Fraud Detection Through Supervised Learning In The E-Commerce Industry

David Mwasikira, Ilham Seladji, Jamal Alshanableh, Khalid Abbas, Khansa Pathan, Zenah Alzubaidi

November 15th, 2020

I. Introduction:

1. Motivation:

Since the e-commerce industry has been booming for a couple of decades now, with companies such as Amazon being on top of the list, it is worthwhile and intriguing to analyze datasets related to the giant business model of e-commerce. We can gain insights into any e-commerce company's hottest selling products, maximum profits, continents and countries with maximum number of customers, months of the year that yield the highest sales, etc and significantly scale our business.

2. Problem Statement:

E-commerce payment fraud is a very common problem that has been prevalent right from the launch of e-commerce platforms. Ever since businesses discovered a way through which customers could safely buy their products without visiting a physical store, hackers have done their best to steal that information and benefit from it. It is very crucial for online businesses to identify fraudulent transactions from the genuine ones the moment they crop up in order to:

1. Reduce their losses and increase profits substantially.
2. Incorrectly identifying a genuine transaction to be a fraudulent one can unnecessarily delay a shipment that might make the e-commerce store lose a good customer and end up with a bad review.

3. Function of the system:

The main function of our classification system is to analyze the supply chain of orders with their associated costs and classify the transactions that are likely to be fraudulent against genuine transactions in addition to identifying gaps that drive fraudulent transactions and take the necessary steps to eliminate them in order to optimize the performance of e-commerce services, using supervised learning.

4. Goals and Beneficiaries:

Goals:

1. Recognize fraudsters from legitimate clients, based on thousands of previous transactions.
2. Help e-commerce businesses to develop a tailored Fraud risk management program.
3. Identify fraudulent activities which block e-commerce platforms.
4. Ultimately save e-commerce businesses great amounts of money.

Beneficiaries:

1. E-commerce Companies: Fraud detection can significantly help the e-commerce industry to conduct its business smoothly without worrying about ambiguous transactions and losing profits in the process.
2. Customers: It can also safeguard the interest of e-commerce customers from being robbed of their personal/credit card information and make illegal transactions on their behalf.

5. Candidate Algorithms:

The following classification algorithms will be used in our analysis:

1. K-Nearest neighbors: Classifies an example based on the majority class within the "K" nearest samples.
2. Neural Networks (Main Algorithm): Mimic the way the human brain operates. Neural networks process information from a layer to another by finding relationships between features and targets.
3. Random forest: Builds multiple decision trees in the training phase and outputs the majority class of those trees.
4. Support Vector Machines: Separate different classes with a hyperplane (or a number of hyperplanes) in such a way that inter-class distance is maximized.
5. Logistic Regression: Just like in a linear regression model, data is acted upon by a logistic function to predict a target categorical variable which can either be binary or multivariate.

Since neural networks are a part of deep learning techniques and are an advanced application of Machine Learning, we will use it as our core algorithm that we can fine tune to get optimal results and compare its performance with other machine learning algorithms.

6. Alternative Solutions:

1. Using different algorithms like Linear Discriminant Analysis, Gaussian Naive Bayes, Extra Trees classification, Extreme Gradient Boosting and metrics like recall, F1 score, precision, log loss and Mean Absolute Error (MAE).
2. Comparing ML algorithms with several Deep Learning networks such as MLPs and CNNs.
3. Probability-predicting regression model can be used as part of a classifier by imposing a decision rule

7. Related Work:

Various ML algorithms were implemented for this data set and compared to each other using metrics such as RMSE and MAE, in Python.

II. Data Exploration:

The dataset used in this project is the **DataCo Smart Supply Chain [1]**. It is specific to the **DataCo Global** company, and it encompasses information about 180,519 orders and shipments across different continents for various categories of products such as *sportswear, footwear, electronics and medical equipment.*

Those orders are characterized by 53 numerical and categorical features, among which the **Order Status** is the target variable. The most important features are described in the following data dictionary:

Table 1: Table continues below

Field Name	Description	Type
Type	Type of transaction made	chr
Benefit per order	Earnings per order placed	num

Field Name	Description	Type
Sales per customer	Total sales per customer	num
Category ID	Product category code	int
Category Name	Description of the product category	chr
Customer City	City where the customer made the purchase	chr
Customer Country	Country where the customer made the purchase	chr
Customer Segment	Types of Customers: Consumer , Corporate , Home Office	chr
Department ID	Department code of store	int
Department Name	Department name of store	chr
Latitude	Latitude corresponding to location of store	num
Longitude	Longitude corresponding to location of store	num
Market	Market to where the order is delivered : Africa , Europe , LATAM , Pacific Asia , USCA	chr
Order Country	Destination country of the order	chr
Order Customer ID	Customer order code	int
Order Date	Date on which the order is made	chr
Order ID	Order code	int
Order Item Discount	Order item discount value	num
Order Item Discount Rate	Order item discount percentage	num
Order Item Id	Order item code	int
Order Item Product Price	Price of products without discount	num
Order Item Profit Ratio	Profit Ratio of an Item in an Order	num
Order Item Quantity	Number of products per order	int
Sales	Value in sales	num
Order Item Total	Total amount per order	num
Order Profit Per Order	Profit of an Order	num
Order Region	Region of the world where the order is delivered : Southeast Asia ,South Asia ,Oceania ,Eastern Asia, West Asia , West of USA , US Center , West Africa, Central Africa ,North Africa ,Western Europe ,Northern , Caribbean , South America ,East Africa ,Southern Europe , East of USA ,Canada ,Southern Africa , Central Asia , Europe , Central America, Eastern Europe , South of USA	chr
Order State	State of the region where the order is delivered	chr
Product Card Id	Product code	chr
Product Category Id	Product category code	int
Product Name	Name of Product	int
Product Price	Price of Product	chr
Product Status	Status of the product stock: 1 if the product is not available, 0 if it is available	num
Shipping date (DateOrders)	Exact date and time of shipment	int
Shipping Mode	The following shipping modes are presented : Standard Class , First Class , Second Class , Same Day	chr
Late delivery risk	Categorical variable that indicates if sending is late (1) or not (0)	chr

