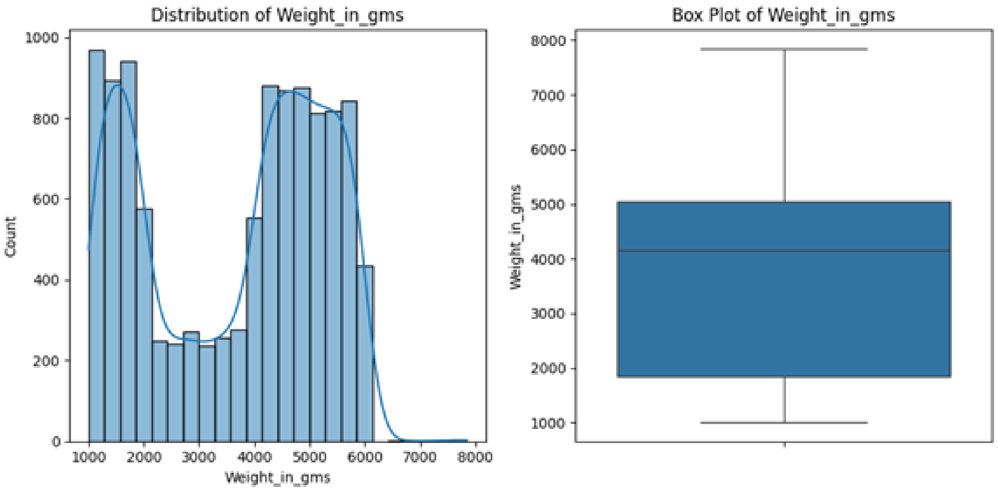
```
Correlation Matrix for Numerical Features:
```

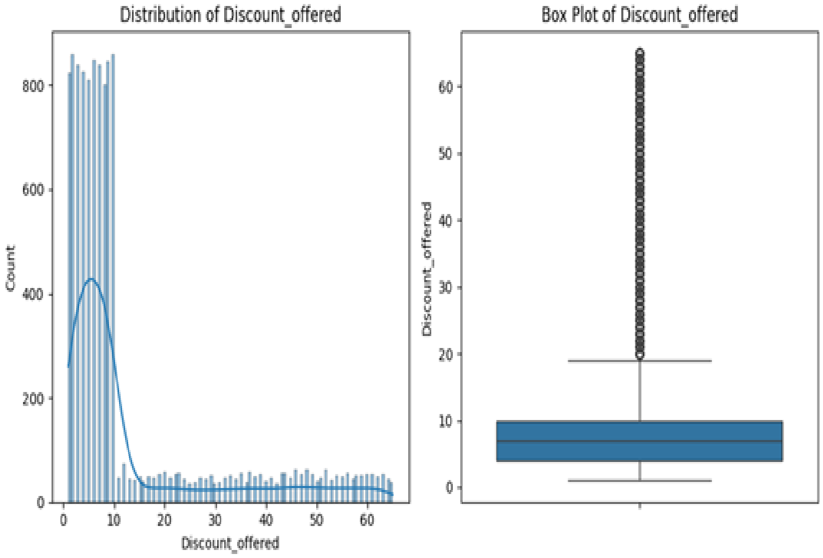
	Customer_care_calls	Customer_rating	Cost_of_the_Product	Prior_purchases	Discount_offered	Weight_in_gms	Reached.on.Time_Y.N
Customer_care_calls	1.000000	0.012209	0.323182	0.180771	-0.130750	-0.276615	-0.067126
Customer_rating	0.012209	1.000000	0.009270	0.013179	-0.003124	-0.001897	0.013119
Cost_of_the_Product	0.323182	0.009270	1.000000	0.123676	-0.138312	-0.132604	-0.073587
Prior_purchases	0.180771	0.013179	0.123676	1.000000	-0.082769	-0.168213	-0.055515
Discount_offered	-0.130750	-0.003124	-0.138312	-0.082769	1.000000	-0.376067	0.397108
Weight_in_gms	-0.276615	-0.001897	-0.132604	-0.168213	-0.376067	1.000000	-0.268793
Reached.on.Time_Y.N	-0.067126	0.013119	-0.073587	-0.055515	0.397108	-0.268793	1.000000

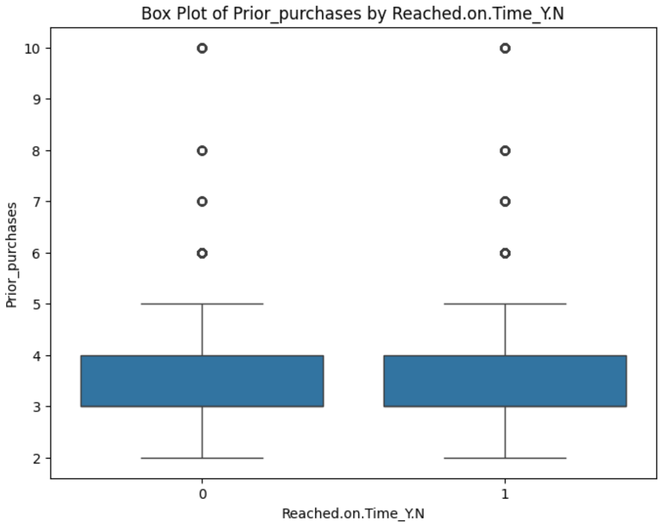
29th September 2025 (Monday)	Conducted Univariate and Bivariate Analysis to study feature distributions and correlations with the target variable (reached_on_time).

