VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

# FRAUD DETECTION IN SUPPLY CHAIN LOGISTICS

**PROJECT REPORT**

[ J-COMPONENT]

**Submitted by**

**Sanidhya Chaudhary (20MIA1006)**

**Sricharan Sridhar (20MIA1014)**

**Gaurav Dwivedi (20MIA1037)**

**Rohit Choudhary (20MIA1069)**

PREPARED FOR

## NO SQL DATABASES [CSE3086]

**Submitted To**

Dr.Sivagami M

Professor [SCOPE]

# Fraud Detection in Supply Chain Logistics

Chaudhary Sanidhya*, Rohit Choudhary*, Sricharan Sridhar*, Gaurav Dwivedi*

**School of computer and Science Engineering, Vellore Institute of Technology, India*

Keywords: Machine Learning Algorithms, Supply Chain logistics, Risk Management, Feature engineering

## 1. *Abstract*

This report presents a comprehensive data science project centred around the development of a fraud detection system in the context of supply chain logistics. The primary objective of this project is to design and implement a machine learning solution capable of identifying fraudulent activities within the supply chain, which can have significant financial and operational implications.

Supply chain logistics plays a pivotal role in the global economy, facilitating the movement of goods from suppliers to consumers. However, this interconnectedness also presents opportunities for fraudulent activities such as theft, counterfeiting, and unauthorised diversions. The project addresses these challenges by leveraging data-driven insights and machine learning techniques.

Throughout the project, several machine learning models are employed, including Logistic Regression, Decision Tree Classifier, K-Nearest Neighbors, and Random Forest. These models are evaluated based on performance metrics such as accuracy, recall, and F1 score, all of which are essential for effectively identifying fraudulent activities within the supply chain.

The report offers a detailed analysis of the results produced by each model, shedding light on their capabilities and limitations in a real-world supply chain setting. Additionally, the report includes visual representations of feature importance, which play a crucial role in understanding how these models make decisions.

# 2.*Introduction*

## 2.1 Background

In the complex and interconnected world of global commerce, supply chain logistics play a pivotal role in the flow of goods from producers to consumers. This intricate system encompasses a myriad of processes, partners, and stakeholders, making it a critical component of modern economies. However, as with any complex system, supply chains are susceptible to various challenges, and one of the most significant is fraud.

The advent of the Internet of Things (IoT) has ignited a technological revolution, permeating various industries and substantially augmenting data generation. This surge in data holds immense potential, offering companies the opportunity to make more informed decisions and gain a competitive edge. One pivotal area where this potential has been harnessed is in the realm of predictive analytics for product sales and fraud detection within supply chains.

Fraud in supply chain logistics can take various forms, from theft and counterfeiting to financial scams and documentation forgery. These fraudulent activities can result in substantial financial losses, operational disruptions, and damage to a company's reputation. They can lead to issues such as increased costs, delays, and even the loss of valuable products.

## 2.2 Objective

The primary objective of this project is to design, develop, and evaluate a robust fraud detection system that is tailored specifically to the intricate dynamics of supply chain logistics. We aim to create a solution that can accurately identify and mitigate fraudulent events, thereby enabling supply chain stakeholders to take timely corrective actions. In doing so, we aspire to enhance the overall security and efficiency of global supply chains.

## 2.3 Significance

The significance of this project is paramount, given the far-reaching impact of fraud in supply chains. The success of this endeavour could revolutionise the way supply chain logistics combat fraudulent activities. By implementing effective fraud detection mechanisms, companies can enhance their financial stability, operational efficiency, and, most crucially, customer trust. A more secure and reliable global trade environment can be achieved by reducing fraudulent activities in the supply chain.

The need for such a system becomes even more pressing as supply chains grow in complexity and scale, often involving multiple regions and numerous partners. The consequences of fraud are not limited to the financial realm; they can also have far-reaching legal, ethical, and societal implications.

This literature review embarks on a comprehensive exploration of the research landscape surrounding these critical domains.

*2.4 Literature Review*

In today's data-driven landscape, enterprises grapple with vast reservoirs of information, stemming from IoT implementations and other data-rich sources. Effectively harnessing this data to serve the strategic growth of companies has become a primary focus. Accurate predictive analytics have the capacity to bolster customer satisfaction, enhance market competitiveness, optimize inventory management, and refine supplier selection, all of which are paramount for contemporary and future business development. Paper 1: Product Fraud Detection and Machine Learning

Paper 1 emphasizes the burgeoning importance of product data and its effective utilization. The fusion of machine learning models, particularly the hybridization of XG-Boost and Random Forest algorithms, offers a promising approach to product fraud detection. The resulting hybrid model exhibits superior performance, as evidenced by impressive Confusion Matrix results. Moreover, this model is evaluated using the Data Co smart supply chain dataset, where it outperforms alternative machine learning approaches, boasting a significantly higher F1 score.

Paper 2: IoT and Sales Prediction with Decision Trees Paper 2 underscores the pivotal role of IoT technology in data generation, particularly for sales prediction. Leveraging the decision tree algorithm, the paper crafts a predictive model that combines feature engineering and algorithmic processing to forecast product sales. Evaluation results demonstrate the model's efficacy, with an accuracy rate surpassing that of logistic regression and naive Bayes models.

Paper 3: Fraud Prediction and SVM Classification Paper 3 delves into fraud prediction within the supply chain domain, driven by the IoT's data-rich landscape. Employing the SVM classification model, the paper conducts feature engineering on the dataset to enable effective modelling. The SVM classification model showcases remarkable accuracy, surpassing logistic regression and naive Bayes models, bolstering its position as a powerful tool for data classification and regression.

Paper 4: Fraud Detection in Supply Chain with Machine Learning The paper then discusses the use of ML for fraud detection in supply chains. ML algorithms can be used to identify patterns in data that may indicate fraud, such as unusual orders, suspicious transactions, or changes in shipping patterns. ML algorithms can also be used to predict the likelihood of fraud occurring, which can help businesses to focus their resources on the most likely areas of risk. The paper reviews a number of studies that have used ML for fraud detection in supply chains. These studies have shown that ML can be an effective tool for detecting fraud, with some studies reporting detection rates of up to 90%. However, the paper also notes that there are a number of challenges that need to be addressed in order to use ML effectively for fraud detection in supply chains.

Paper 5: Supply chain fraud prediction with machine learning and artificial intelligence Ml and AI monitor the supply chain in real-time, allowing companies to quickly identify and respond to potential threats, automatically flag suspicious activities. the data comes from real-world, around 180000 observations taken from supply chain transactions that have been mined over three years from a large manufacturing company. To deal with class imbalance problems they used the random under sampler (RUS) with synthetic minority over-sampling technique (smote). Feature engineering deployed for extracting feature importance to avoid useless attributes. logistic regression, cat boost, random forest classifier, ai sequential model.
Paper 6: Understanding and mitigating supply chain fraud Areas of potential supply chain fraud: 1. Misappropriation 2. Corruption 3. Financial statement fraud.
Motivation for fraud - 3 potential reasons: 1. Perceived pressure - layoffs, cheating to survive 2. Opportunistic - knowing loopholes and taking advantage of them 3. Rationalizing fraud elaborated on a detection system that gives a score based on the possibility of fraud happening and its impact on the environment.