Field Name	Description	Type
Delivery Status	Delivery status of orders: Advance shipping , Late delivery , Shipping canceled , Shipping on time	int
Order Status (Target Attribute)	Order Status : COMPLETE , PENDING , CLOSED , PENDING_PAYMENT ,CANCELED , PROCESSING ,SUSPECTED_FRAUD ,ON_HOLD ,PAYMENT_REVIEW	chr

Examples
Debit, Transfer, Cash
91.2, -249.1
315, 311
73
Sporting Goods, Cameras
Caguas, San Jose
Puerto Rico, USA
Consumer, Home Office
2
Fitness
18.3
-66
Pacific Asia
Indonesia, Australia
20755
1/31/2018 22:56
77202
13.1
0.04
180517
328
0.29
1
328
315
91.2
Southeast Asia
Queensland, Java Occidental
1360
73
Smart Watch
328
0, 1
2/3/2018 22:56
Standard Class, First Class, Same Day, Second Class
0, 1
Advance Shipping, Late Delivery
Complete, Pending, Closed

1. Understanding The Data: Preliminary Data Inspection:

Preview of The Original Dataset:

Table 3: Table continues below

Type	Days.for.shipping.(real)	Days.for.shipment.(scheduled)
DEBIT	3	4
TRANSFER	5	4
CASH	4	4
DEBIT	3	4
PAYMENT	2	4
TRANSFER	6	4

Table 4: Table continues below

Benefit.per.order	Sales.per.customer	Delivery.Status
91.25	314.6	Advance shipping
-249.1	311.4	Late delivery
-247.8	309.7	Shipping on time
22.86	304.8	Advance shipping
134.2	298.2	Advance shipping
18.58	295	Shipping canceled

Table 5: Table continues below

Late_delivery_risk	Category.Id	Category.Name	Customer.City
0	73	Sporting Goods	Caguas
1	73	Sporting Goods	Caguas
0	73	Sporting Goods	San Jose
0	73	Sporting Goods	Los Angeles
0	73	Sporting Goods	Caguas
0	73	Sporting Goods	Tonawanda

Table 6: Table continues below

Customer.Country	Customer.Email	Customer.Fname	Customer.Id
Puerto Rico	XXXXXXXXX	Cally	20755
Puerto Rico	XXXXXXXXX	Irene	19492
EE. UU.	XXXXXXXXX	Gillian	19491
EE. UU.	XXXXXXXXX	Tana	19490
Puerto Rico	XXXXXXXXX	Orli	19489
EE. UU.	XXXXXXXXX	Kimberly	19488

Table 7: Table continues below

Customer.Lname	Customer.Password	Customer.Segment	Customer.State
Holloway	XXXXXXXXX	Consumer	PR

Customer.Lname	Customer.Password	Customer.Segment	Customer.State
Luna	XXXXXXXXX	Consumer	PR
Maldonado	XXXXXXXXX	Consumer	CA
Tate	XXXXXXXXX	Home Office	CA
Hendricks	XXXXXXXXX	Corporate	PR
Flowers	XXXXXXXXX	Consumer	NY

Table 8: Table continues below

Customer.Street	Customer.Zipcode	Department.Id	Department.Name
5365 Noble Nectar Island	725	2	Fitness
2679 Rustic Loop	725	2	Fitness
8510 Round Bear Gate	95125	2	Fitness
3200 Amber Bend	90027	2	Fitness
8671 Iron Anchor Corners	725	2	Fitness
2122 Hazy Corner	14150	2	Fitness

Table 9: Table continues below

Latitude	Longitude	Market	Order.City	Order.Country
18.25	-66.04	Pacific Asia	Bekasi	Indonesia
18.28	-66.04	Pacific Asia	Bikaner	India
37.29	-121.9	Pacific Asia	Bikaner	India
34.13	-118.3	Pacific Asia	Townsville	Australia
18.25	-66.04	Pacific Asia	Townsville	Australia
43.01	-78.88	Pacific Asia	Toowoomba	Australia

Table 10: Table continues below

Order.Customer.Id	order.date.(DateOrders)	Order.Id
20755	1/31/2018 22:56	77202
19492	1/13/2018 12:27	75939
19491	1/13/2018 12:06	75938
19490	1/13/2018 11:45	75937
19489	1/13/2018 11:24	75936
19488	1/13/2018 11:03	75935

Table 11: Table continues below

Order.Item.Cardprod.Id	Order.Item.Discount	Order.Item.Discount.Rate
1360	13.11	0.04
1360	16.39	0.05
1360	18.03	0.06
1360	22.94	0.07
1360	29.5	0.09
1360	32.78	0.1

Table 12: Table continues below

Order.Item.Id	Order.Item.Product.Price	Order.Item.Profit.Ratio
180517	327.8	0.29
179254	327.8	-0.8
179253	327.8	-0.8
179252	327.8	0.08
179251	327.8	0.45
179250	327.8	0.06

Table 13: Table continues below

Order.Item.Quantity	Sales	Order.Item.Total	Order.Profit.Per.Order
1	327.8	314.6	91.25
1	327.8	311.4	-249.1
1	327.8	309.7	-247.8
1	327.8	304.8	22.86
1	327.8	298.2	134.2
1	327.8	295	18.58

Table 14: Table continues below

Order.Region	Order.State	Order.Status	Order.Zipcode
Southeast Asia	Java Occidental	COMPLETE	NA
South Asia	RajastÃ¡n	PENDING	NA
South Asia	RajastÃ¡n	CLOSED	NA
Oceania	Queensland	COMPLETE	NA
Oceania	Queensland	PENDING_PAYMENT	NA
Oceania	Queensland	CANCELED	NA

Table 15: Table continues below

Product.Card.Id	Product.Category.Id	Product.Description
1360	73	NA
1360	73	NA
1360	73	NA
1360	73	NA
1360	73	NA
1360	73	NA