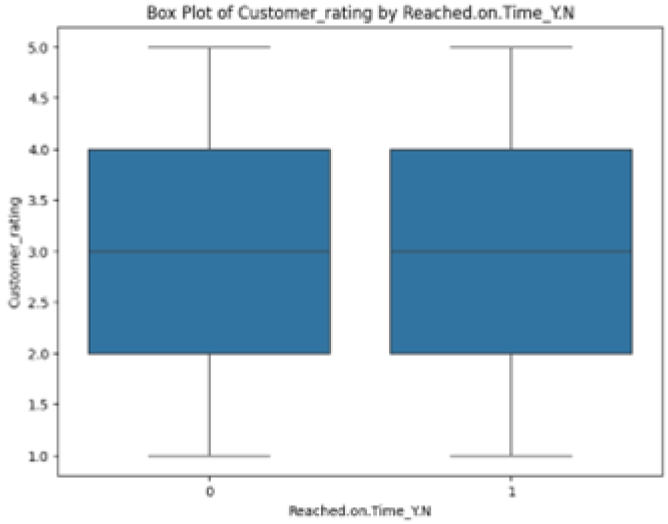




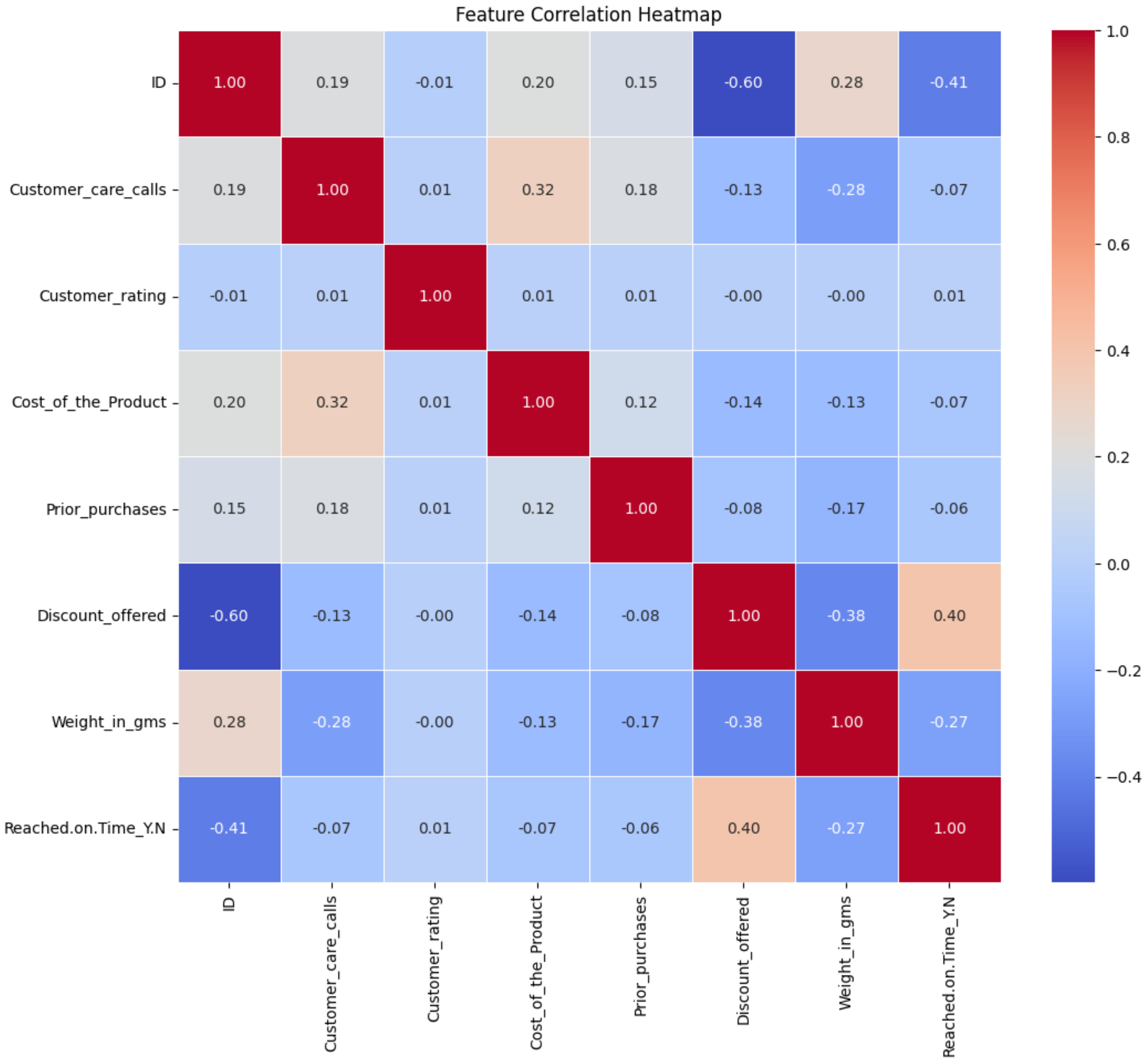

Distribution of Weight_in_gms
Count
1000
800
600
400
200
0
1000
2000
3000
4000
5000
6000
7000
8000
Weight_in_gms
Box Plot of Weight_in_gms
Weight_in_gms
8000
7000
6000
5000
4000
3000
2000
1000