## 3. *Data Exploration*

### 3.1 Data Source

The foundation of any data-driven project is the quality and relevance of the dataset. In this project, we leveraged a dataset that captures a wide range of information related to supply chain logistics. The dataset is attributed to "DataCo Global" and encompasses various attributes, including customer details, order specifics, product information, and more. DataCo Global specialises in products like clothing, sports equipment, and electronic supplies. The dataset has been made publicly available for research purposes and is appropriately cited as:

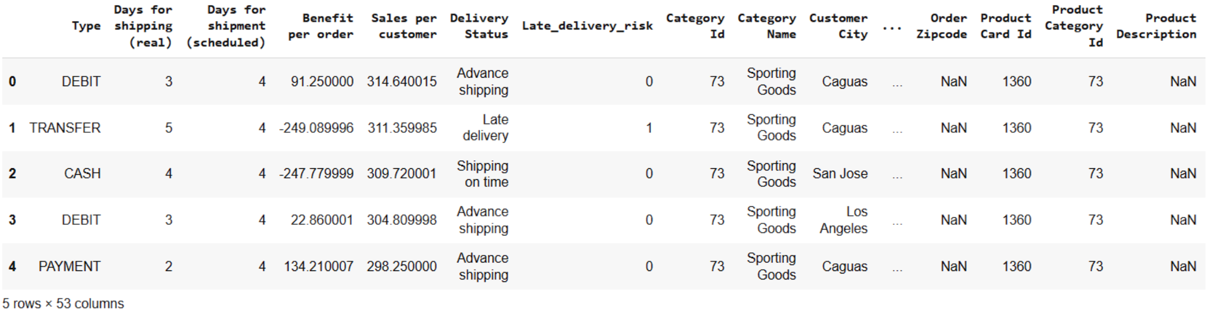
Constante, Fabian; Silva, Fernando; Pereira, António (2019), “DataCo SMART SUPPLY CHAIN FOR BIG DATA ANALYSIS”, Mendeley Data, V5, doi: 10.17632/8gx2fvg2k6.5

https://data.mendeley.com/datasets/8gx2fvg2k6/5

The dataset comprises a substantial 180,520 rows, capturing detailed information on supply chain operations. It represents a realistic portrayal of the complexities and intricacies of supply chain logistics operations.

### 3.2 Data Structure and Dimensions

Our dataset comprises 180,520 rows and 53 columns, highlighting the richness and comprehensiveness of the information it contains. These columns encompass various data types, including numerical, categorical, and binary features. Each row represents a unique order within the supply chain logistics system, and the columns contain information about different aspects of that order.



| | Type | Days for shipping (real) | Days for shipment (scheduled) | Benefit per order | Sales per customer | Delivery Status | Late_delivery_risk | Category Id | Category Name | Customer City | ... | Order Zipcode | Product Card Id | Product Category Id | Product Description |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | DEBIT | 3 | 4 | 91.250000 | 314.640015 | Advance shipping | 0 | 73 | Sporting Goods | Caguas | ... | NaN | 1360 | 73 | NaN |
| **1** | TRANSFER | 5 | 4 | -249.089996 | 311.359985 | Late delivery | 1 | 73 | Sporting Goods | Caguas | ... | NaN | 1360 | 73 | NaN |
| **2** | CASH | 4 | 4 | -247.779999 | 309.720001 | Shipping on time | 0 | 73 | Sporting Goods | San Jose | ... | NaN | 1360 | 73 | NaN |
| **3** | DEBIT | 3 | 4 | 22.860001 | 304.809998 | Advance shipping | 0 | 73 | Sporting Goods | Los Angeles | ... | NaN | 1360 | 73 | NaN |
| **4** | PAYMENT | 2 | 4 | 134.210007 | 298.250000 | Advance shipping | 0 | 73 | Sporting Goods | Caguas | ... | NaN | 1360 | 73 | NaN |

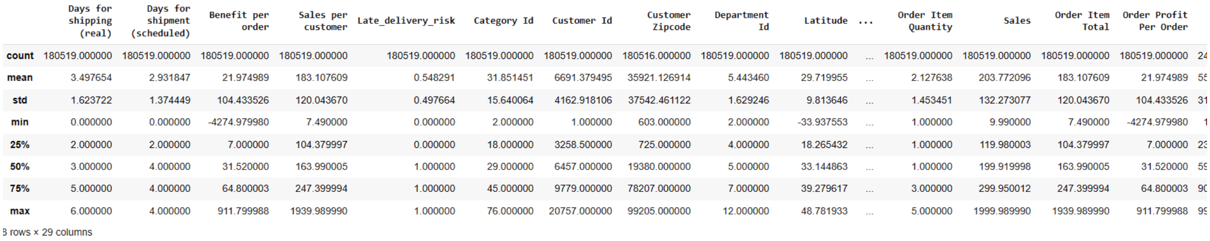
5 rows × 53 columns

## 3.3 Data Overview

In the pursuit of gaining insights from the dataset, we conducted a comprehensive exploration of the data. We began by examining the first few rows to get a glimpse of the data's structure and contents. This initial overview helped us understand the dataset's general format and provided a foundation for more detailed exploration.

## 3.4 Data Statistics and Summary

For a more in-depth understanding of the dataset, we computed descriptive statistics for the numerical features. These statistics include measures such as the mean, standard deviation, minimum, and maximum values. By analysing these statistics, we gained insights into the distribution and variability of numerical data, which can be crucial for feature selection and model training.



