CSE422: Artificial Intelligence

Project Report

Group: 12 Section: 1

Prepared by:

Fahmida Afrin- 23201306

Table Of Contents

Introduction

In the rapidly expanding e-commerce industry, on-time delivery of shipments plays a vital role in ensuring customer satisfaction and overall business performance. This project examines various operational features, including customer service calls, product pricing, shipment weight, and applied discounts, to predict whether deliveries will be completed on schedule. Through data analysis and machine learning techniques, the study aims to determine the key factors influencing delivery outcomes, identify inefficiencies within logistics operations, and generate practical insights that can assist e-commerce companies in improving their delivery systems and service quality.

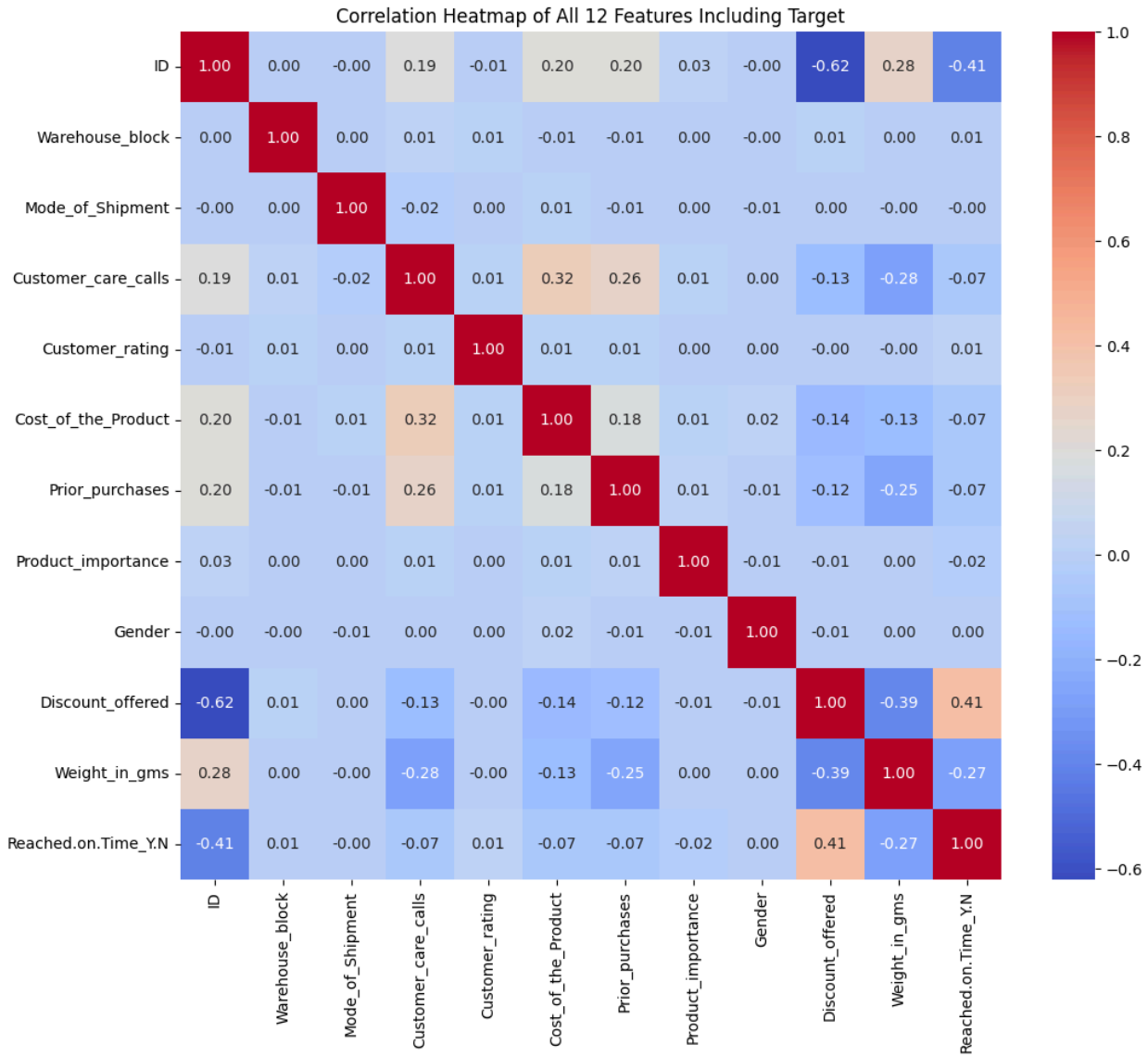
Dataset Description

The dataset used in this study consists of 11 independent features, excluding the target variable, and includes a total of 10,999 observations. Each record corresponds to an individual shipment in an e-commerce logistics system.

The problem addressed in this project is a **classification task**, as the target variable *Reached.on.Time_Y.N* is categorical in nature. It indicates whether a shipment was delivered on time (coded as 1) or delivered late (coded as 0). The primary objective is to predict this delivery outcome based on the available shipment and customer-related features.

The dataset contains a combination of numerical and categorical variables. As most machine learning models require numerical inputs, categorical variables were transformed into numerical representations through Label encoding techniques prior to model training.

To explore relationships between variables, a correlation analysis was performed.


Correlation Heatmap of All 12 Features Including Target
ID 1.00 0.00 -0.00 0.19 -0.01 0.20 0.20 0.03 -0.00 -0.62 0.28 -0.41
Warehouse_block 0.00 1.00 0.00 0.01 0.01 -0.01 -0.01 0.00 -0.00 0.01 0.00 0.01
Mode_of_Shipment -0.00 0.00 1.00 -0.02 0.00 0.01 -0.01 0.00 -0.01 0.00 -0.00 -0.00
Customer_care_calls 0.19 0.01 -0.02 1.00 0.01 0.32 0.26 0.01 0.00 -0.13 -0.28 -0.07
Customer_rating -0.01 0.01 0.00 0.01 1.00 0.01 0.01 0.00 0.00 -0.00 -0.00 0.01
Cost_of_the_Product 0.20 -0.01 0.01 0.32 0.01 1.00 0.18 0.01 0.02 -0.14 -0.13 -0.07
Prior_purchases 0.20 -0.01 -0.01 0.26 0.01 0.18 1.00 0.01 -0.01 -0.12 -0.25 -0.07
Product_importance 0.03 0.00 0.00 0.01 0.00 0.01 0.01 1.00 -0.01 -0.01 0.00 -0.02
Gender -0.00 -0.00 -0.01 0.00 0.00 0.02 -0.01 -0.01 1.00 -0.01 0.00 0.00
Discount_offered -0.62 0.01 0.00 -0.13 -0.00 -0.14 -0.12 -0.01 -0.01 1.00 -0.39 0.41
Weight_in_gms 0.28 0.00 -0.00 -0.28 -0.00 -0.13 -0.25 0.00 0.00 -0.39 1.00 -0.27
Reached.on.Time_Y.N -0.41 0.01 -0.00 -0.07 0.01 -0.07 -0.07 -0.02 0.00 0.41 -0.27 1.00
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
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6

Key Correlation Statistics from the Heatmap:

Feature	Correlation with Target
Discount_offered	+0.41
Product_weight_gms	-0.27
Cost_of_the_Product	-0.07
Customer_care_calls	-0.07
Customer_rating	0.01
Prior_purchases	-0.07
Warehouse_block	0.01
Mode_of_Shipment	-0.0
Product_importance	-0.02
Gender	+0.00
ID	-0.41

The table above indicates that **Discount_offered** has the strongest positive correlation with the target variable, while **ID** exhibits a strong negative correlation. **Product_weight_gms** also shows a moderate negative correlation with on-time delivery. All remaining features display weak or negligible correlations, with values close to zero. This suggests that, apart from **Discount_offered**, **Product_weight_gms**, and **ID**, most variables are not strongly linearly related to delivery timeliness and may have limited individual predictive power.

Imbalanced Dataset

```
Class distribution:
Reached.on.Time_Y.N
1    6563
0    4436
Name: count, dtype: int64
```

There is class imbalance in the dataset.