Distribution of Discount_offered
Count
800
600
400
200
0
0
10
20
30
40
50
60
Discount_offered
Box Plot of Discount_offered
Discount_offered
60
50
40
30
20
10
0


Box Plot of Prior_purchases by Reached.on.Time_Y.N
Prior_purchases
10
9
8
7
6
5
4
3
2
0
1
Reached.on.Time_Y.N

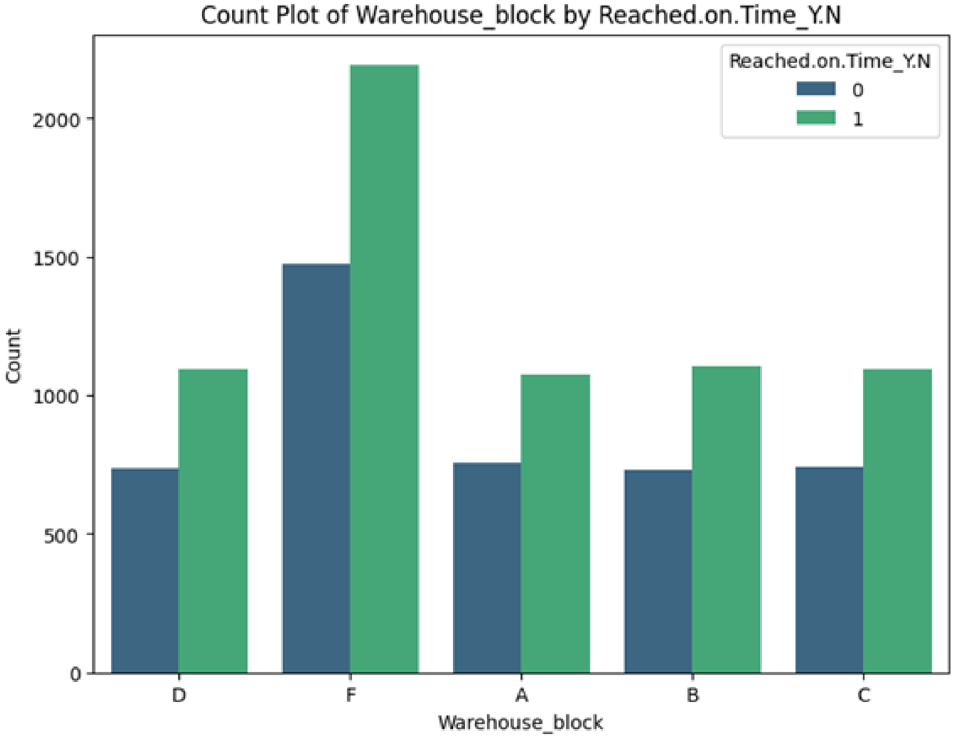

Box Plot of Customer_rating by Reached.on.Time_Y.N
Customer_rating
5.0
4.5
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0
1
Reached.on.Time_Y.N