| | Days for shipping (real) | Days for shipment (scheduled) | Benefit per order | Sales per customer | Late_delivery_risk | Category Id | Customer Id | Customer Zipcode | Department Id | Latitude | ... | Order Item Quantity | Sales | Order Item Total | Order Profit Per Order | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **count** | 180519.000000 | 180519.000000 | 180519.000000 | 180519.000000 | 180519.000000 | 180519.000000 | 180519.000000 | 180516.000000 | 180519.000000 | 180519.000000 | ... | 180519.000000 | 180519.000000 | 180519.000000 | 180519.000000 | 24 |
| **mean** | 3.497654 | 2.931847 | 21.974989 | 183.107609 | 0.548291 | 31.851451 | 6691.379495 | 35921.126914 | 5.443460 | 29.719955 | ... | 2.127638 | 203.772096 | 183.107609 | 21.974989 | 55 |
| **std** | 1.623722 | 1.374449 | 104.433526 | 120.043670 | 0.497664 | 15.640064 | 4162.918106 | 37542.461122 | 1.629246 | 9.813646 | ... | 1.453451 | 132.273077 | 120.043670 | 104.433526 | 31 |
| **min** | 0.000000 | 0.000000 | -4274.979980 | 7.490000 | 0.000000 | 2.000000 | 1.000000 | 603.000000 | 2.000000 | -33.937553 | ... | 1.000000 | 9.990000 | 7.490000 | -4274.979980 | 1 |
| **25%** | 2.000000 | 2.000000 | 7.000000 | 104.379997 | 0.000000 | 18.000000 | 3258.500000 | 725.000000 | 4.000000 | 18.265432 | ... | 1.000000 | 119.980003 | 104.379997 | 7.000000 | 23 |
| **50%** | 3.000000 | 4.000000 | 31.520000 | 163.990005 | 1.000000 | 29.000000 | 6457.000000 | 19380.000000 | 5.000000 | 33.144863 | ... | 1.000000 | 199.919998 | 163.990005 | 31.520000 | 59 |
| **75%** | 5.000000 | 4.000000 | 64.800003 | 247.399994 | 1.000000 | 45.000000 | 9779.000000 | 78207.000000 | 7.000000 | 39.279617 | ... | 3.000000 | 299.950012 | 247.399994 | 64.800003 | 90 |
| **max** | 6.000000 | 4.000000 | 911.799988 | 1939.989990 | 1.000000 | 76.000000 | 20757.000000 | 99205.000000 | 12.000000 | 48.781933 | ... | 5.000000 | 1999.989990 | 1939.989990 | 911.799988 | 99 |

3 rows × 29 columns

## 3.5 Data Visualization

Visualising data is a powerful way to identify patterns, trends, and outliers. We employed various data visualisation techniques to gain insights into the dataset. We created plots and charts to visualise distributions, relationships between variables, and any potential outliers.

These visualisations were instrumental in identifying potential areas of concern, such as discrepancies between expected delivery times and actual delivery times, as well as trends in fraudulent orders. Visualisations also provided a foundation for more advanced analysis.

### 3.6 Initial Observations

During the data exploration phase, several initial observations were made:

- There are various types of orders, ranging from "SUSPECTED_FRAUD" to "COMPLETE," providing a spectrum of order statuses.
- Delivery status includes categories such as "Late delivery" and "Advance shipping," which can be used to analyze potential correlations with fraudulent activities.
- Some features, such as "Days for shipping (real)," "Benefit per order," and "Product Price," appear to be critical for understanding patterns and detecting anomalies.
- Categorical data, including customer details and product names, is available, necessitating preprocessing to make it suitable for machine learning models.

The initial data exploration stage was instrumental in providing a comprehensive understanding of the dataset's structure and contents. It laid the foundation for the subsequent data preprocessing and model selection phases.

## 4. Methodology

### 4.1 Data Preprocessing and Cleaning

The success of any machine learning model hinges on the quality of the input data. In this project, we meticulously cleaned and preprocessed the dataset:

#### 4.1.1 Handling Missing Data:

- We identified columns with missing values and employed strategies like imputation to fill in these gaps. Effective handling of missing data ensures that the model receives complete information.

#### 4.1.2 Encoding Categorical Data:

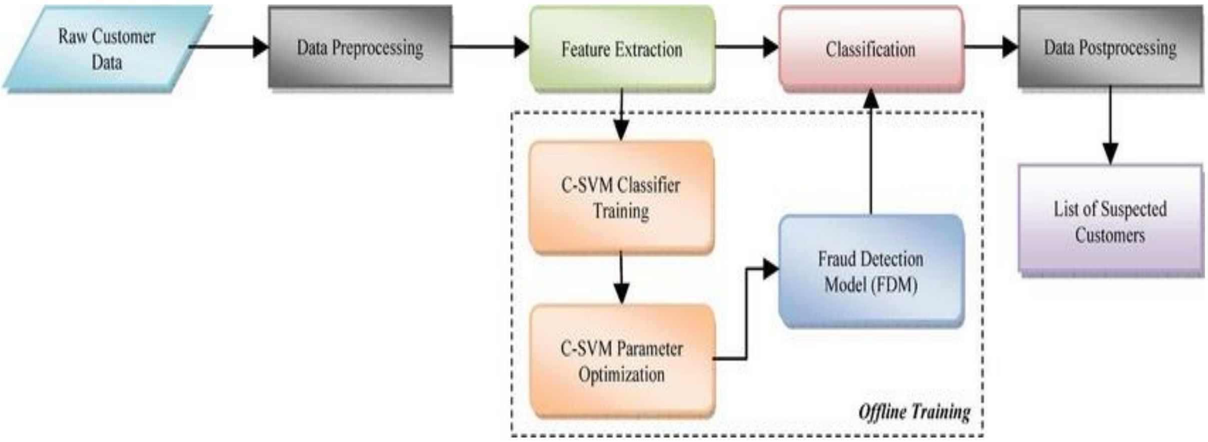
- Many attributes in the dataset were categorical. Through label encoding, we transformed these categories into numerical values. This preprocessing step allowed us to incorporate these attributes effectively into our machine learning models.