Table 16: Table continues below

Product.Image	Product.Name	Product.Price
http://images.acmesports.sports/Smart+watch	Smart watch	327.8
http://images.acmesports.sports/Smart+watch	Smart watch	327.8
http://images.acmesports.sports/Smart+watch	Smart watch	327.8
http://images.acmesports.sports/Smart+watch	Smart watch	327.8

Product.Image	Product.Name	Product.Price
http://images.acmesports.sports/Smart+watch	Smart watch	327.8
http://images.acmesports.sports/Smart+watch	Smart watch	327.8

Product.Status	shipping.date.(DateOrders)	Shipping.Mode
0	2/3/2018 22:56	Standard Class
0	1/18/2018 12:27	Standard Class
0	1/17/2018 12:06	Standard Class
0	1/16/2018 11:45	Standard Class
0	1/15/2018 11:24	Standard Class
0	1/19/2018 11:03	Standard Class

As we can see, some features are duplicated (e.g. *Category ID* and *Category Name*) and others carry sensitive information which has been hidden (e.g. *Customer Email* and *Customer Password*). Hence, they will be dropped.

2. Data Cleaning:

2.1. Dropping Columns by Column name:

The following features are irrelevant to our analysis at this stage and should therefore be dropped: *Customer City, Customer Country, Customer Email, Customer Fname, Customer Lname, Customer Password, Customer Street, Order Item Cardprod ID, Order State, Order Zipcode, Product Category Id, Product Description, Product Image, Product Name, Customer State, Order Customer ID, Order ID, Customer Zipcode, Department Name.*

The remaining columns will be reviewed based on our analysis of the importance and weight of each attribute towards working out our solution.

Table 18: Table continues below

Type	Days.for.shipping.(real)	Days.for.shipment.(scheduled)
DEBIT	3	4
TRANSFER	5	4
CASH	4	4
DEBIT	3	4
PAYMENT	2	4
TRANSFER	6	4

Table 19: Table continues below

Benefit.per.order	Sales.per.customer	Delivery.Status
91.25	314.6	Advance shipping
-249.1	311.4	Late delivery
-247.8	309.7	Shipping on time
22.86	304.8	Advance shipping
134.2	298.2	Advance shipping
18.58	295	Shipping canceled

Table 20: Table continues below

Late_delivery_risk	Category.Id	Category.Name	Customer.Id
0	73	Sporting Goods	20755
1	73	Sporting Goods	19492
0	73	Sporting Goods	19491
0	73	Sporting Goods	19490
0	73	Sporting Goods	19489
0	73	Sporting Goods	19488

Table 21: Table continues below

Customer.Segment	Department.Id	Latitude	Longitude	Market
Consumer	2	18.25	-66.04	Pacific Asia
Consumer	2	18.28	-66.04	Pacific Asia
Consumer	2	37.29	-121.9	Pacific Asia
Home Office	2	34.13	-118.3	Pacific Asia
Corporate	2	18.25	-66.04	Pacific Asia
Consumer	2	43.01	-78.88	Pacific Asia

Table 22: Table continues below

Order.City	Order.Country	order.date.(DateOrders)	Order.Item.Discount
Bekasi	Indonesia	1/31/2018 22:56	13.11
Bikaner	India	1/13/2018 12:27	16.39
Bikaner	India	1/13/2018 12:06	18.03
Townsville	Australia	1/13/2018 11:45	22.94
Townsville	Australia	1/13/2018 11:24	29.5
Toowoomba	Australia	1/13/2018 11:03	32.78

Table 23: Table continues below

Order.Item.Discount.Rate	Order.Item.Id	Order.Item.Product.Price
0.04	180517	327.8
0.05	179254	327.8
0.06	179253	327.8
0.07	179252	327.8
0.09	179251	327.8
0.1	179250	327.8

Table 24: Table continues below

Order.Item.Profit.Ratio	Order.Item.Quantity	Sales	Order.Item.Total
0.29	1	327.8	314.6
-0.8	1	327.8	311.4
-0.8	1	327.8	309.7
0.08	1	327.8	304.8
0.45	1	327.8	298.2

Order.Item.Profit.Ratio	Order.Item.Quantity	Sales	Order.Item.Total
0.06	1	327.8	295

Table 25: Table continues below

Order.Profit.Per.Order	Order.Region	Order.Status	Product.Card.Id
91.25	Southeast Asia	COMPLETE	1360
-249.1	South Asia	PENDING	1360
-247.8	South Asia	CLOSED	1360
22.86	Oceania	COMPLETE	1360
134.2	Oceania	PENDING_PAYMENT	1360
18.58	Oceania	CANCELED	1360

Product.Price	Product.Status	shipping.date.(DateOrders)	Shipping.Mode
327.8	0	2/3/2018 22:56	Standard Class
327.8	0	1/18/2018 12:27	Standard Class
327.8	0	1/17/2018 12:06	Standard Class
327.8	0	1/16/2018 11:45	Standard Class
327.8	0	1/15/2018 11:24	Standard Class
327.8	0	1/19/2018 11:03	Standard Class

2.2. Checking for missing values in the dataset:

Columns with missing values have already been dropped in the previous stage. Thus, the data is complete for future processing.