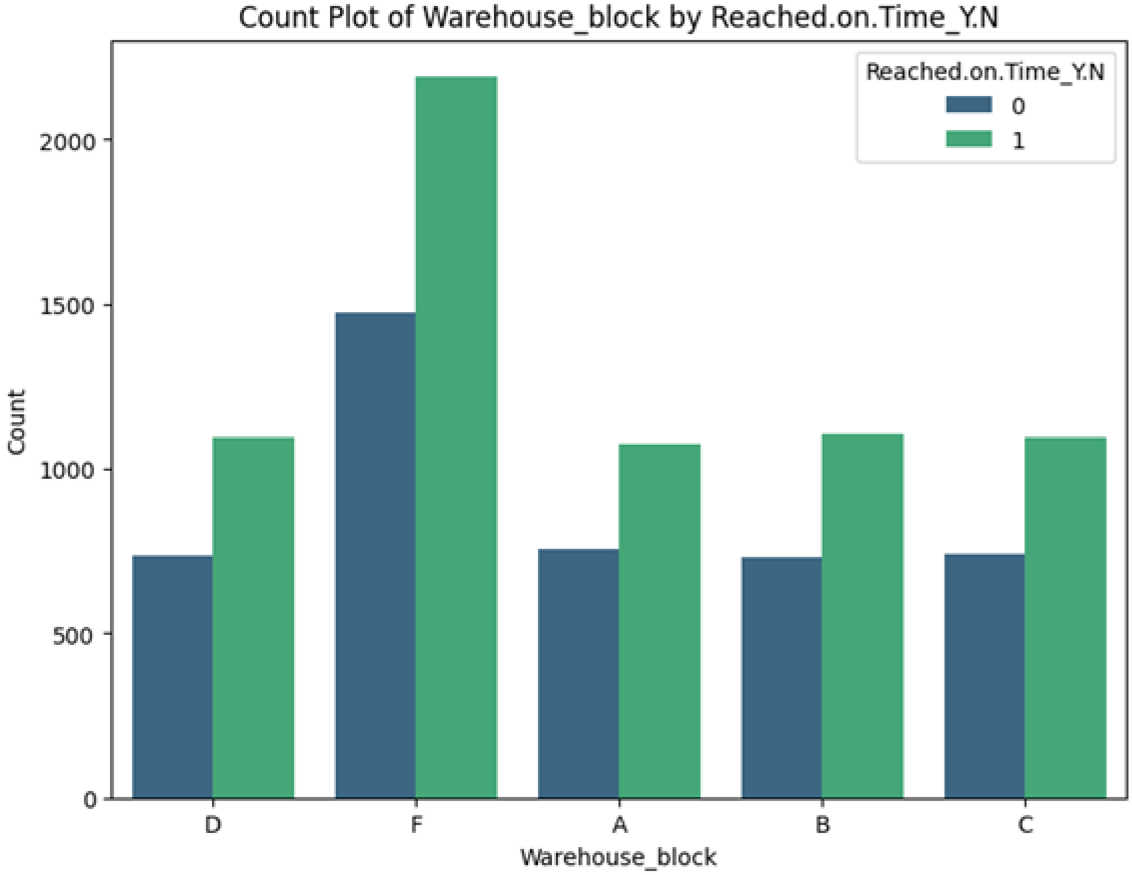
30th September 2025 (Tuesday)	Performed Data Visualization using Matplotlib and Seaborn including histograms, boxplots, scatter plots, and heatmaps for detailed insights.