#### 4.1.3 Feature Engineering:

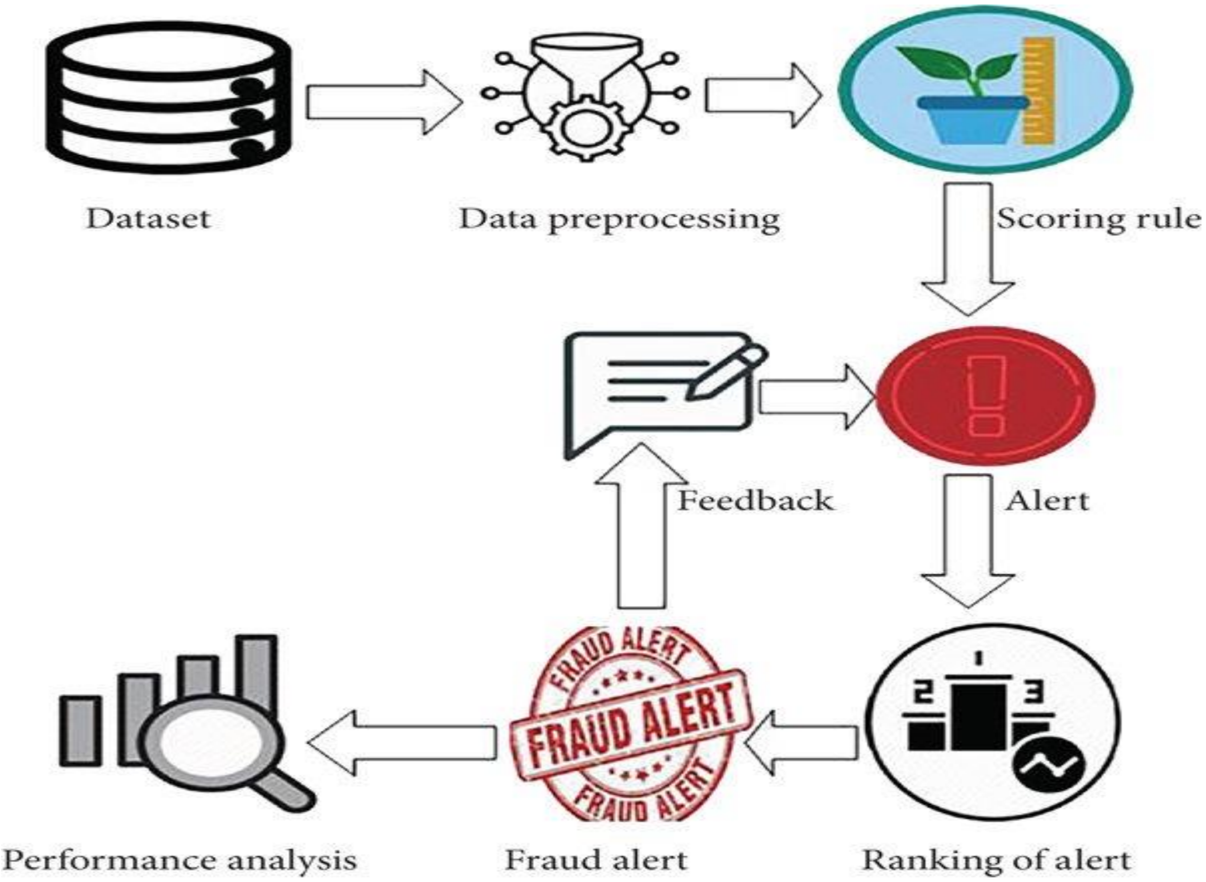
- To enhance the predictive power of our models, we engineered new features from existing ones. This included creating time-based features, aggregating data, and introducing interaction terms, providing the models with more comprehensive information.

## BLOCK DIAGRAMS

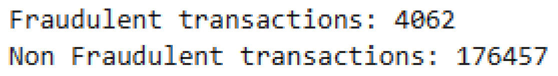
### MACHINE LEARNING LIFECYCLE OF THE PROJECT


Raw Customer Data
Data Preprocessing
Feature Extraction
Classification
Data Postprocessing
C-SVM Classifier Training
C-SVM Parameter Optimization
Fraud Detection Model (FDM)
List of Suspected Customers
Offline Training

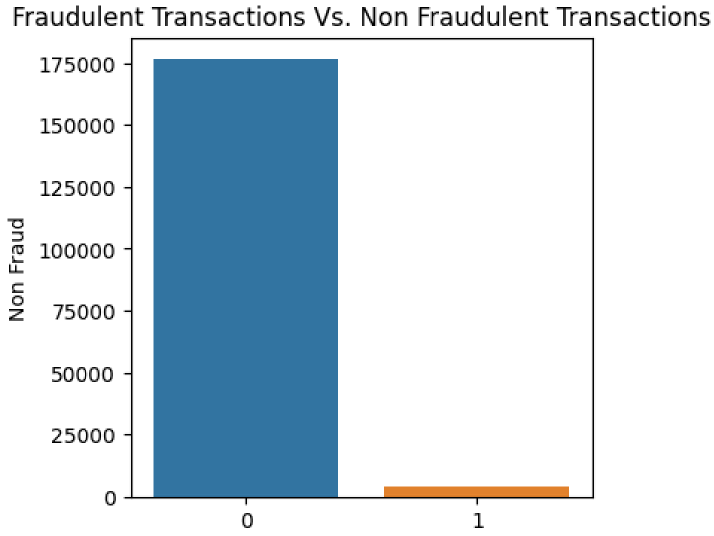
This is the block diagram of the complete flow of the project which involves the explanation of risk/fraud assessment in Supply Chain Logistics



## 4.2 HANDLING IMBALANCED DATASET



```
Fraudulent transactions: 4062
Non Fraudulent transactions: 176457
```

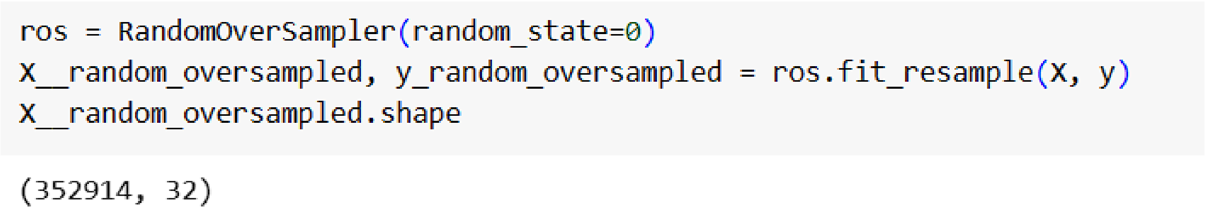


### 4.2.1 Random Oversampling

Random oversampling is a technique used to address the issue of class imbalance in a dataset. It involves increasing the size of the minority class by randomly replicating samples until the dataset becomes balanced.

The process of random oversampling is relatively straightforward. The first step is to identify the class that is in the minority and the class that is in the majority. Once this has been done, the minority class is randomly oversampled until its size is increased to the same size as the majority class. This can be done using different approaches, such as randomly duplicating samples from the minority class, or creating new synthetic samples that are similar to existing samples in the minority class.

While random oversampling is a simple and easy-to-implement technique, there are some potential drawbacks to consider. One major concern is that by replicating samples from the minority class, the resulting dataset may become overfit to the minority class and result in a decrease in the overall performance of the machine learning model. Additionally, random oversampling can lead to biased results if the replicated samples are not truly representative of the minority class.



```
ros = RandomOverSampler(random_state=0)
X__random_oversampled, y_random_oversampled = ros.fit_resample(X, y)
X__random_oversampled.shape
```

```
(352914, 32)
```

### 4.2.2 ADASYN oversampling

ADASYN is a more generic framework, for each of the minority observations it first finds the impurity of the neighborhood, by taking the ratio of majority observations in the neighborhood and k. Calculate the density distribution: ADASYN begins by calculating the density distribution of each minority class sample in the dataset. This is done by estimating the density of each minority sample based on the distances to its k nearest neighbors, and then normalizing the densities to sum up to 1.