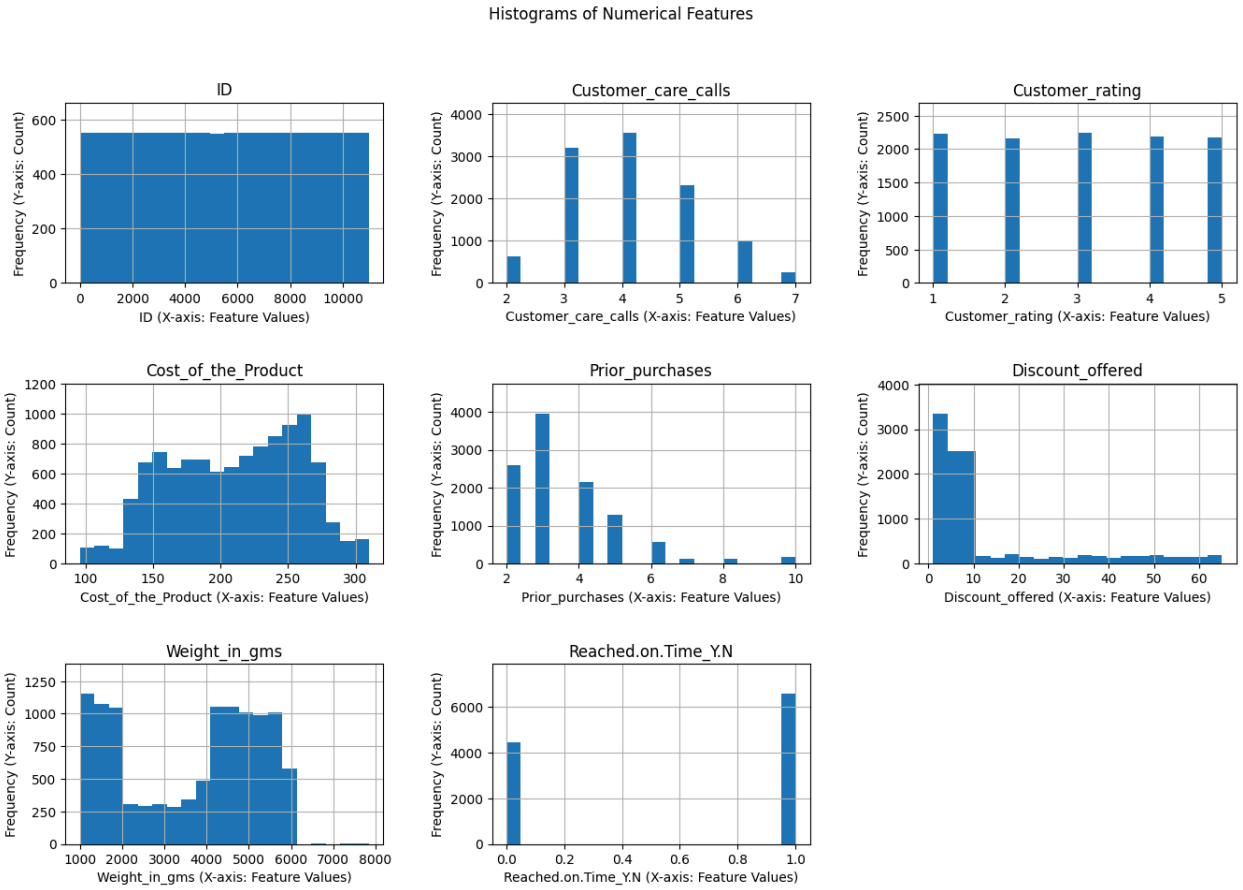
Exploratory Data Analysis

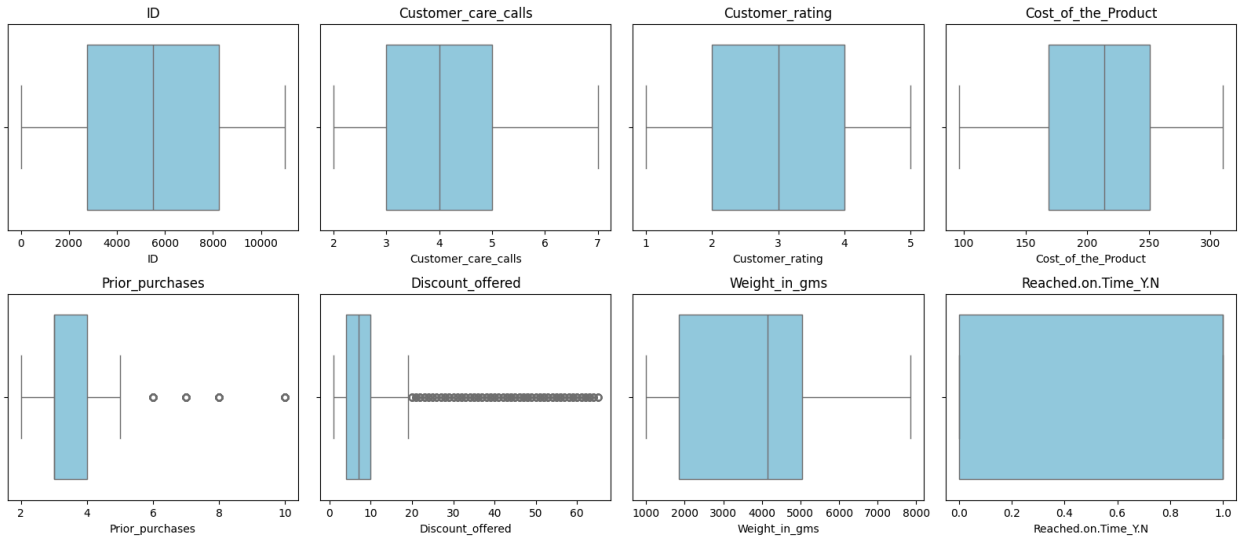
Exploratory Data Analysis (EDA) was conducted to identify underlying patterns and relationships within the dataset. Several notable observations emerged from this analysis:

- Shipments with higher discount values and greater shipment weights tend to experience a higher likelihood of delayed delivery.
- On-time delivery rates vary across different warehouse blocks and modes of shipment.

- Late deliveries are associated with a slightly higher number of customer care calls, suggesting potential customer dissatisfaction.

These insights are supported by multiple visualizations, including boxplots, histograms.





Insights:

- Outliers are clearly observed in Discount_offered and Prior_purchases (appearing as dots beyond the whiskers).
- Other features do not exhibit notable outliers.
- Decision: Apply capping or transformation to outliers only in Discount_offered and Prior_purchases; scale the remaining features as necessary.

Dataset pre-processing

Effective data pre-processing is critical to ensure the quality, consistency, and reliability of machine learning models. The following steps were carried out to address common data issues and prepare the dataset for analysis:

1. Handling Null / Missing Values

The dataset was carefully examined for missing or null values. No missing values were found in any of the features; hence, no imputation or deletion of rows/columns was necessary.

2. Categorical Variables

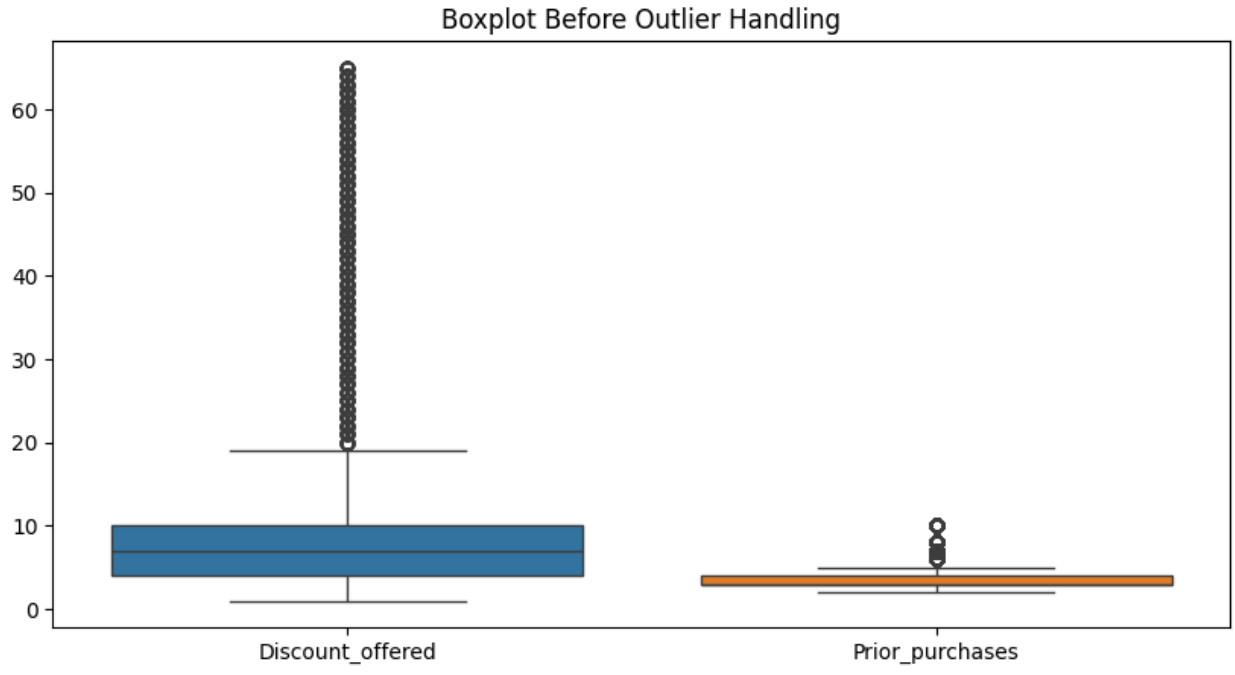
Several features, including Warehouse_block, Mode_of_Shipment, Product_importance, and Gender, are categorical. Since most machine learning algorithms require numerical input, these variables were converted into numeric form using **Label Encoding**, which assigns a unique integer to each category. This ensured compatibility with the modeling algorithms.