2.3. Data Summary:

The following summary shows the most extreme values (minimum and maximum), lower and upper quartiles, mean and median of each numerical feature. They give a better understanding of what kind of values we are dealing with:

```
##      Type           Days.for.shipping.(real) Days.for.shipment.(scheduled)
##  Length:180519      Min.   :0.000            Min.   :0.000
##  Class :character   1st Qu.:2.000            1st Qu.:2.000
##  Mode  :character   Median :3.000            Median :4.000
##                     Mean   :3.498            Mean   :2.932
##                     3rd Qu.:5.000            3rd Qu.:4.000
##                     Max.   :6.000            Max.   :4.000
##  Benefit.per.order  Sales.per.customer Delivery.Status    Late_delivery_risk
##  Min.   :-4274.98   Min.   :   7.49    Length:180519      Min.   :0.0000
##  1st Qu.:    7.00   1st Qu.: 104.38    Class :character   1st Qu.:0.0000
##  Median :   31.52   Median : 163.99    Mode  :character   Median :1.0000
##  Mean   :   21.98   Mean   : 183.11                       Mean   :0.5483
##  3rd Qu.:   64.80   3rd Qu.: 247.40                       3rd Qu.:1.0000
##  Max.   :  911.80   Max.   :1939.99                       Max.   :1.0000
##   Category.Id    Category.Name       Customer.Id    Customer.Segment
##  Min.   : 2.00   Length:180519      Min.   :    1   Length:180519
##  1st Qu.:18.00   Class :character   1st Qu.: 3258   Class :character
##  Median :29.00   Mode  :character   Median : 6457   Mode  :character
```

```
##  Mean   :31.85                     Mean   : 6691
##  3rd Qu.:45.00                     3rd Qu.: 9779
##  Max.   :76.00                     Max.   :20757
##  Department.Id       Latitude        Longitude          Market
##  Min.   : 2.000   Min.   :-33.94   Min.   :-158.03   Length:180519
##  1st Qu.: 4.000   1st Qu.: 18.27   1st Qu.: -98.45   Class :character
##  Median : 5.000   Median : 33.14   Median : -76.85   Mode  :character
##  Mean   : 5.443   Mean   : 29.72   Mean   : -84.92
##  3rd Qu.: 7.000   3rd Qu.: 39.28   3rd Qu.: -66.37
##  Max.   :12.000   Max.   : 48.78   Max.   : 115.26
##   Order.City        Order.Country      order.date.(DateOrders)
##  Length:180519      Length:180519      Length:180519
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##  Order.Item.Discount Order.Item.Discount.Rate Order.Item.Id
##  Min.   :  0.00      Min.   :0.0000           Min.   :     1
##  1st Qu.:  5.40      1st Qu.:0.0400           1st Qu.: 45131
##  Median : 14.00      Median :0.1000           Median : 90260
##  Mean   : 20.66      Mean   :0.1017           Mean   : 90260
##  3rd Qu.: 29.99      3rd Qu.:0.1600           3rd Qu.:135390
##  Max.   :500.00      Max.   :0.2500           Max.   :180519
##  Order.Item.Product.Price Order.Item.Profit.Ratio Order.Item.Quantity
##  Min.   :   9.99          Min.   :-2.7500         Min.   :1.000
##  1st Qu.:  50.00          1st Qu.: 0.0800         1st Qu.:1.000
##  Median :  59.99          Median : 0.2700         Median :1.000
##  Mean   : 141.23          Mean   : 0.1206         Mean   :2.128
##  3rd Qu.: 199.99          3rd Qu.: 0.3600         3rd Qu.:3.000
##  Max.   :1999.99          Max.   : 0.5000         Max.   :5.000
##      Sales         Order.Item.Total  Order.Profit.Per.Order Order.Region
##  Min.   :   9.99   Min.   :   7.49   Min.   :-4274.98       Length:180519
##  1st Qu.: 119.98   1st Qu.: 104.38   1st Qu.:    7.00       Class :character
##  Median : 199.92   Median : 163.99   Median :   31.52       Mode  :character
##  Mean   : 203.77   Mean   : 183.11   Mean   :   21.98
##  3rd Qu.: 299.95   3rd Qu.: 247.40   3rd Qu.:   64.80
##  Max.   :1999.99   Max.   :1939.99   Max.   :  911.80
##  Order.Status       Product.Card.Id  Product.Price     Product.Status
##  Length:180519      Min.   :  19.0   Min.   :   9.99   Min.   :0
##  Class :character   1st Qu.: 403.0   1st Qu.:  50.00   1st Qu.:0
##  Mode  :character   Median : 627.0   Median :  59.99   Median :0
##                     Mean   : 692.5   Mean   : 141.23   Mean   :0
##                     3rd Qu.:1004.0   3rd Qu.: 199.99   3rd Qu.:0
##                     Max.   :1363.0   Max.   :1999.99   Max.   :0
##  shipping.date.(DateOrders) Shipping.Mode
##  Length:180519              Length:180519
##  Class :character           Class :character
##  Mode  :character           Mode  :character
##
##
##
```

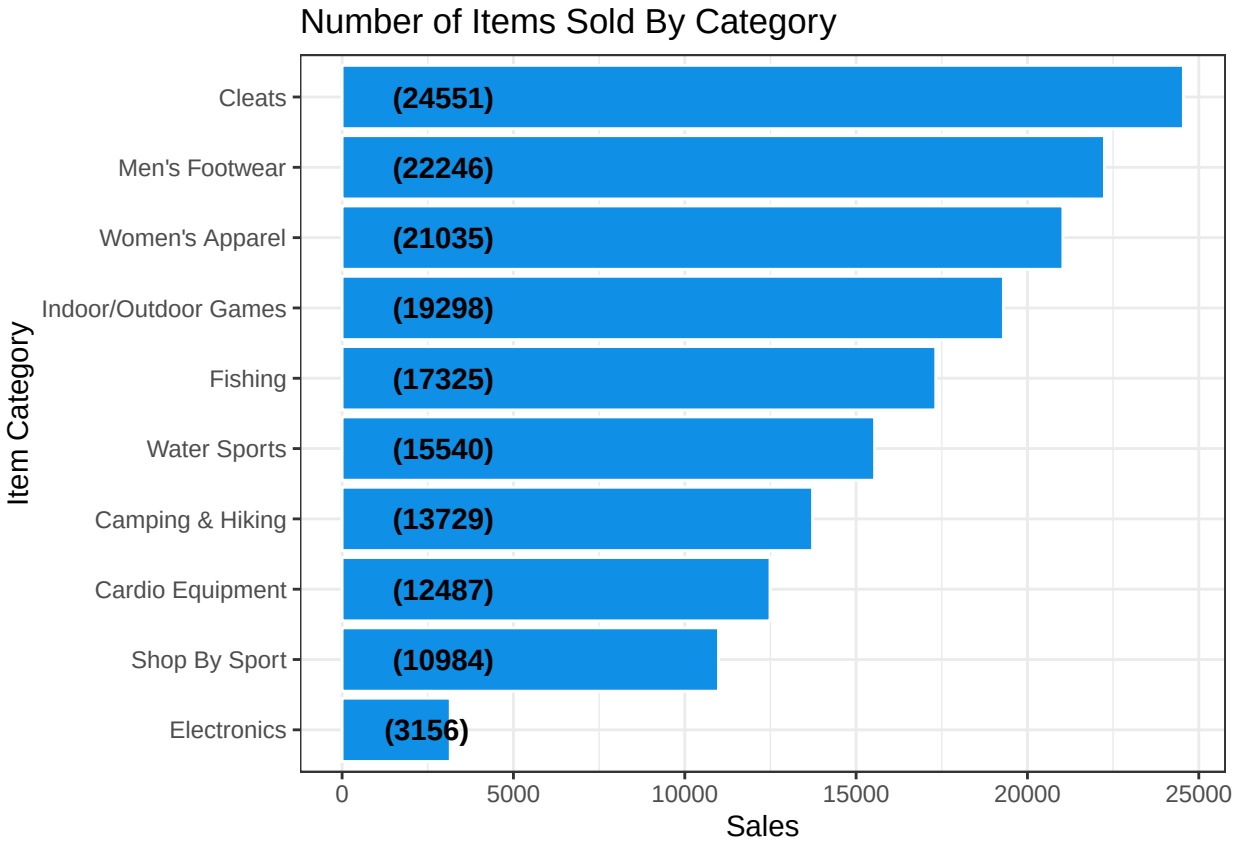
III. Data Analysis & Visualization:

1. Analysis of Order Delivery Performance:

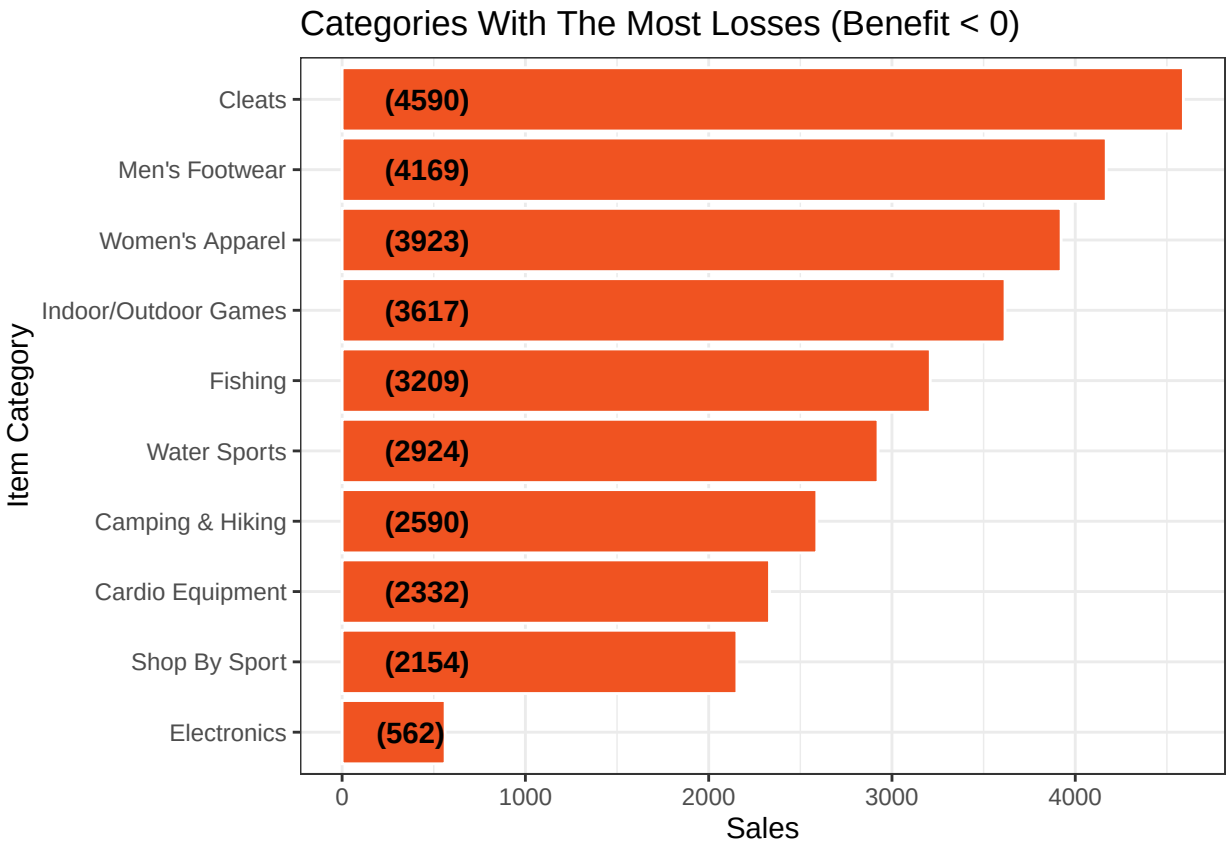
Here, we seek to answer several questions outlined below and try further to gain insights from the data, extracting valuable information:

1. Which Product Categories generate most sales? Do these sales always generate profits to the company?
2. Where are orders delivered?
3. What is the distribution of Late Deliveries compared to On Time and Advanced Deliveries?
4. What are the shipping modes used?
5. In which month(s) of the year are there higher orders?
6. What are the most common statuses of orders?
7. Which shipping modes suffer from late deliveries the most?
8. What are late deliveries caused by?
9. How many fraudulent transactions are there, compared to genuine ones?
10. Where are those frauds coming from?
11. Who are the biggest fraudsters?