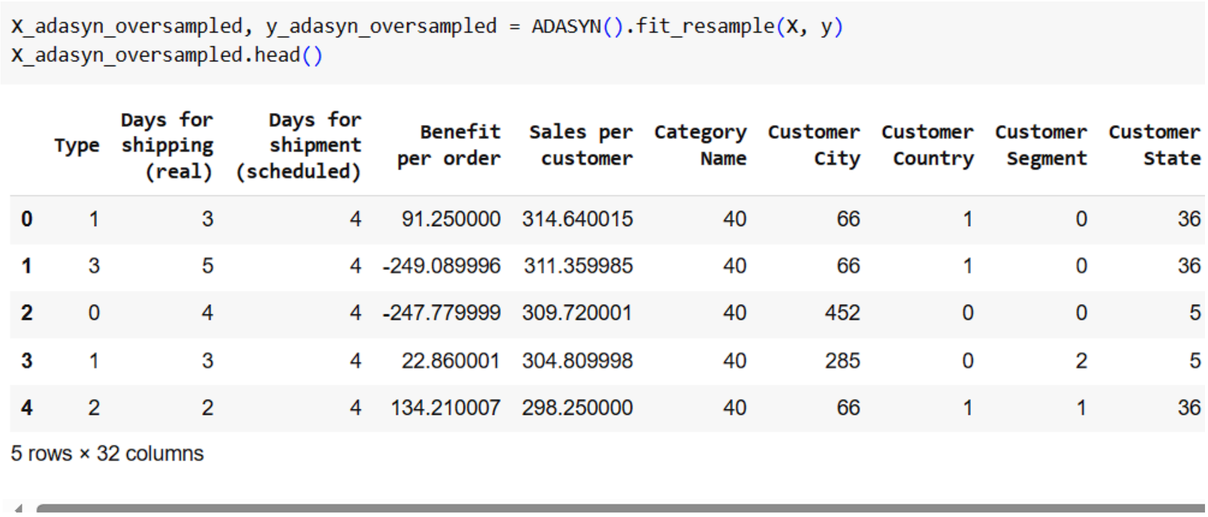
1. Compute the imbalance ratio: ADASYN then computes the imbalance ratio of the dataset, which is the ratio between the number of majority class samples and the number of minority class samples.

2. Calculate the number of synthetic samples: For each minority class sample, ADASYN calculates the number of synthetic samples to be generated based on the local density distribution and the imbalance ratio. Specifically, the number of synthetic samples is calculated as:
   #synthetic samples = (#majority samples - #minority samples) * density_i / sum(density) where density i is the density of the i-th minority sample, and sum(density) is the sum of all densities of the minority  samples.
   The number of synthetic samples is then rounded to the nearest integer.

3. Generate synthetic samples: For each minority class sample, ADASYN generates the calculated number of synthetic samples by applying the SMOTE algorithm, but only to the minority samples in the k nearest neighbors that have lower densities than the current sample. This is done to ensure that the synthetic samples are generated in the regions of the feature space where the class boundary is more difficult to learn. Specifically, for each minority sample, ADASYN selects its k nearest neighbors that belong to the minority class and have lower densities, and generates synthetic samples by applying the SMOTE algorithm. The number of synthetic samples is determined by the previously calculated number of synthetic samples.

4. Combine the original and synthetic samples: Finally, ADASYN combines the original minority

samples with the newly generated synthetic samples to create a balanced dataset.



```
X_adasyn_oversampled, y_adasyn_oversampled = ADASYN().fit_resample(X, y)
X_adasyn_oversampled.head()
```

| | Type | Days for shipping (real) | Days for shipment (scheduled) | Benefit per order | Sales per customer | Category Name | Customer City | Customer Country | Customer Segment | Customer State |
|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | 3 | 4 | 91.250000 | 314.640015 | 40 | 66 | 1 | 0 | 36 |
| **1** | 3 | 5 | 4 | -249.089996 | 311.359985 | 40 | 66 | 1 | 0 | 36 |
| **2** | 0 | 4 | 4 | -247.779999 | 309.720001 | 40 | 452 | 0 | 0 | 5 |
| **3** | 1 | 3 | 4 | 22.860001 | 304.809998 | 40 | 285 | 0 | 2 | 5 |
| **4** | 2 | 2 | 4 | 134.210007 | 298.250000 | 40 | 66 | 1 | 1 | 36 |

5 rows × 32 columns

### 4.2.3 SMOTE

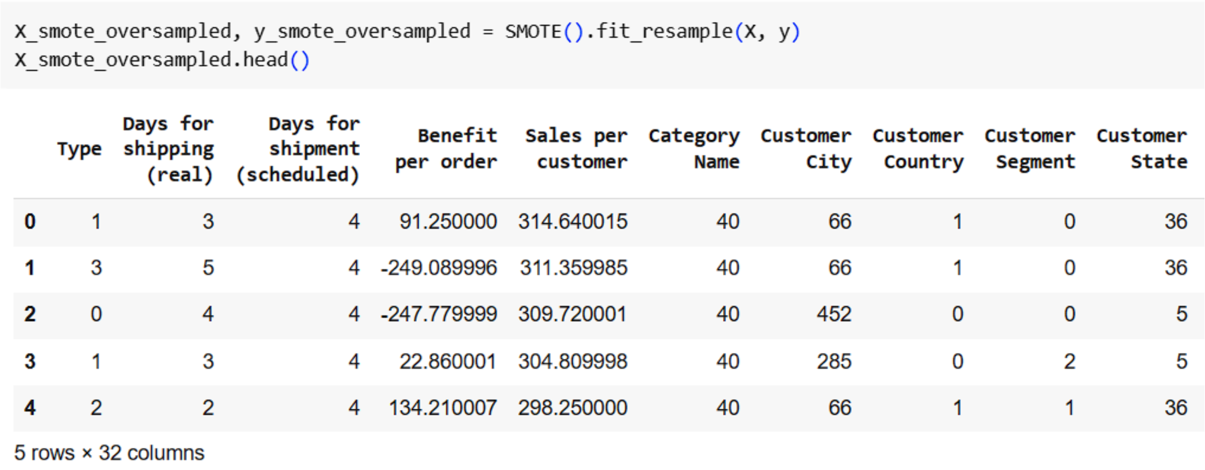
SMOTE (Synthetic Minority Over-sampling Technique) is a popular algorithm used for imbalanced data sets. It is a technique used to oversample the minority class in a data set by generating synthetic samples instead of replicating existing samples. This helps to balance the class distribution, which can improve the performance of machine learning models.

The basic idea behind SMOTE is to create new synthetic examples of the minority class by interpolating between existing minority class samples. This is done by selecting a random minority class example and then selecting one of its k nearest neighbours. A new synthetic example is then created by randomly interpolating between the two examples.

The interpolation process is done as follows: for each attribute in the selected minority class example, a new value is generated as a linear combination of the attribute values of the two examples. The weight of the linear combination is a random number between 0 and 1. This results in a new synthetic example that is different from both the selected minority class example and its nearest neighbour, but is still representative of the minority class.

The process of creating new synthetic examples continues until the desired ratio of minority class to majority class is achieved. This is often done by specifying a target percentage of the minority class in the final data set.