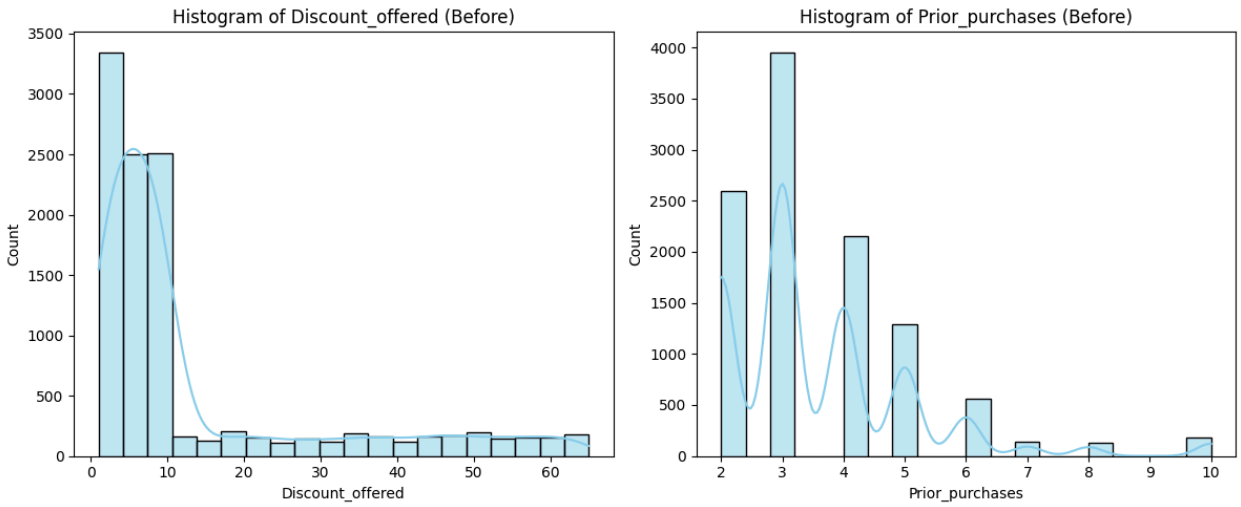
3. Feature Scaling

The dataset contains features with varying scales and units (e.g., Cost_of_the_Product, Weight_in_gms, Discount_offered). To ensure that all features contribute equally to the models and to improve the performance of algorithms sensitive to feature scale (such as KMeans and neural networks), MinMaxScaler was applied.

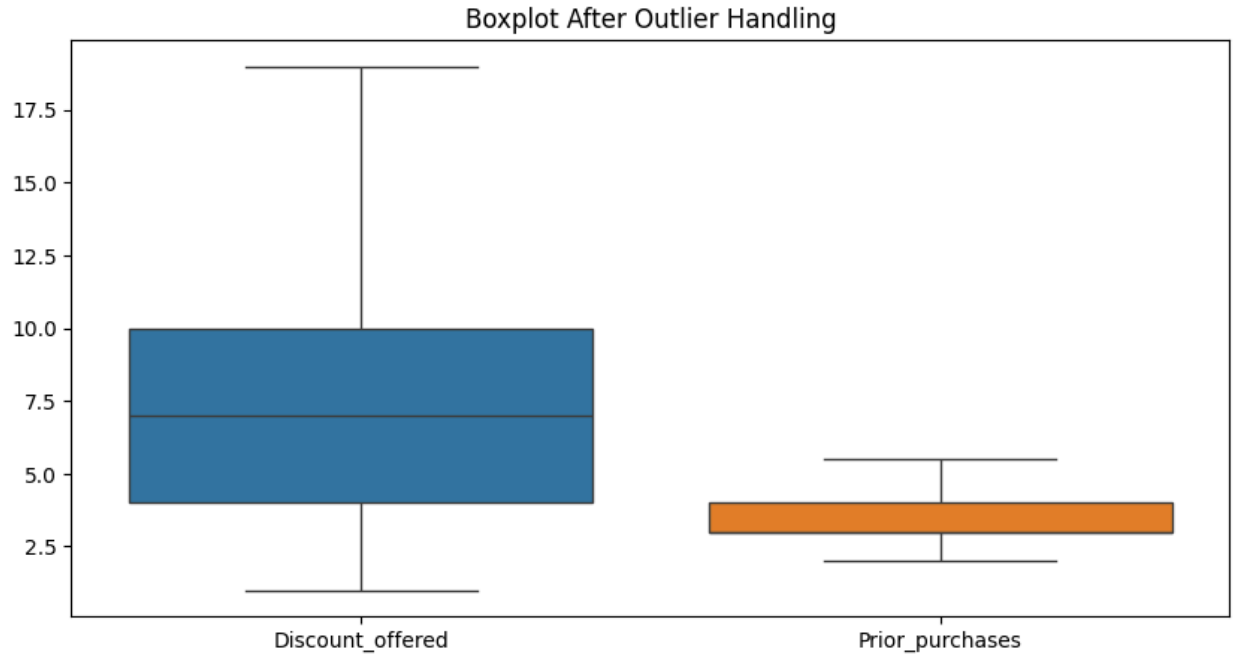
4. Outlier Treatment

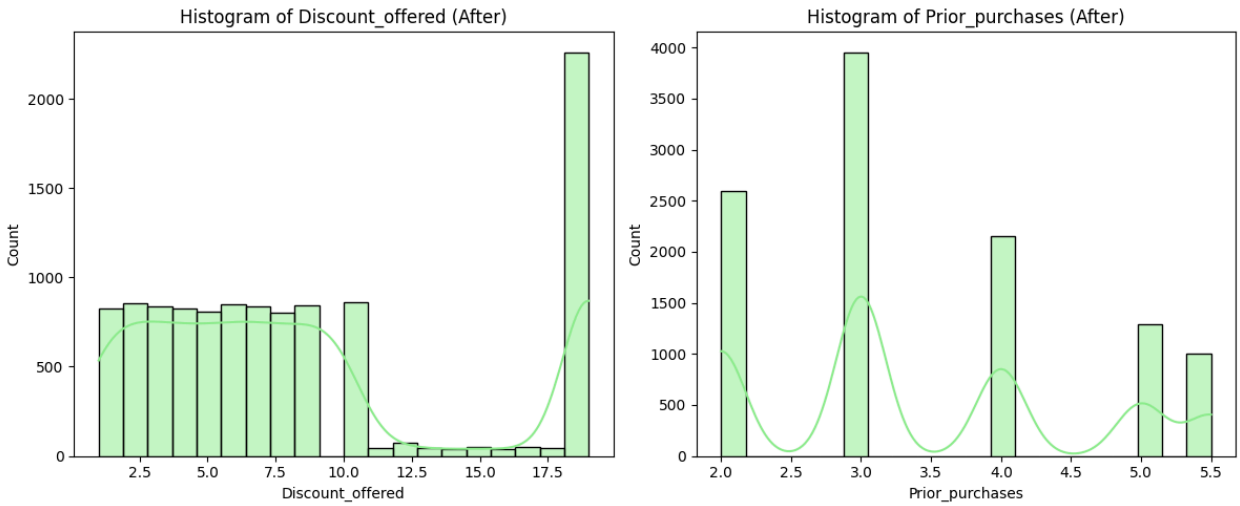
Outliers were observed in Discount_offered and Prior_purchases (visible as points beyond the whiskers in boxplots).