1.1. Which Product Categories generate most sales? (Top 10 Item Categories Delivered):

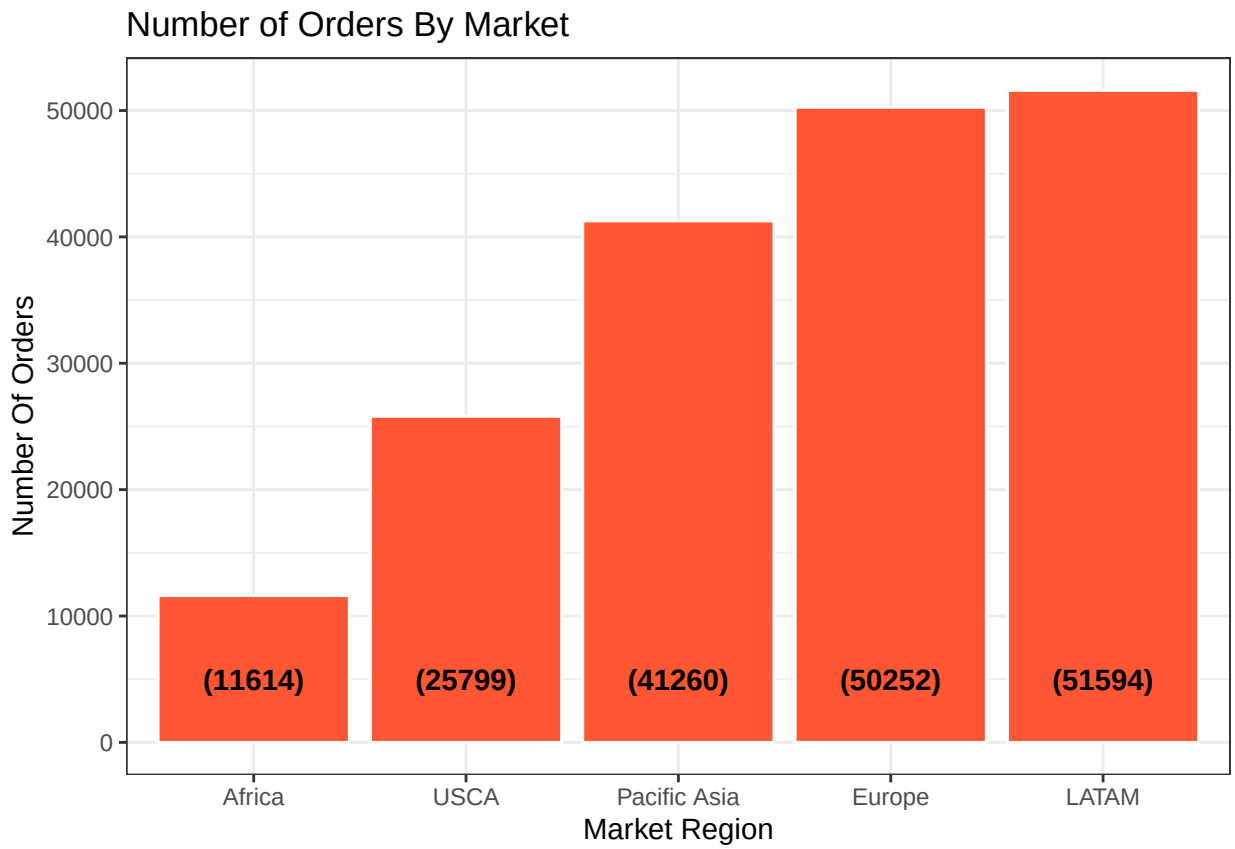


Cleats, Men's Footwear, Women's Apparel, Indoor/Outdoor Games, Fishing, Water Sports, Camping & Hiking, Cardio Equipment, Sporting Equipment and *Electronics* are the 10 most sold Item Categories by **DataCo Global**. But do they always generate profits to the company?



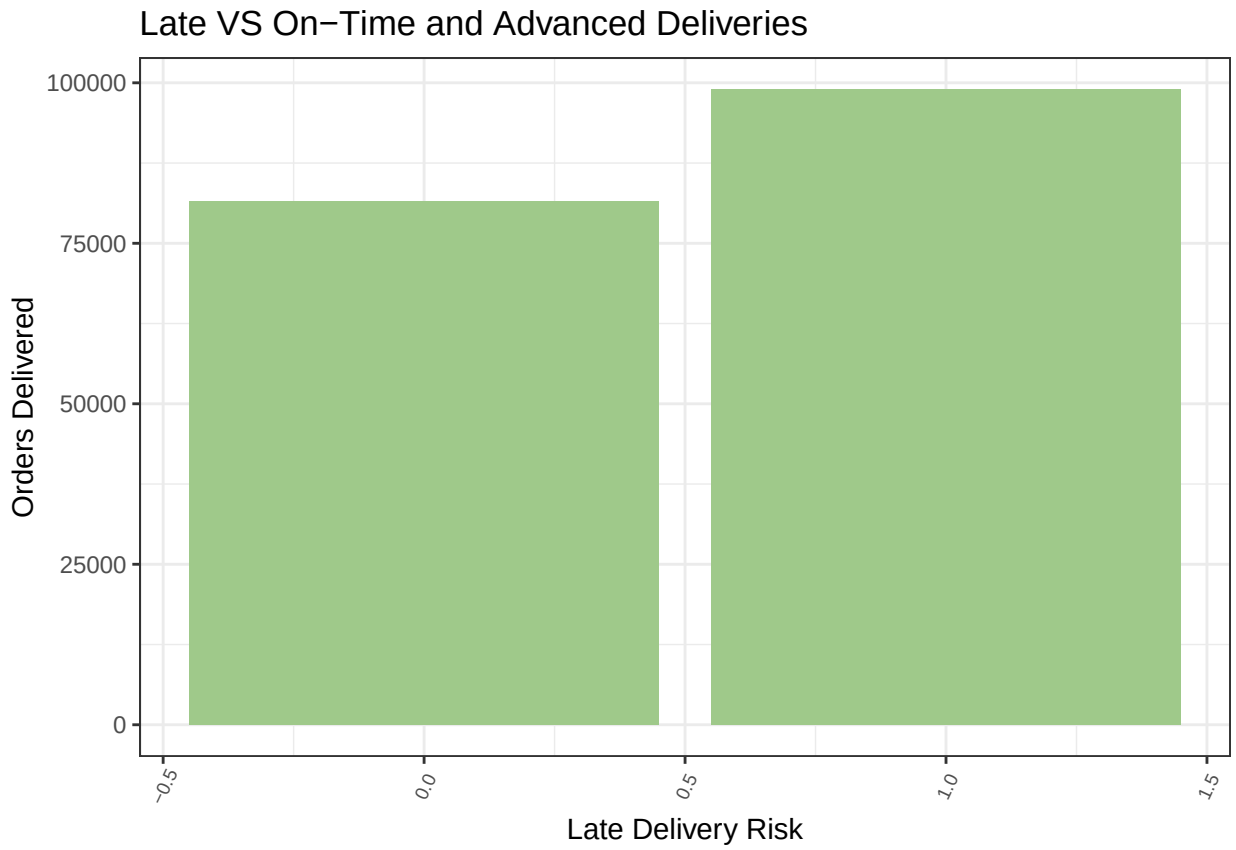
As we can see, the categories with most sales cause the highest losses to the company. For instance, 18.7% of *Cleats* sales, 18.74% of *Men's Footwear* sales and 18.65% of *Women's Apparel* sales generate losses to the company. These losses might be related to frauds, cancellations or discounts related to late deliveries.