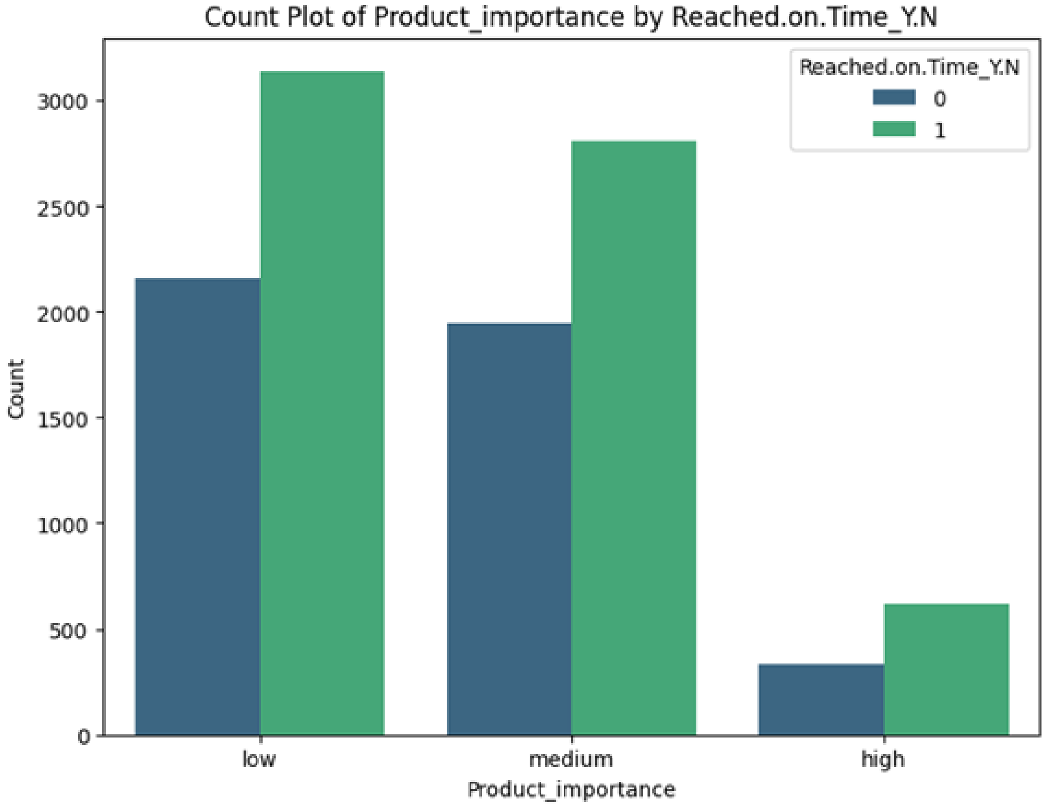


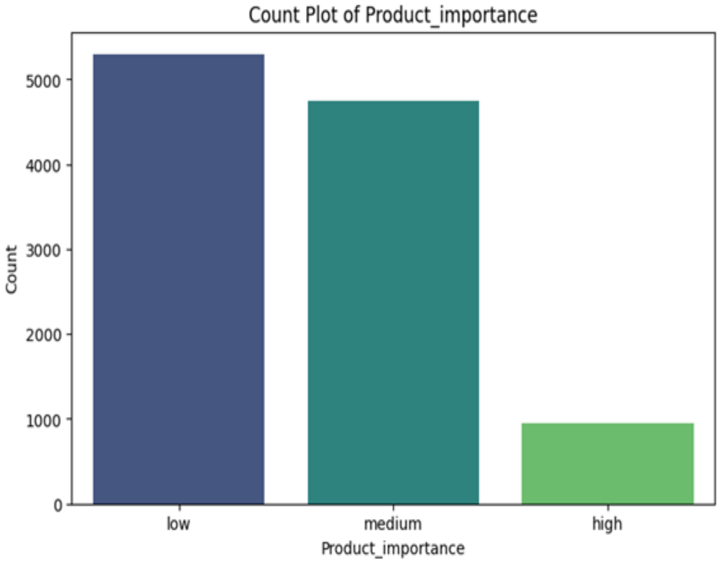
Feature Correlation Heatmap

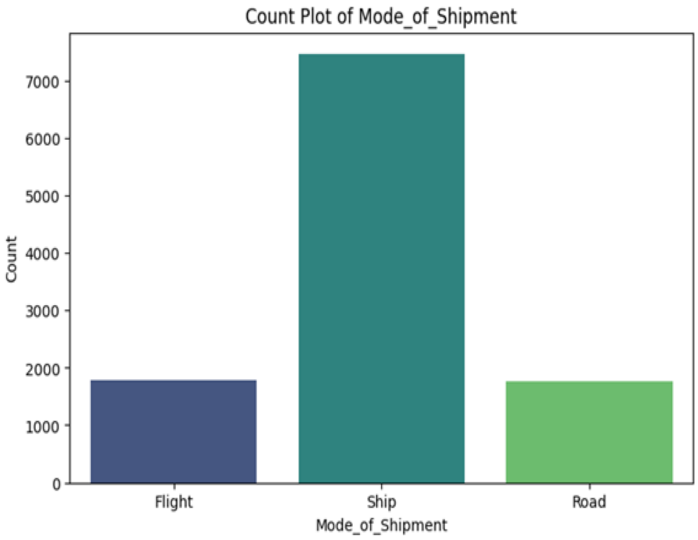
	ID	Customer_care_calls	Customer_rating	Cost_of_the_Product	Prior_purchases	Discount_offered	Weight_in_gms	Reached.on.Time_Y.N
ID	1.00	0.19	-0.01	0.20	0.15	-0.60	0.28	-0.41
Customer_care_calls	0.19	1.00	0.01	0.32	0.18	-0.13	-0.28	-0.07
Customer_rating	-0.01	0.01	1.00	0.01	0.01	-0.00	-0.00	0.01
Cost_of_the_Product	0.20	0.32	0.01	1.00	0.12	-0.14	-0.13	-0.07
Prior_purchases	0.15	0.18	0.01	0.12	1.00	-0.08	-0.17	-0.06
Discount_offered	-0.60	-0.13	-0.00	-0.14	-0.08	1.00	-0.38	0.40
Weight_in_gms	0.28	-0.28	-0.00	-0.13	-0.17	-0.38	1.00	-0.27
Reached.on.Time_Y.N	-0.41	-0.07	0.01	-0.07	-0.06	0.40	-0.27	1.00

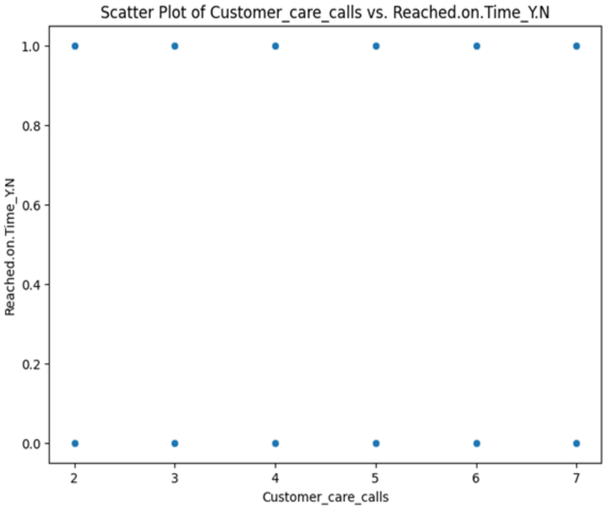

Count Plot of Warehouse_block by Reached.on.Time_Y.N
Reached.on.Time_Y.N
0
1
Count
0
500
1000
1500
2000
D
F
A
B
C
Warehouse_block

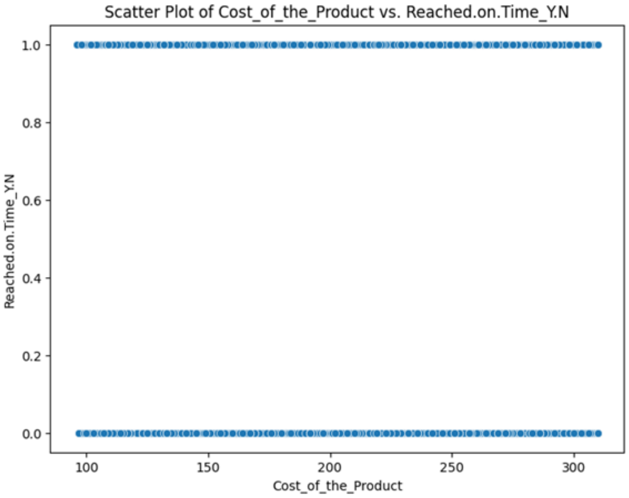

Count Plot of Warehouse_block by Reached.on.Time_Y.N
Reached.on.Time_Y.N
0
1
Count
0
500
1000
1500
2000
D
F
A
B
C
Warehouse_block