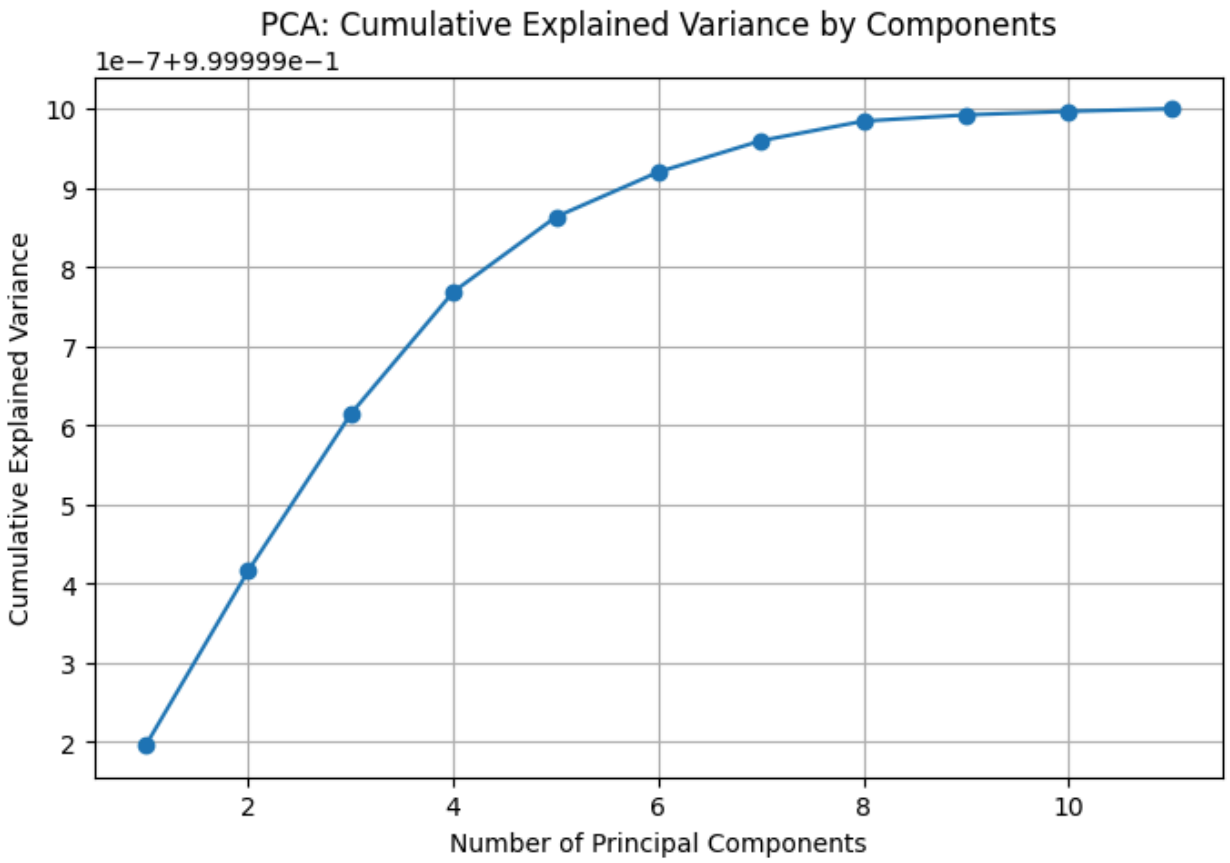
To minimize their impact on model performance, capping was applied to these features, while other features were left unchanged.





5. Dimensionality Reduction (PCA)

To facilitate visualization and better understand the underlying structure of the dataset, **Principal Component Analysis (PCA)** was applied. PCA reduces the dataset to a smaller set of principal components while retaining most of the variance. This technique aided in visualizing clusters formed by KMeans and interpreting relationships among features in a lower-dimensional space.



6. Oversampling

- Over Sampling was done to handle class imbalance

Summary of Pre-processing Steps

- No missing values were detected; no imputation or deletion was required.
- Categorical variables were converted to numeric form using label encoding.
- Features were standardized to ensure comparability and optimal model performance.
- Outliers in Discount_offered and Prior_purchases were treated.
- PCA was applied for dimensionality reduction and cluster visualization.
- Over Sampling was done to handle class imbalance

These pre-processing steps ensured that the dataset was clean, consistent, and well-prepared for subsequent analysis, modeling, and interpretation.

Dataset splitting

Train-Test Split

The dataset was divided into training and test sets, with **80% of the data used for training** and **20% for testing**. A **stratified split** was applied to ensure that the proportion of each class in the target variable was preserved in both sets.

Model Training & Testing (Supervised)

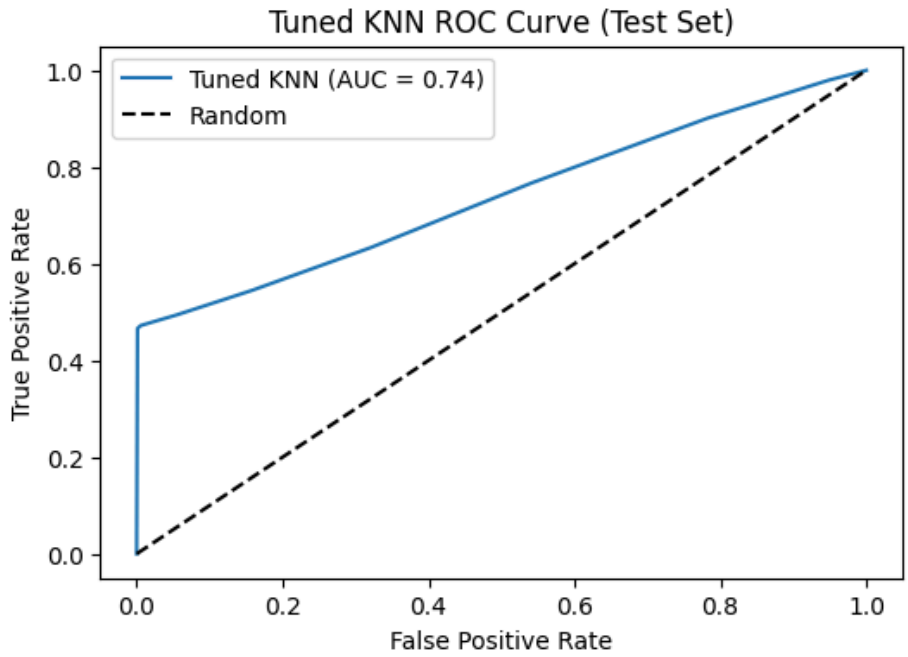
A variety of supervised machine learning models were trained and evaluated to predict **on-time delivery**. Model performance was assessed using **accuracy, precision, recall**, and other relevant metrics.

1. K-Nearest Neighbors (KNN)

Initial Parameters: n_neighbors = 5 (default), weights = 'uniform'
Hyperparameter Tuning: Grid search over n_neighbors (1–20), weights ('uniform', 'distance')
Best Parameters: n_neighbors = 8, weights = 'uniform'
Performance:

Comparison of KNN Performance Before and After Tuning:

Metric	Default KNN	Tuned KNN
Training Accuracy	0.8066	0.7787
Test Accuracy	0.6532	0.6645
CV Accuracy (mean ± std)	0.7201 ± 0.0226	0.7350 ± 0.0170
CV F1-score (mean ± std)	0.7201 ± 0.0226	0.7266 ± 0.0161