1.2 Where are orders Delivered?:



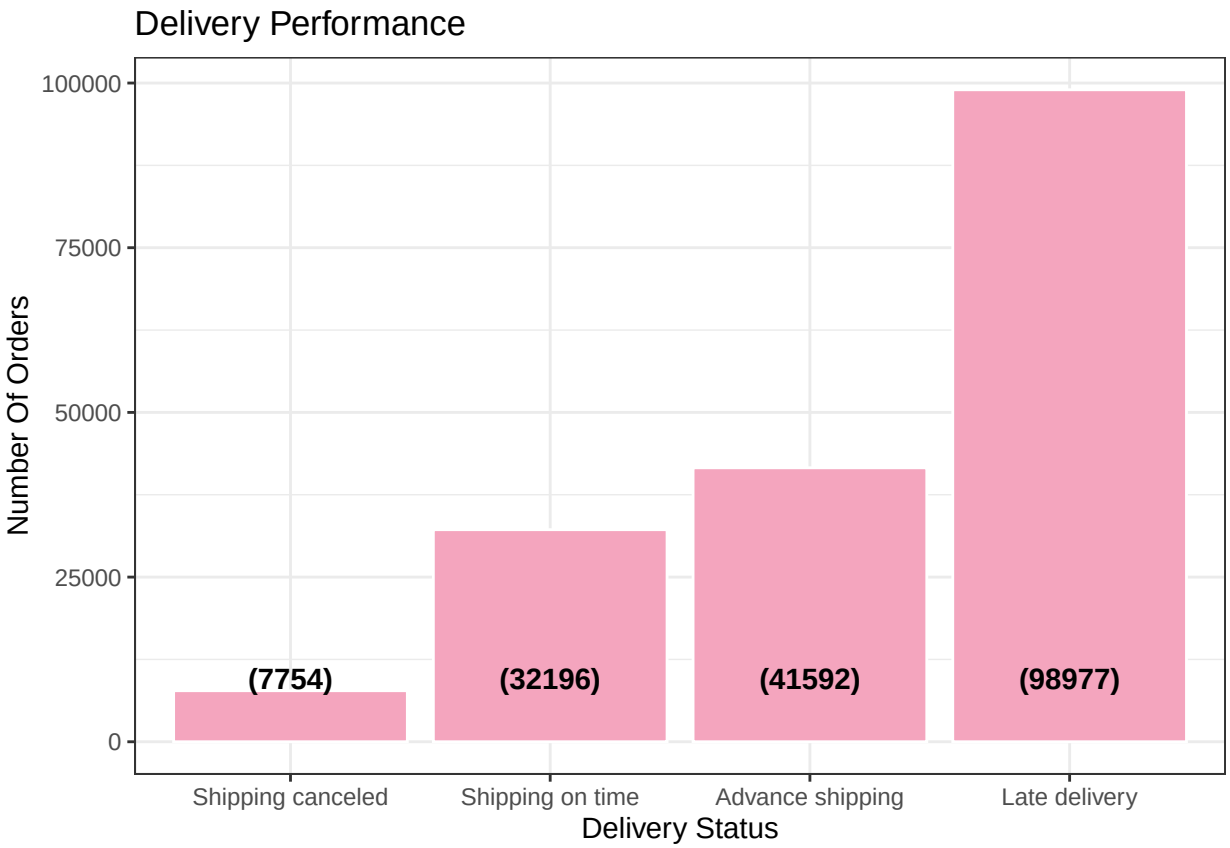
Products are delivered to *Africa, United States and Canada, Pacific Asia, Europe* and *Latin America*. 56.42% of those products are delivered to *Europe* and *Latin America*, with *Latin America* being the top delivery destination.