It is important to note that while SMOTE can improve the performance of machine learning models, it is not a silver bullet for imbalanced data sets. Careful consideration must be given to the choice of k (the number of nearest neighbors to consider), as well as other factors such as the choice of classifier and the evaluation metric used to measure performance. Additionally, SMOTE should only be used on the training data and not on the test data, as this can lead to overfitting.



```
X_smote_oversampled, y_smote_oversampled = SMOTE().fit_resample(X, y)
X_smote_oversampled.head()
```

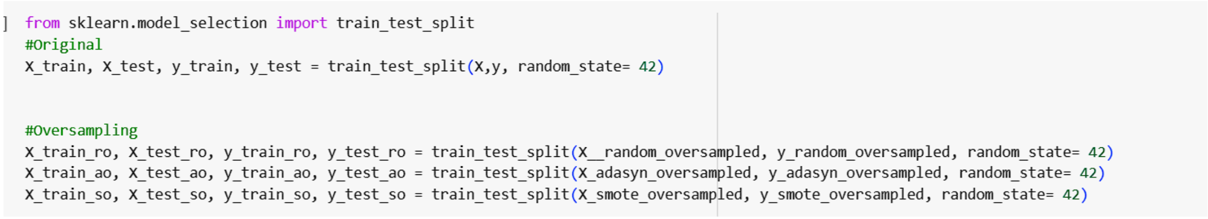
| | Type | Days for shipping (real) | Days for shipment (scheduled) | Benefit per order | Sales per customer | Category Name | Customer City | Customer Country | Customer Segment | Customer State |
|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | 3 | 4 | 91.250000 | 314.640015 | 40 | 66 | 1 | 0 | 36 |
| **1** | 3 | 5 | 4 | -249.089996 | 311.359985 | 40 | 66 | 1 | 0 | 36 |
| **2** | 0 | 4 | 4 | -247.779999 | 309.720001 | 40 | 452 | 0 | 0 | 5 |
| **3** | 1 | 3 | 4 | 22.860001 | 304.809998 | 40 | 285 | 0 | 2 | 5 |
| **4** | 2 | 2 | 4 | 134.210007 | 298.250000 | 40 | 66 | 1 | 1 | 36 |

5 rows × 32 columns

## 4.3 Data Splitting

For the purpose of model training and evaluation, we divided the dataset into training and testing subsets. Our goal was to train models that could predict fraud in supply chain logistics effectively. We performed this task for various classification models:

Preparing 4 diff datasets after performing sampling techniques, splitting the datasets into their respective train and test split datasets as shown below.



```
]  from sklearn.model_selection import train_test_split
   #Original
   X_train, X_test, y_train, y_test = train_test_split(X,y, random_state= 42)


   #Oversampling
   X_train_ro, X_test_ro, y_train_ro, y_test_ro = train_test_split(X__random_oversampled, y_random_oversampled, random_state= 42)
   X_train_ao, X_test_ao, y_train_ao, y_test_ao = train_test_split(X_adasyn_oversampled, y_adasyn_oversampled, random_state= 42)
   X_train_so, X_test_so, y_train_so, y_test_so = train_test_split(X_smote_oversampled, y_smote_oversampled, random_state= 42)
```

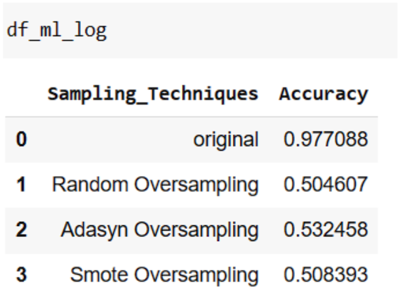
## 4.4 Models with Results

### 4.4.1 Logistic Regression:

- Logistic regression served as a baseline model for fraud detection. This classification technique provides interpretable results and insights into feature importance.

Performance:

- Accuracy: The Logistic Regression model under original dataset achieved an accuracy of 97.76%. This indicates that 97.76% of the predictions were correct.
- Recall: The recall score was 56.91%. This metric measures the ability of the model to correctly identify positive cases (fraud). A recall score of 56.91% suggests that 56.91% of actual fraud cases were correctly identified.
- F1 Score: The F1 score, a combination of precision and recall, was 30.49%.



```
df_ml_log
```

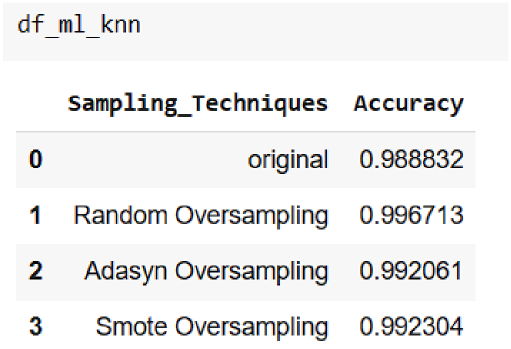
| | Sampling_Techniques | Accuracy |
|---|---|---|
| 0 | original | 0.977088 |
| 1 | Random Oversampling | 0.504607 |
| 2 | Adasyn Oversampling | 0.532458 |
| 3 | Smote Oversampling | 0.508393 |

### 4.4.2 K-Nearest Neighbors (KNN):

- KNN, a simple yet effective classification algorithm, was used to explore its potential for detecting fraudulent activities.

Performance:

- Accuracy: The KNN model achieve an highest accuracy of 99.67% under random oversampling.



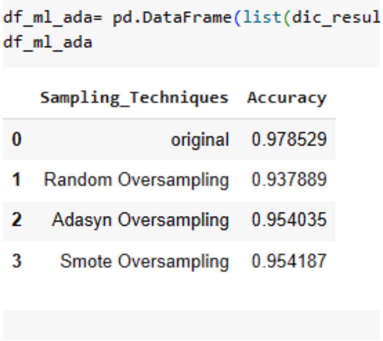
```
df_ml_knn
```

| | Sampling_Techniques | Accuracy |
|---|---|---|
| 0 | original | 0.988832 |
| 1 | Random Oversampling | 0.996713 |
| 2 | Adasyn Oversampling | 0.992061 |
| 3 | Smote Oversampling | 0.992304 |

### 4.4.3 AdaBoost Boosting

The boosting method enhances the strength of weak learners, and the gradient boosting algorithm can be introduced by first discussing the AdaBoost algorithm. AdaBoost commences by training a decision tree with uniform weights assigned to each datapoint. After assessing the first tree's performance, it increases the weights of difficult-to-classify observations and decreases the weights of easy-to-classify ones. The second tree is then trained on this weighted data to improve upon the initial model's predictions, and the resulting model is the sum of Tree 1 and Tree 2. The classification error of this new 2-tree ensemble model is used to build a third tree that predicts the revised residuals. This iterative process continues for a specific number of iterations, with the final ensemble model's predictions being the weighted sum of the previous models' predictions.