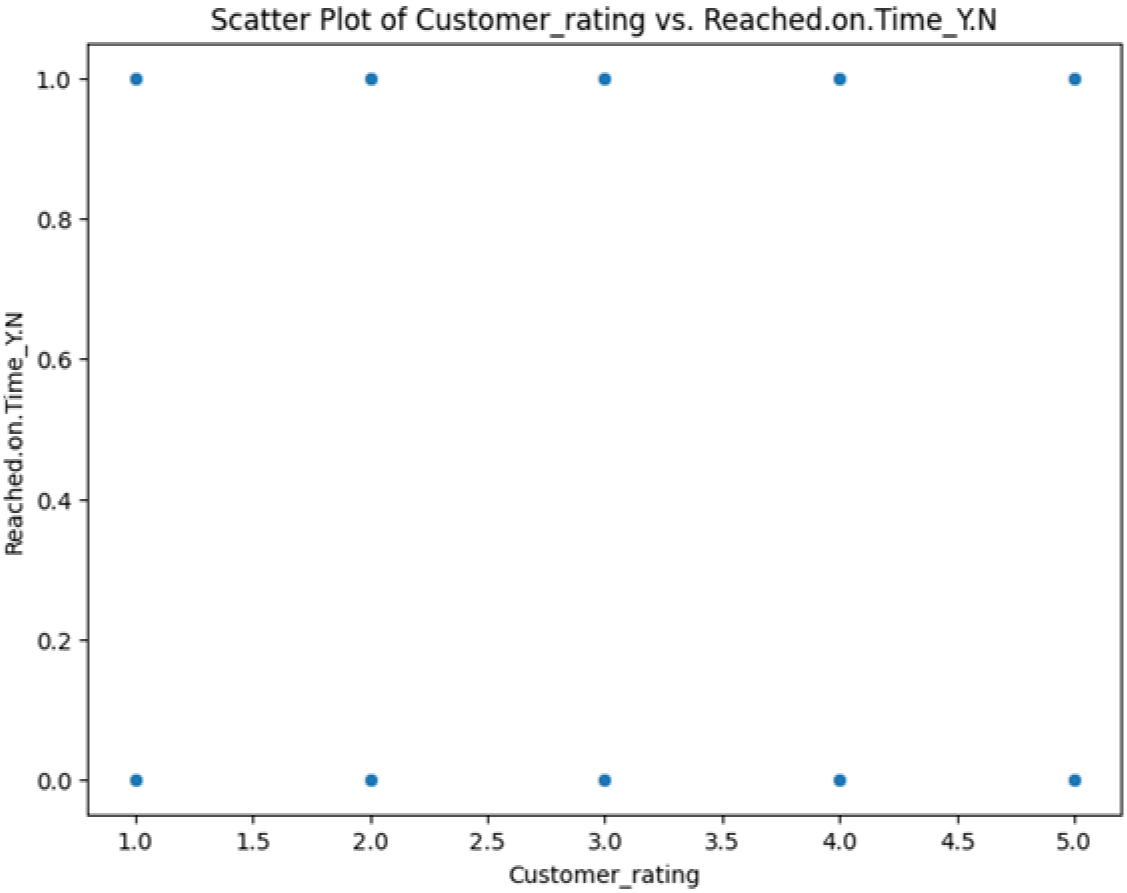

Count Plot of Product_importance by Reached.on.Time_Y.N
Reached.on.Time_Y.N
0
1
3000
2500
2000
1500
1000
500
0
Count
low
medium
high
Product_importance


Count Plot of Product_importance
5000
4000
3000
2000
1000
0
Count
low
medium
high
Product_importance


Count Plot of Mode_of_Shipment
7000
6000
5000
4000
3000
2000
1000
0
Count
Flight
Ship
Road
Mode_of_Shipment


Scatter Plot of Customer_care_calls vs. Reached.on.Time_Y.N
Reached.on.Time_Y.N
1.0
0.8
0.6
0.4
0.2
0.0
2
3
4
5
6
7
Customer_care_calls


Scatter Plot of Cost_of_the_Product vs. Reached.on.Time_Y.N
Reached.on.Time_Y.N
1.0
0.8
0.6
0.4
0.2
0.0
100
150
200
250
300
Cost_of_the_Product


Scatter Plot of Customer_rating vs. Reached.on.Time_Y.N
Reached.on.Time_Y.N
1.0
0.8
0.6
0.4
0.2
0.0
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
Customer_rating

1st October
2025
(Wednesday)