2. Decision Tree

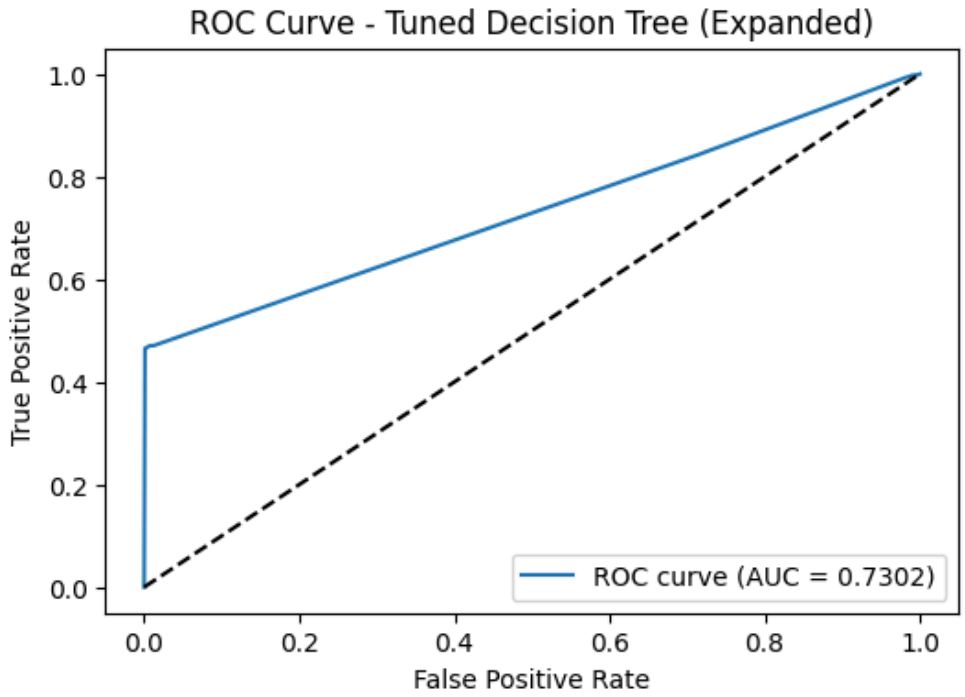
Training Accuracy: 0.694

Validation Accuracy (CV mean ± std): 0.691 ± 0.010

Test Accuracy: 0.680

Best Parameters: criterion = 'gini', max_depth = 5, max_features = None, min_samples_leaf = 3, min_samples_split = 2

Best Cross-Validated Accuracy: 0.691

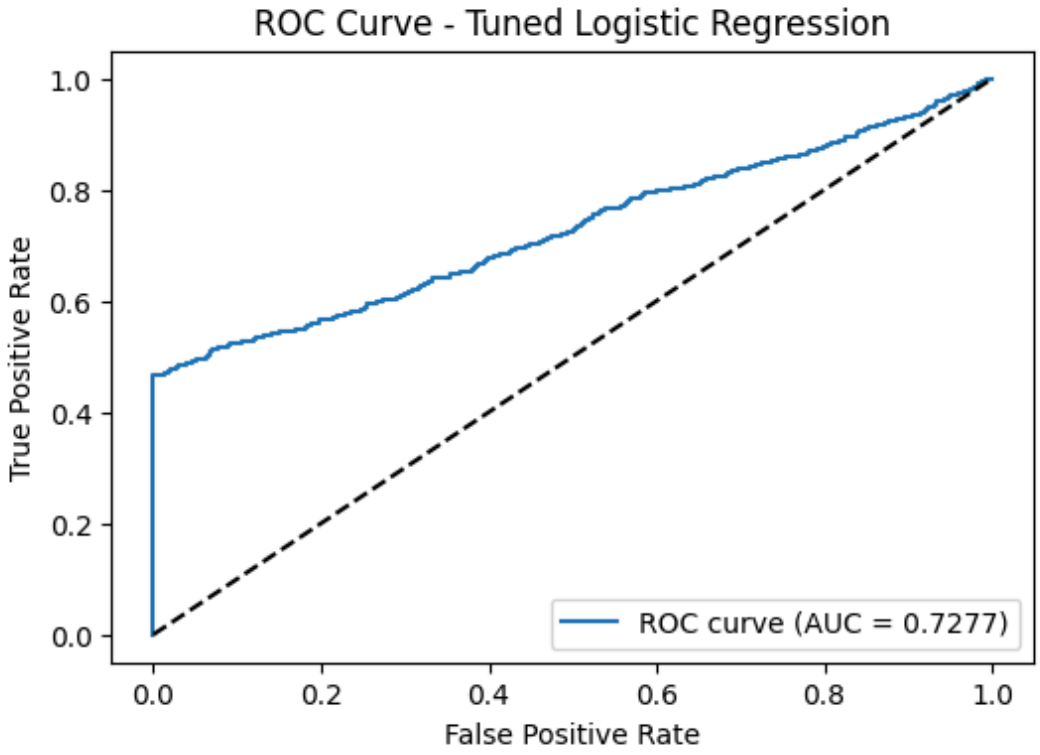


3. Logistic Regression

Performance:

Metric	Before Tuning	After Tuning
Training Accuracy	0.6581	0.6655
Validation Accuracy	0.6581 ± 0.0027	0.6655 ± 0.0035
Test Accuracy	0.6491	0.6468

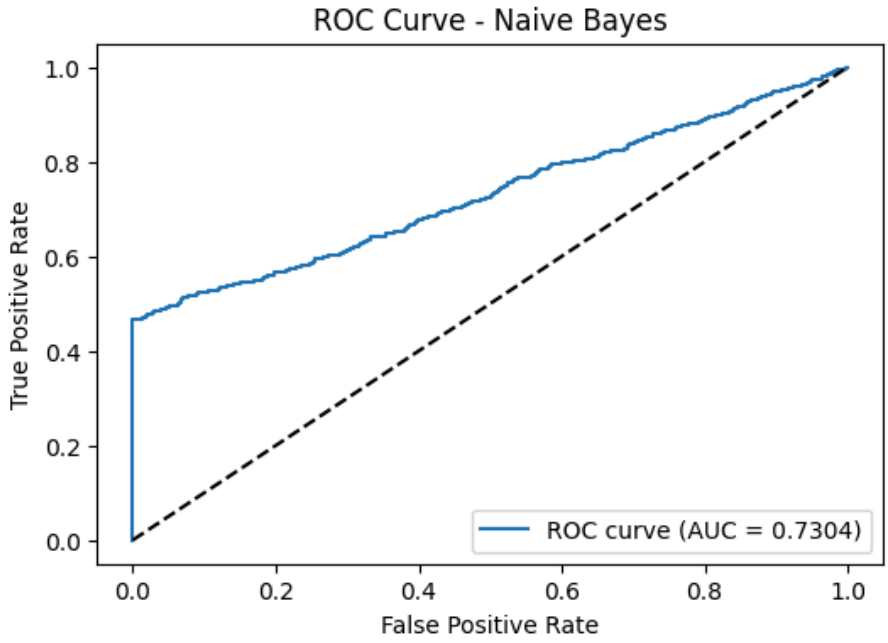
- **Before Tuning:** Default hyperparameters (C=1.0, penalty='l2', solver='lbfgs').
- **After Tuning:** Best parameters from GridSearchCV (C=0.01, penalty='l2', solver='liblinear').



4. Naive Bayes

Initial Parameters: GaussianNB
Performance:

- Training accuracy: 0.65
- Test accuracy: 0.64
- Precision (Class 1): 0.73
- Recall (Class 1): 0.65



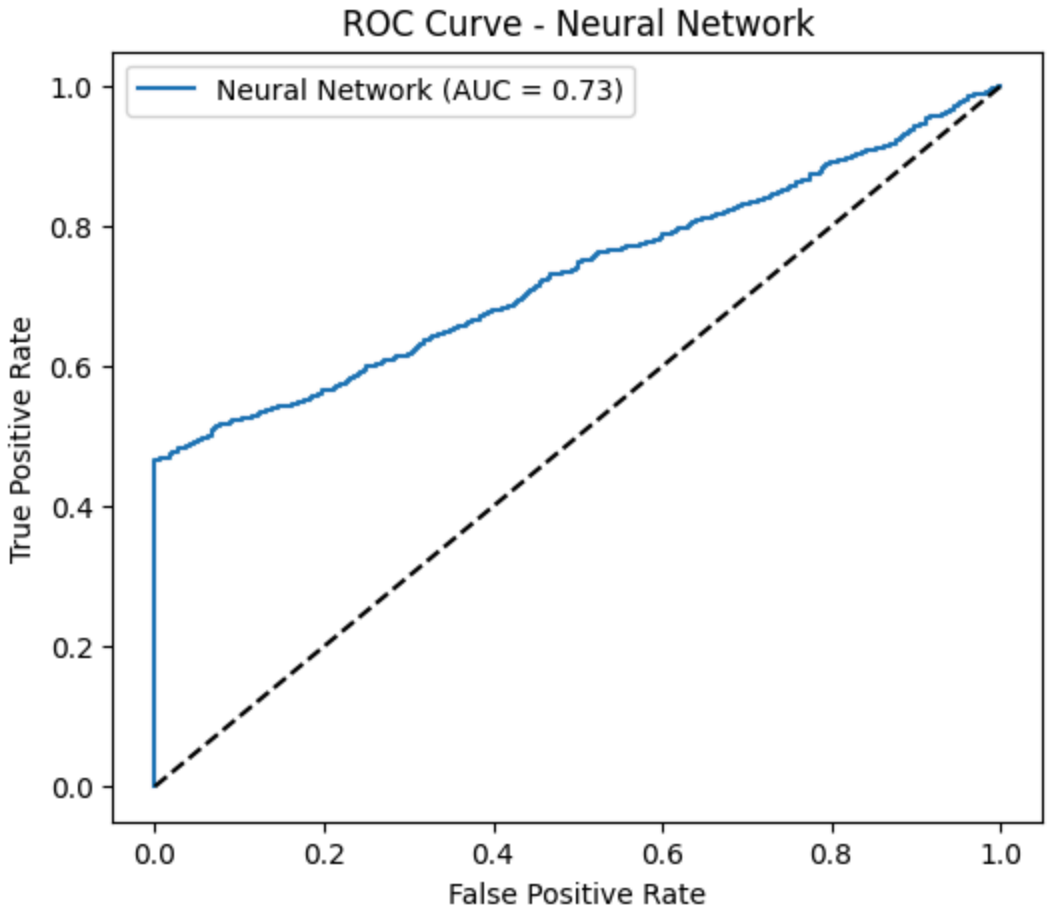
5. Neural Network (MLPClassifier)