On the other hand, Gradient Boosting trains multiple models slowly, progressively, and sequentially. The primary distinction between AdaBoost and Gradient Boosting lies in how they detect the weak learners' shortcomings. AdaBoost identifies them using observations with high weights, while Gradient Boosting employs gradients in the loss function to achieve the same



```
df_ml_ada= pd.DataFrame(list(dic_resul
df_ml_ada
```

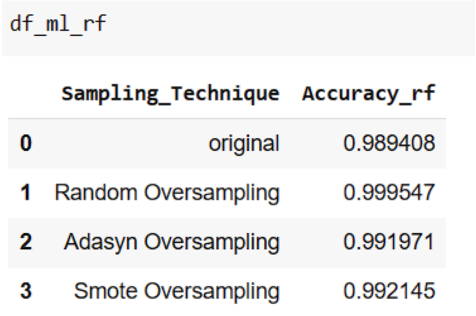
|  | Sampling_Techniques | Accuracy |
|---|---|---|
| 0 | original | 0.978529 |
| 1 | Random Oversampling | 0.937889 |
| 2 | Adasyn Oversampling | 0.954035 |
| 3 | Smote Oversampling | 0.954187 |

4.4.4 Random Forest:

- The random forest is a popular algorithm due to its simplicity and diversity, and it is an ensemble technique that employs multiple decision trees. Each tree generates a set of predictions, and the final output prediction is determined by the majority vote of the classes, as depicted in figure 4. Rather than searching for the most important features to split, the random forest randomly selects a subset of features to determine the best feature for node splitting during the tree growing process.

Performance:

- Accuracy: The Random Forest model achieved an accuracy of 99.95% under random oversampling.
- Recall: It had a high recall score of 97.59%.
- F1 Score: The F1 score for Random Forest was 98.00%.



```
df_ml_rf
```

|  | Sampling_Technique | Accuracy_rf |
|---|---|---|
| 0 | original | 0.989408 |
| 1 | Random Oversampling | 0.999547 |
| 2 | Adasyn Oversampling | 0.991971 |
| 3 | Smote Oversampling | 0.992145 |

### 4.5 Model Evaluation Metrics

The models were evaluated based on essential metrics:

#### 4.5.1 Accuracy:

- Accuracy measures the proportion of correctly predicted fraud cases. It provides a general sense of how well the model is performing.

#### 4.5.2 Recall:

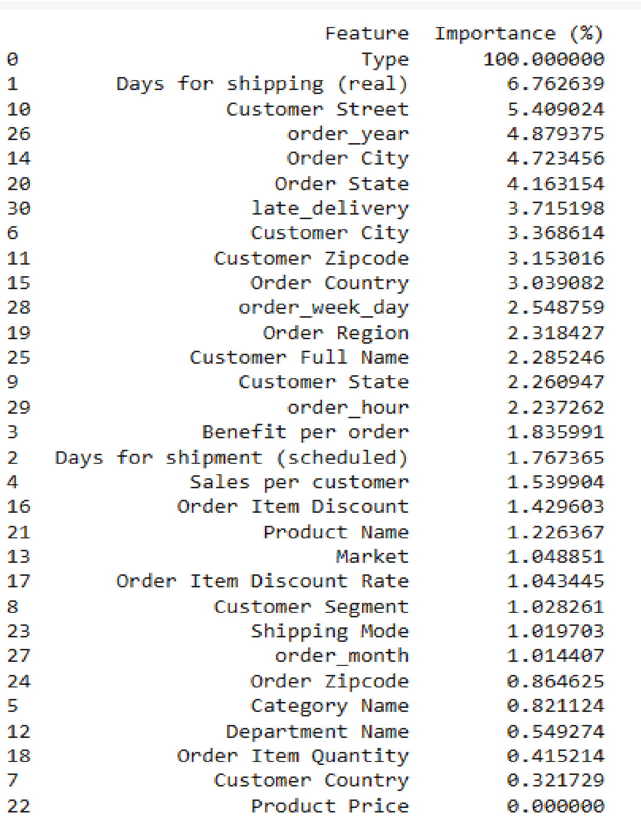
- Recall (Sensitivity) quantifies the model's ability to identify actual fraud cases among all true fraud cases. This metric is vital for minimising false negatives.

#### 4.5.3 F1 Score:

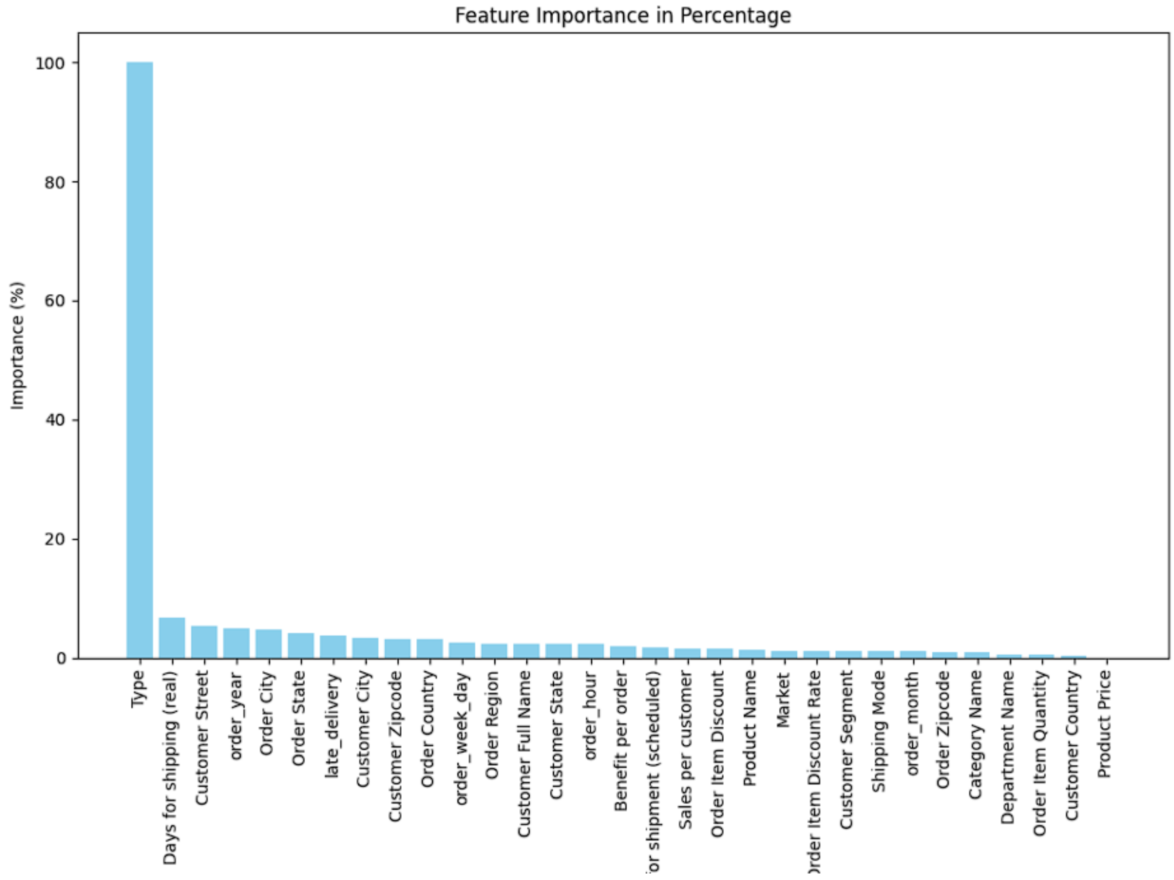
- The F1 score is the harmonic mean of precision and recall. It strikes a balance between identifying fraud cases (recall) and avoiding false alarms (precision).