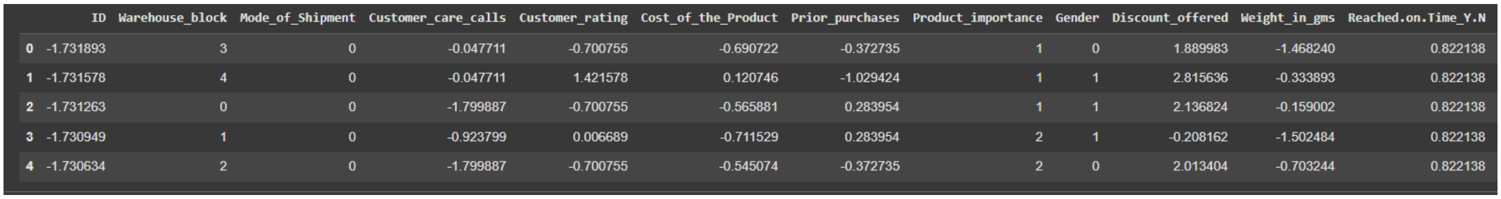
Addressed class imbalance problem in the target variable (reached_on_time) by applying oversampling to balance 'On Time' and 'Not On Time' classes.

OUTPUT:

```
Class distribution for 'Reached.on.Time_Y.N':
Reached.on.Time_Y.N
1 59.669061
0 40.330939
```

3rd October
2025
(Friday)

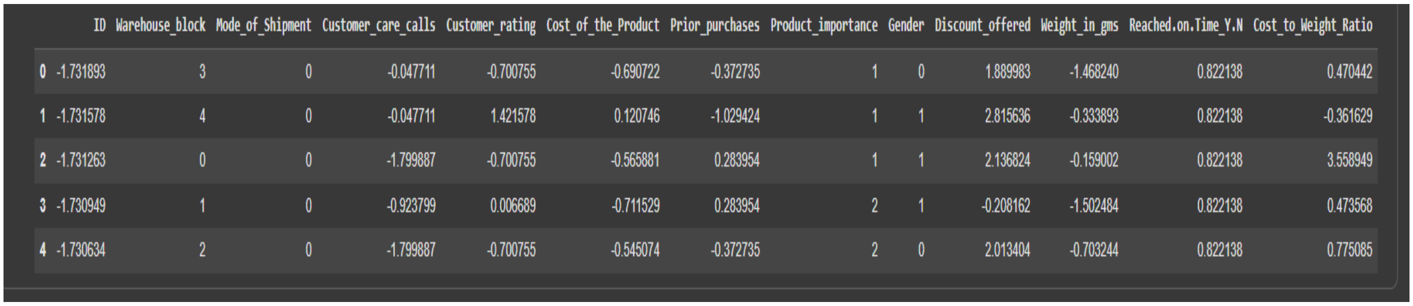
- Began encoding of categorical features in the dataset.
- Used **Label Encoding** technique to convert non-numeric categories (e.g., Warehouse_block, Mode_of_Shipment, Product_importance, Gender) into numerical values.



	ID	Warehouse_block	Mode_of_Shipment	Customer_care_calls	Customer_rating	Cost_of_the_Product	Prior_purchases	Product_importance	Gender	Discount_offered	Weight_in_gms	Reached.on.Time_Y.N
0	-1.731893	3	0	-0.047711	-0.700755	-0.690722	-0.372735	1	0	1.889983	-1.468240	0.822138
1	-1.731578	4	0	-0.047711	1.421578	0.120746	-1.029424	1	1	2.815636	-0.333893	0.822138
2	-1.731263	0	0	-1.799887	-0.700755	-0.565881	0.283954	1	1	2.136824	-0.159002	0.822138
3	-1.730949	1	0	-0.923799	0.006689	-0.711529	0.283954	2	1	-0.208162	-1.502484	0.822138
4	-1.730634	2	0	-1.799887	-0.700755	-0.545074	-0.372735	2	0	2.013404	-0.703244	0.822138

6th October 2025 (Monday)

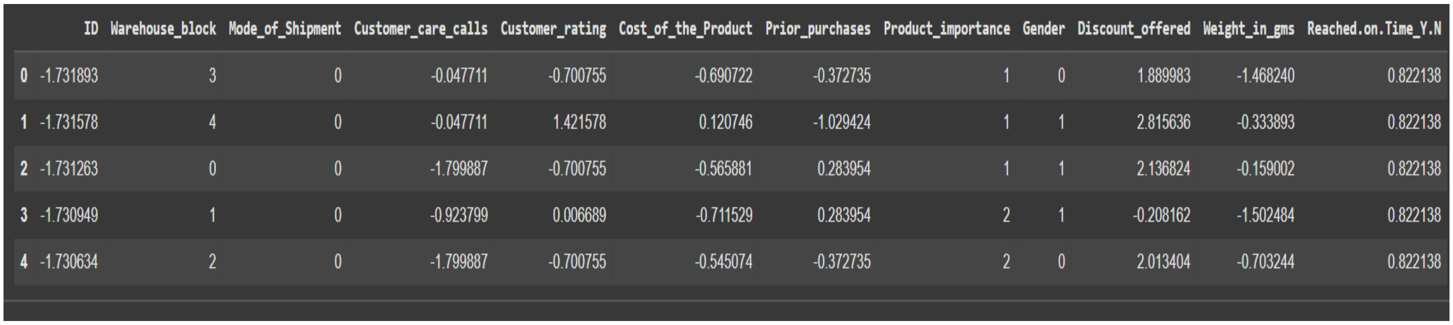
- Conducted Feature Engineering to create additional meaningful insights from existing attributes.
- Added a new derived feature: **Cost-to-Weight Ratio**, calculated by dividing Cost_of_the_Product by Weight_in_gms.