Neural Network Model Results Overview

Model Version	Training Accuracy	Validation Accuracy (CV mean ± std)	Test Accuracy
Default Parameters	0.6830	0.6613 ± 0.0232	0.6768
Tuned Parameters	0.6745	0.6825 ± 0.0091	0.6600

- **Default model** had higher training and test accuracy, but lower cross-validation accuracy, suggesting mild overfitting.
- **Tuned model** (with GridSearchCV, early stopping, and patience) improved cross-validation accuracy and reduced overfitting, but test accuracy dropped slightly.
- The best parameters found were: activation='relu', alpha=0.0001, hidden_layer_sizes=(100,), max_iter=200, solver='adam'.

- **After tuning and retraining with best parameters:**
  - Training Accuracy: 0.6745
  - Cross-Validation Accuracy: 0.6825 ± 0.0091
  - Test Accuracy: 0.6600

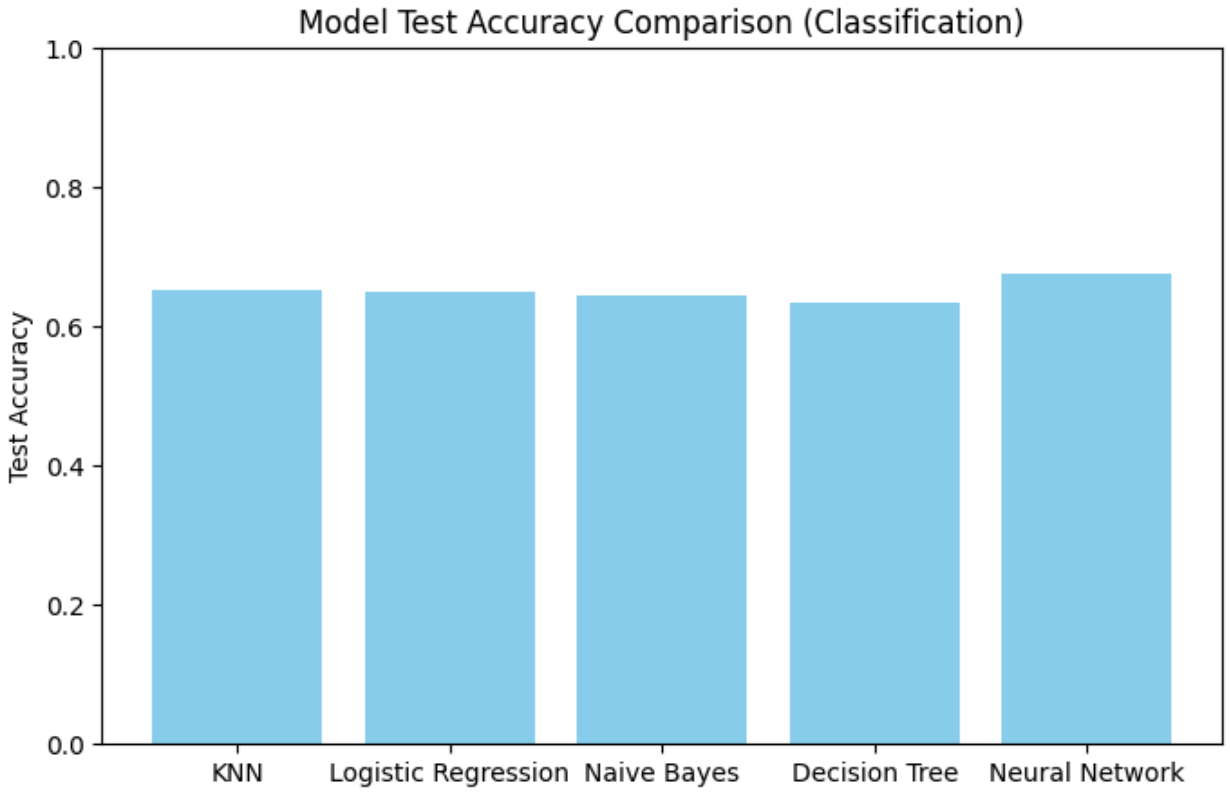


Model Selection & Comparison Analysis

To evaluate and compare the performance of all classification models, several metrics and visualizations were used:

1. Bar Chart of Prediction Accuracy

A bar chart was created to compare the **test accuracy** of all models (KNN, Decision Tree, Logistic Regression, Naive Bayes, Neural Network). Accuracies ranged from **0.64 to 0.67**, with the neural network achieving the highest accuracy.



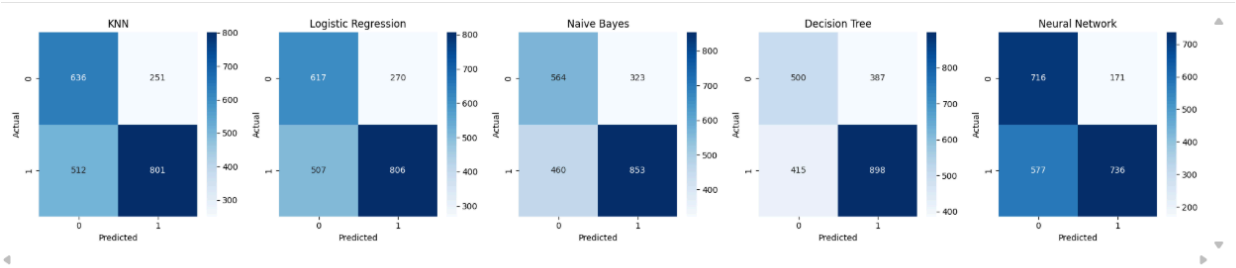
2. Precision and Recall Comparison

A summary table compared the **precision** and **recall** for the positive class (on-time delivery). Values were almost similar across models, indicating consistent but moderate performance in correctly identifying on-time shipments.

	Model	Precision (Class 1)	Recall (Class 1)
0	KNN	0.84	0.55
1	Logistic Regression	0.81	0.60
2	Naive Bayes	0.79	0.58
3	Decision Tree	0.80	0.59
4	Neural Network	0.90	0.52