### 4.6 Feature Importance Analysis

For random forest under random oversampling models, we examined feature importance. This analysis helped us understand which features significantly influenced the model's decisions in detecting fraud.



```
                          Feature  Importance (%)
0                            Type      100.000000
1        Days for shipping (real)        6.762639
10                Customer Street        5.409024
26                     order_year        4.879375
14                     Order City        4.723456
20                    Order State        4.163154
30                  late_delivery        3.715198
6                   Customer City        3.368614
11               Customer Zipcode        3.153016
15                  Order Country        3.039082
28                 order_week_day        2.548759
19                   Order Region        2.318427
25             Customer Full Name        2.285246
9                  Customer State        2.260947
29                     order_hour        2.237262
3               Benefit per order        1.835991
2   Days for shipment (scheduled)        1.767365
4              Sales per customer        1.539904
16            Order Item Discount        1.429603
21                   Product Name        1.226367
13                         Market        1.048851
17       Order Item Discount Rate        1.043445
8                Customer Segment        1.028261
23                  Shipping Mode        1.019703
27                    order_month        1.014407
24                  Order Zipcode        0.864625
5                   Category Name        0.821124
12                Department Name        0.549274
18            Order Item Quantity        0.415214
7                Customer Country        0.321729
22                  Product Price        0.000000
```



4.7 Ensemble Learning

Ensemble methods were explored to combine predictions from multiple models. By leveraging the collective intelligence of multiple classifiers, we aimed to enhance overall prediction accuracy.

5.1 Conclusion

The application of machine learning models for fraud detection in supply chain logistics data demonstrates significant promise. Through the evaluation of various classifiers, including logistic regression, ADA Boost, K-nearest neighbors, and random forests, we have gained valuable insights into their performance. Key findings from our analysis include:

- Random forests outperformed logistic regression and K-nearest neighbors, achieving high accuracy and recall rates.
- Feature importance analysis highlighted the role of specific features such as “TYPE”,"Days for Shipping (Real)," "Customer Street," and "Late Delivery" in fraud detection.

Overall, the results suggest that supply chain logistics companies can effectively employ machine learning models, particularly decision trees and random forests, for fraud detection. These models can help identify fraudulent activities, ultimately reducing financial losses and improving operational efficiency.

## 5.2 Recommendations

Based on our analysis, we offer the following recommendations for supply chain logistics companies:

1. Implement and Random Forest Models : Consider implementing decision tree and random forest models as primary tools for fraud detection. These models exhibit strong performance in terms of accuracy and recall.

2. Enhance Data Collection : Continue to gather comprehensive and high-quality data. The success of machine learning models is highly dependent on the quality and quantity of data.

3. Investigate Shipping Durations : Pay close attention to the actual days for shipping (real) and late delivery rates. Monitor and address outliers and discrepancies to reduce potential fraud.

4. Geospatial Analysis : Further investigate the significance of "Customer Street" in fraud detection. Geospatial analysis could provide insights into specific locations or regions with a higher likelihood of fraud.

5. Continuous Monitoring : Implement real-time monitoring systems to promptly detect and respond to potential fraudulent activities.

7. Ensemble Learning : Consider ensemble learning techniques that combine predictions from multiple models. This can enhance the overall accuracy of fraud detection.

8. Education and Training : Train personnel and stakeholders in recognizing fraudulent activities and how to use the machine learning models effectively.

9. Collaboration : Collaborate with data scientists and analysts to continuously improve and adapt the fraud detection models as new data becomes available.

# 6. Future Work

To further enhance the fraud detection capabilities, supply chain logistics companies may consider the following avenues for future research and development:

- Implement advanced anomaly detection algorithms to identify unusual patterns in the data.
- Explore natural language processing (NLP) techniques for processing unstructured text data, such as customer feedback or communication records, to gain additional insights.
- Investigate the use of deep learning models, like recurrent neural networks (RNNs) or convolutional neural networks (CNNs), for more complex fraud detection tasks.

In conclusion, the application of machine learning in fraud detection within supply chain logistics is a powerful tool for safeguarding operations, reducing financial losses, and ensuring the integrity of the supply chain. By implementing the recommendations and embracing future developments, logistics companies can stay ahead in the ongoing battle against fraudulent activities.

**Individual Contribution by the team:**

1. Sanidhya Chaudhary 20MIA1037: Methodology and Sampling Techniques
2. Sricharan Sridhar 20MIA1014: Abstract, Introduction, Documentation and Dataset collection
3. Gaurav Dwivedi 20MIA1037: Concepts Applied, Literature Survey and block diagrams.
4. Rohit Choudhary 20MIA1069: Model Implementation and coding along with Dataset collection along with mongo db implementation.

## 7. References

Lokanan, Mark & Maddhesia, Vikas Kumar. (2022). Supply Chain Fraud Prediction with Machine Learning and Artificial intelligence. 10.21203/rs.3.rs-1996324/v1.

Kumar, Sheo & Gunjan, Vinit & Ansari, Mohd Dilshad & Pathak, Rashmi. “Credit Card Fraud Detection Using Support Vector Machine.”/10.1007/978-981-16-6407-6_3/. (2022).

“A Machine Learning Model for Product Fraud Detection Based on SVM.” *Ieeexplore.ieee.org*, ieeexplore.ieee.org/document/9479632. Accessed 7 Nov. 2023.

Johannes, Raden, and Andry Alamsyah. *Sales Prediction Model Using Classification Decision Tree Approach for Small Medium Enterprise Based on Indonesian E-Commerce Data*.

L. Patterson, James , et al. “Understanding and Mitigating Supply Chain Fraud.” *Journal of Marketing Development and Competitiveness*, vol. 12, no. 1, 1 May 2018, https://doi.org/10.33423/jmdc.v12i1.1411. Accessed 12 Sept. 2019.

Seify, Mahdi, et al. “Fraud Detection in Supply Chain with Machine Learning.” *IFAC-PapersOnLine*, vol. 55, no. 10, 2022, pp. 406–411, https://doi.org/10.1016/j.ifacol.2022.09.427.