	ID	Warehouse_block	Mode_of_Shipment	Customer_care_calls	Customer_rating	Cost_of_the_Product	Prior_purchases	Product_importance	Gender	Discount_offered	Weight_in_gms	Reached.on.Time_Y.N	Cost_to_Weight_Ratio
0	-1.731893	3	0	-0.047711	-0.700755	-0.690722	-0.372735	1	0	1.889983	-1.468240	0.822138	0.470442
1	-1.731578	4	0	-0.047711	1.421578	0.120746	-1.029424	1	1	2.815636	-0.333893	0.822138	-0.361629
2	-1.731263	0	0	-1.799887	-0.700755	-0.565881	0.283954	1	1	2.136824	-0.159002	0.822138	3.558949
3	-1.730949	1	0	-0.923799	0.006689	-0.711529	0.283954	2	1	-0.208162	-1.502484	0.822138	0.473568
4	-1.730634	2	0	-1.799887	-0.700755	-0.545074	-0.372735	2	0	2.013404	-0.703244	0.822138	0.775085

7th October 2025 (Tuesday)

- Applied **Normalization** to numerical features using **StandardScaler** to standardize the data distribution.
- Scaled features like Cost_of_the_Product, Discount_offered, and Weight_in_gms to have zero mean and unit variance.



	ID	Warehouse_block	Mode_of_Shipment	Customer_care_calls	Customer_rating	Cost_of_the_Product	Prior_purchases	Product_importance	Gender	Discount_offered	Weight_in_gms	Reached.on.Time_Y.N
0	-1.731893	3	0	-0.047711	-0.700755	-0.690722	-0.372735	1	0	1.889983	-1.468240	0.822138
1	-1.731578	4	0	-0.047711	1.421578	0.120746	-1.029424	1	1	2.815636	-0.333893	0.822138
2	-1.731263	0	0	-1.799887	-0.700755	-0.565881	0.283954	1	1	2.136824	-0.159002	0.822138
3	-1.730949	1	0	-0.923799	0.006689	-0.711529	0.283954	2	1	-0.208162	-1.502484	0.822138
4	-1.730634	2	0	-1.799887	-0.700755	-0.545074	-0.372735	2	0	2.013404	-0.703244	0.822138

10th October
2025
(Friday)

- Executed Train-Test Split to prepare the dataset for modeling.

- Used Scikit-learn’s **train_test_split()** to divide the data into **80%** training and **20%** testing subsets.

OUTPUT :

```
Data split complete!

Training size: (8799, 11)

Testing size: (2200, 11)
```

Progress so far :

The key milestones completed so far are as follows:

- ✓ **Dataset Collection and Exploration:**
  A suitable dataset related to shipment and delivery prediction was identified and downloaded from Kaggle. The dataset contained over 10,000 rows with multiple shipment-related attributes such as product cost, weight, shipment mode, and customer ratings. Initial exploration was performed to understand data structure, types, and overall quality.
- ✓ **Data Cleaning and Preprocessing:**
  Missing values in both numerical and categorical columns were systematically handled.
  - Numerical columns were imputed using mean values.
  - Categorical columns were imputed using mode values.
    Additionally, duplicate checks and consistency validations were carried out to ensure data reliability.
- ✓ **Exploratory Data Analysis (EDA):**
  Conducted univariate and bivariate analysis to understand feature distributions and relationships with the target variable (`Reached.on.Time_Y.N`).
  Various visualization tools like **Matplotlib** and **Seaborn** were used to generate histograms, boxplots, scatter plots, and correlation heatmaps for identifying patterns and dependencies.
- ✓ **Feature Engineering:**
  Created an additional derived feature — **Cost-to-Weight Ratio** — to capture the relationship between shipment cost and product weight, providing deeper insights into delivery efficiency.
- ✓ **Feature Encoding and Normalization:**
  Applied **Label Encoding** to categorical features like `Warehouse_block`, `Mode_of_Shipment`, and `Product_importance` for model compatibility.
  Standardized numerical features using **StandardScaler** to bring all data to a uniform scale, improving model accuracy and performance.
- ✓ **Train-Test Split:**
  Divided the dataset into **training (80%)** and **testing (20%)** subsets using Scikit-learn's `train_test_split()` function, ensuring proper distribution and randomness for unbiased model evaluation.

References

[1] Kaggle Dataset: 'Shipment Delivery Status Prediction Dataset', 2023.

[2] John et al., 'Predictive Modeling for Delivery Timeliness', IEEE, 2022.

[3] Smith et al., 'AI in Logistics Optimization', Elsevier, 2023.