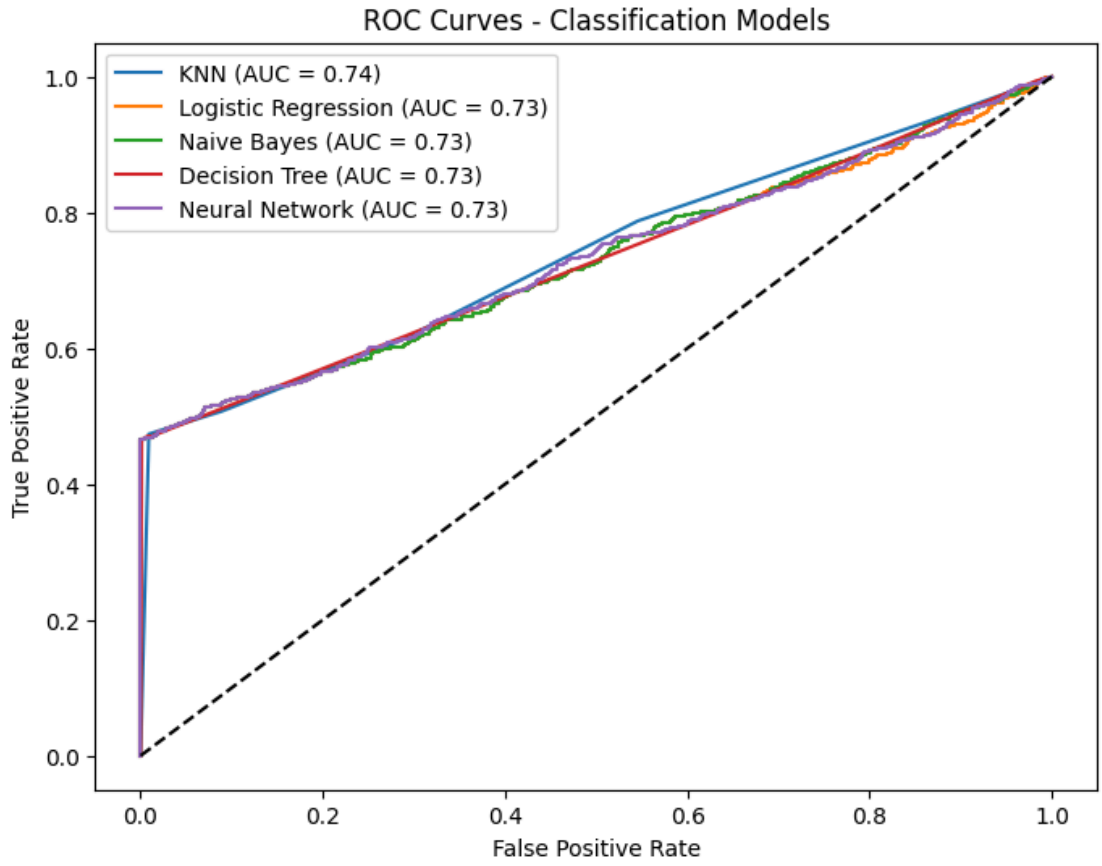
3. Confusion Matrices

Confusion matrices were plotted for each model to visualize **true positives, true negatives, false positives, and false negatives**. All models struggled more with the minority class (on-time deliveries), reflecting the dataset's imbalance.



4. AUC Score and ROC Curves

ROC curves and **AUC scores** were calculated for each model.



Why Model Accuracy is Limited

Despite applying several classification models (KNN, Logistic Regression, Naive Bayes, Decision Tree, Neural Network), the **test accuracies remain similar and moderate**. This indicates that the limitations are due to the data itself rather than the choice of algorithm. Possible reasons include:

- The features may not contain enough predictive information to fully separate the classes.
- Noise, overlap, or hidden factors in the dataset may prevent accurate classification.
- The problem may be inherently difficult, with classes that are not well-separated.

A simple experiment shows that even a highly complex model (Random Forest) cannot achieve significantly higher accuracy, confirming that the **data is the primary limiting factor**.

```
Random Forest Test Accuracy: 0.635

Best Previous Test Accuracy: 0.677
```

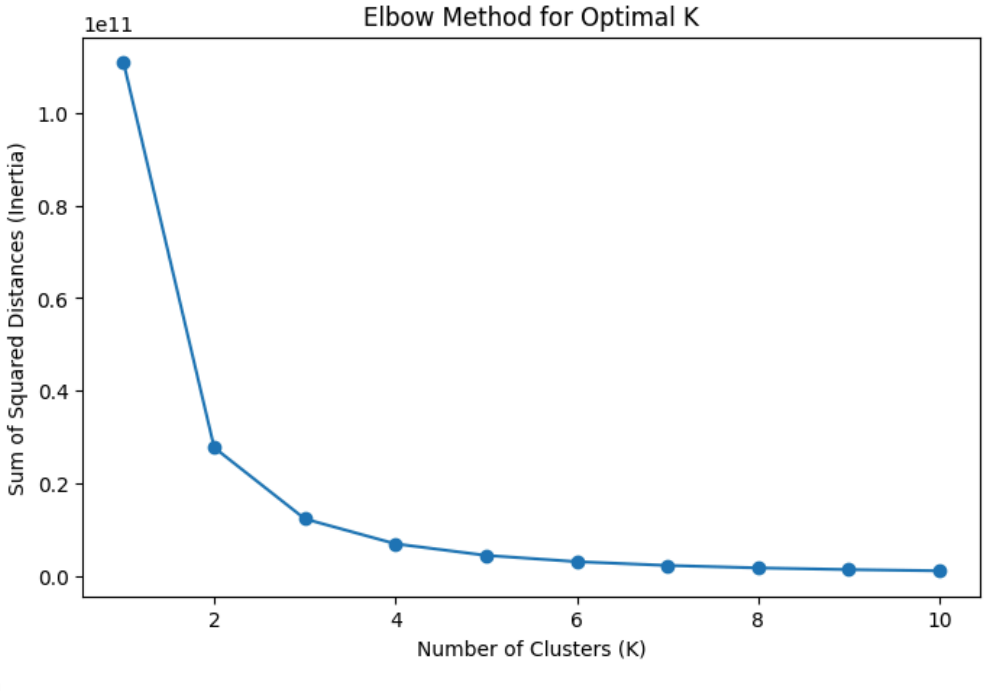
Unsupervised

KMeans Clustering Summary

KMeans clustering was applied to the dataset to identify natural groupings among shipments based on operational features.

1. Optimal Number of Clusters

The **elbow method** was used to determine the optimal number of clusters. The plot indicated that **K = 2** is appropriate for this dataset.

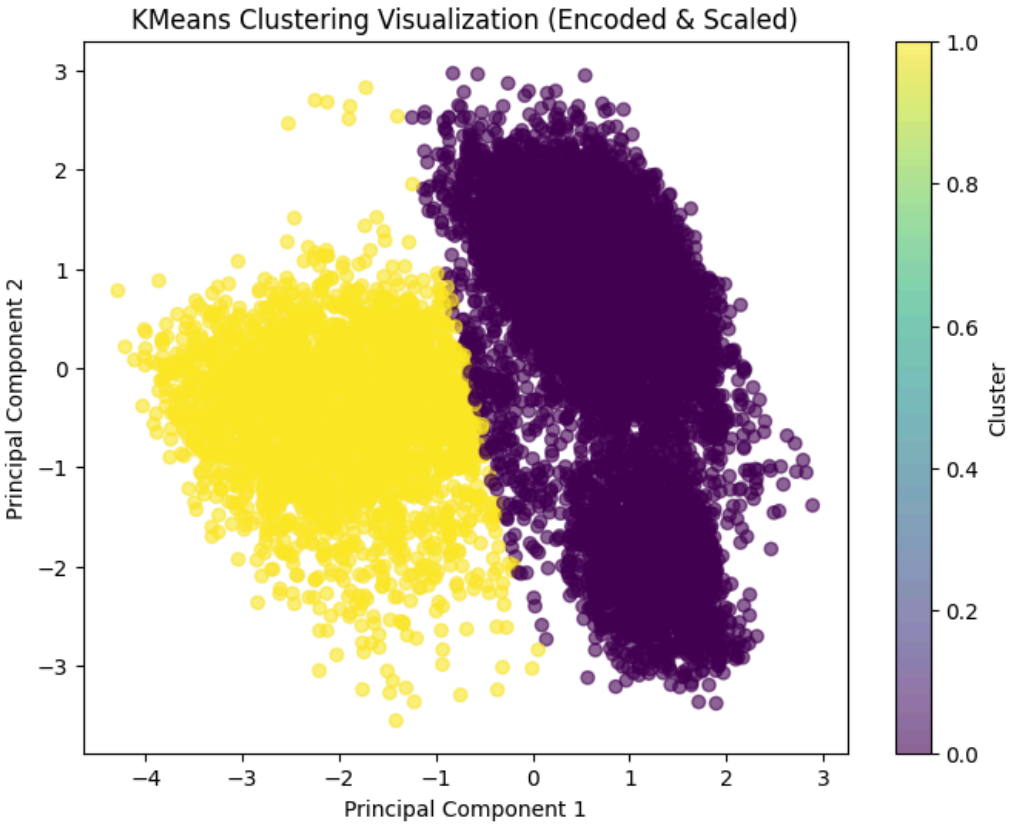


2. Feature Preparation

All categorical features were **label-encoded**, and numerical features were **scaled** to ensure comparability across features.

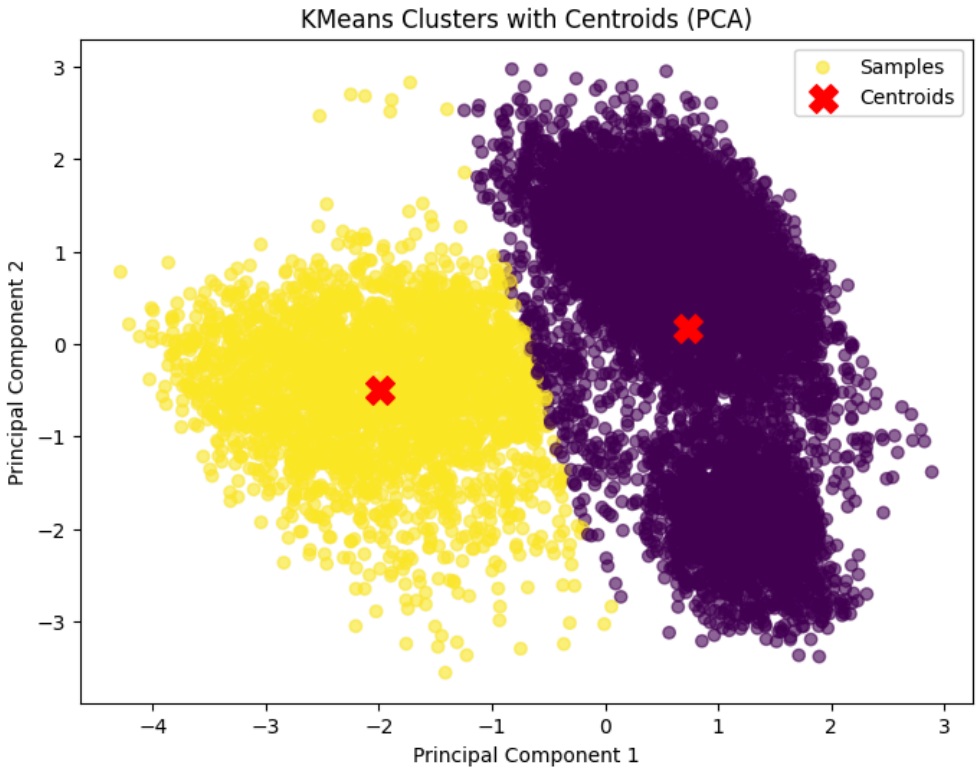
3. Cluster Visualization

Principal Component Analysis (PCA) was used to reduce the dataset to two dimensions for visualization. The resulting scatter plot shows two well-separated clusters.



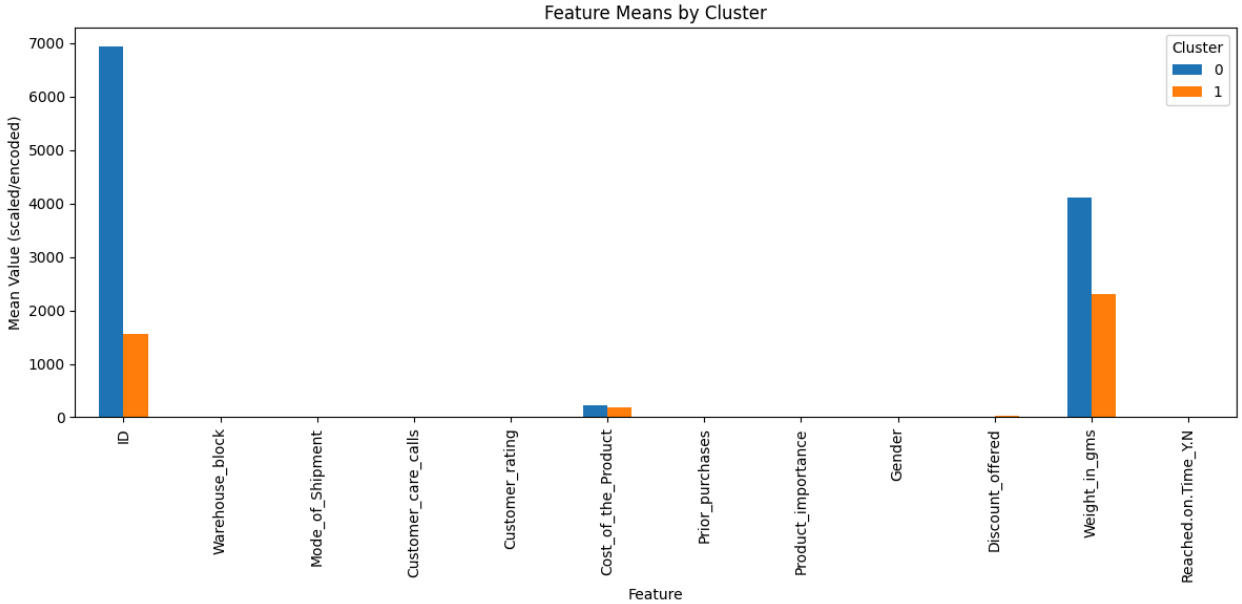
4. Centroid Analysis

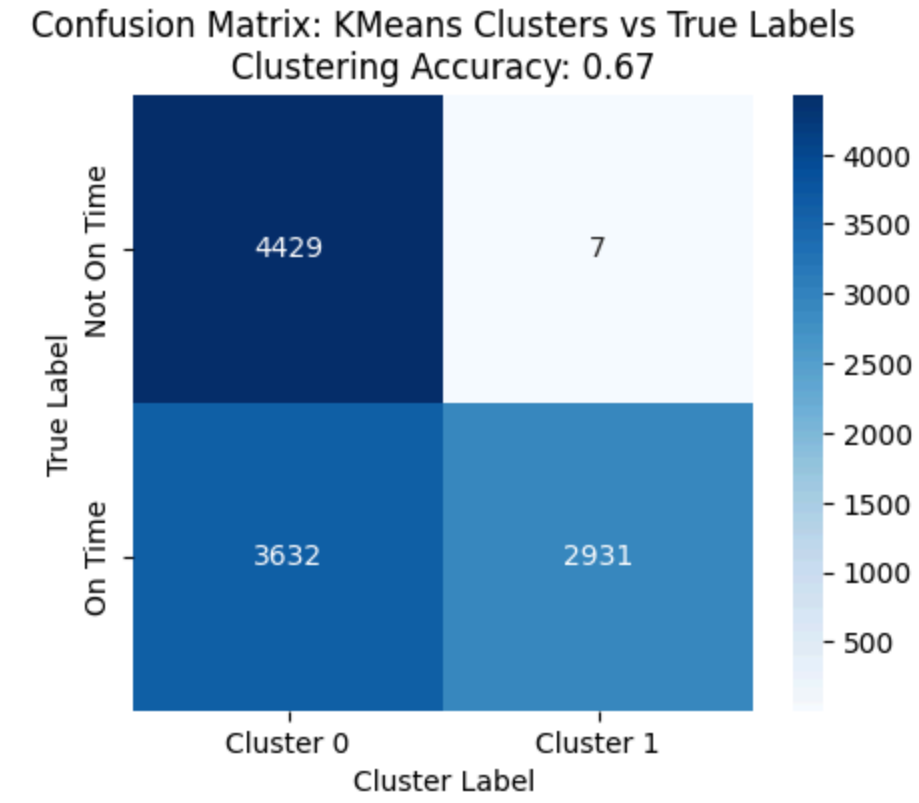
Cluster centroids were projected onto the PCA plot, highlighting the center of each group.



5. Cluster Interpretation

- Cluster feature means were compared to understand the characteristics of each group.
- Clusters were compared to the true delivery status using a **confusion matrix** and clustering accuracy.





6. Key Insights

- KMeans identified **two main clusters**, which partially align with on-time and late deliveries.
- Features such as **discount offered** and **shipment weight** were most influential in cluster formation.
- The clustering accuracy was **moderate(0.67)**, indicating that while clusters capture some underlying structure, additional features may be needed to fully explain delivery outcomes.

Conclusion

The results indicate that all models achieved **moderate and similar accuracy**, suggesting that the available features only partially explain delivery outcomes. The neural network performed slightly better than other models based on test accuracy, but no model achieved high predictive power.

Model performance was limited by **overlapping feature distributions** and the absence of highly

informative variables. Even after oversampling, models struggled to correctly predict the minority class, highlighting inherent limitations in the dataset.

Key challenges identified include:

- **Class imbalance** in the target variable
- **Limited feature informativeness**
- **Potential unmeasured factors** influencing delivery outcomes

Overall, the study emphasizes the need for **richer data, additional predictive features, and more nuanced modeling approaches** to improve accuracy and better capture the factors affecting shipment delivery.