1.3. What is the distribution of Late Deliveries compared to On-Time and Advanced Deliveries?:

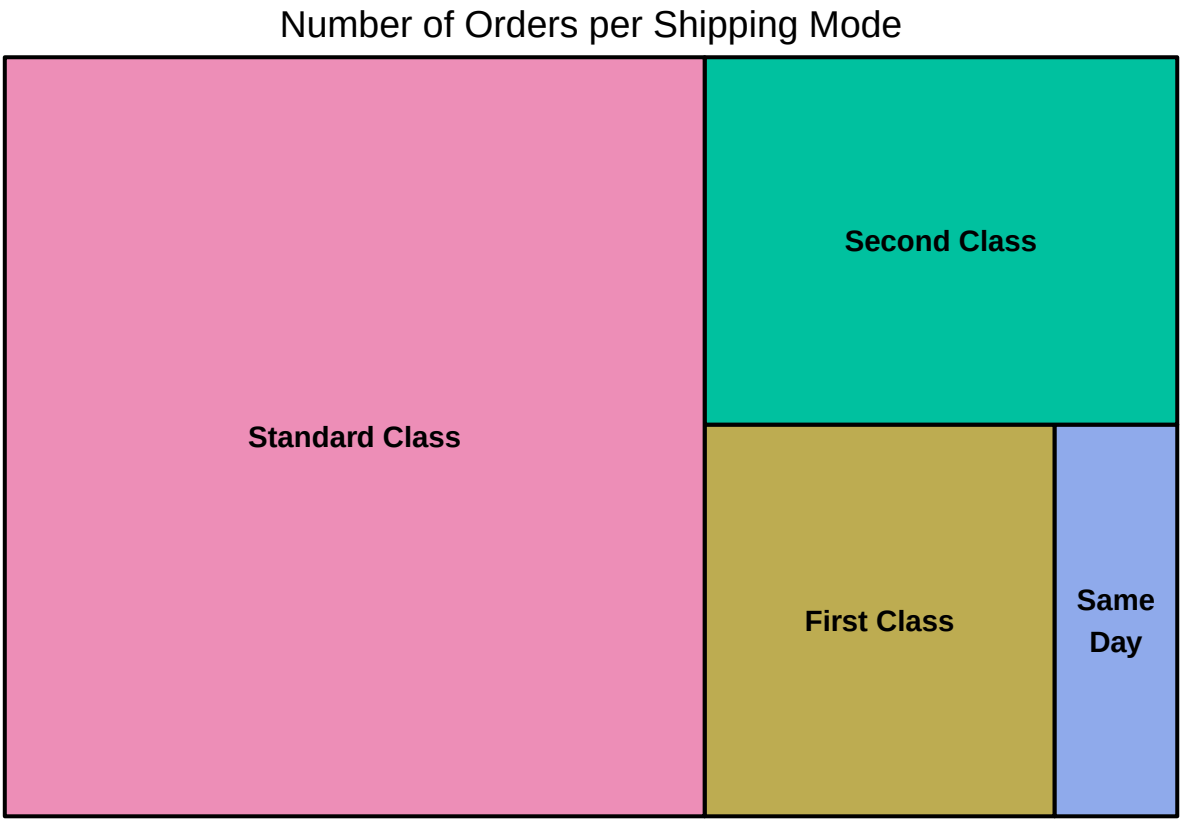


98,977 orders are not delivered on time, which represent 54.83% of the total number of orders. That is a huge and surprising number which needs some investigation to know what factors drive so many late deliveries, such as planning issues, misspelled addresses or even lost packages.

The remaining orders (45.17%) are not late (they can either be delivered in advance, on time or are counted towards canceled orders), as we can see in the following bar chart:

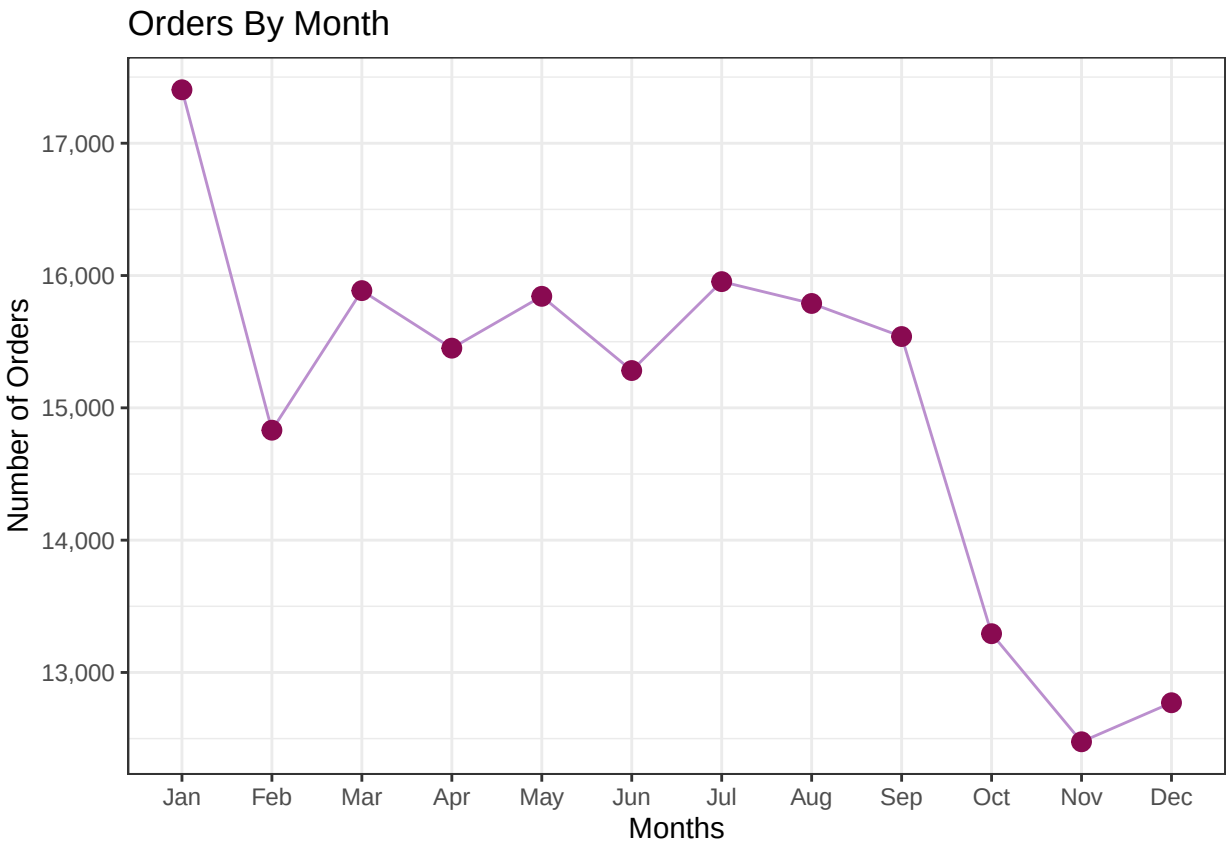

Delivery Performance
100000
75000
50000
25000
0
Number Of Orders
(7754)
(32196)
(41592)
(98977)
Shipping canceled
Shipping on time
Advance shipping
Late delivery
Delivery Status

1.4. What are the Shipping Modes used?:



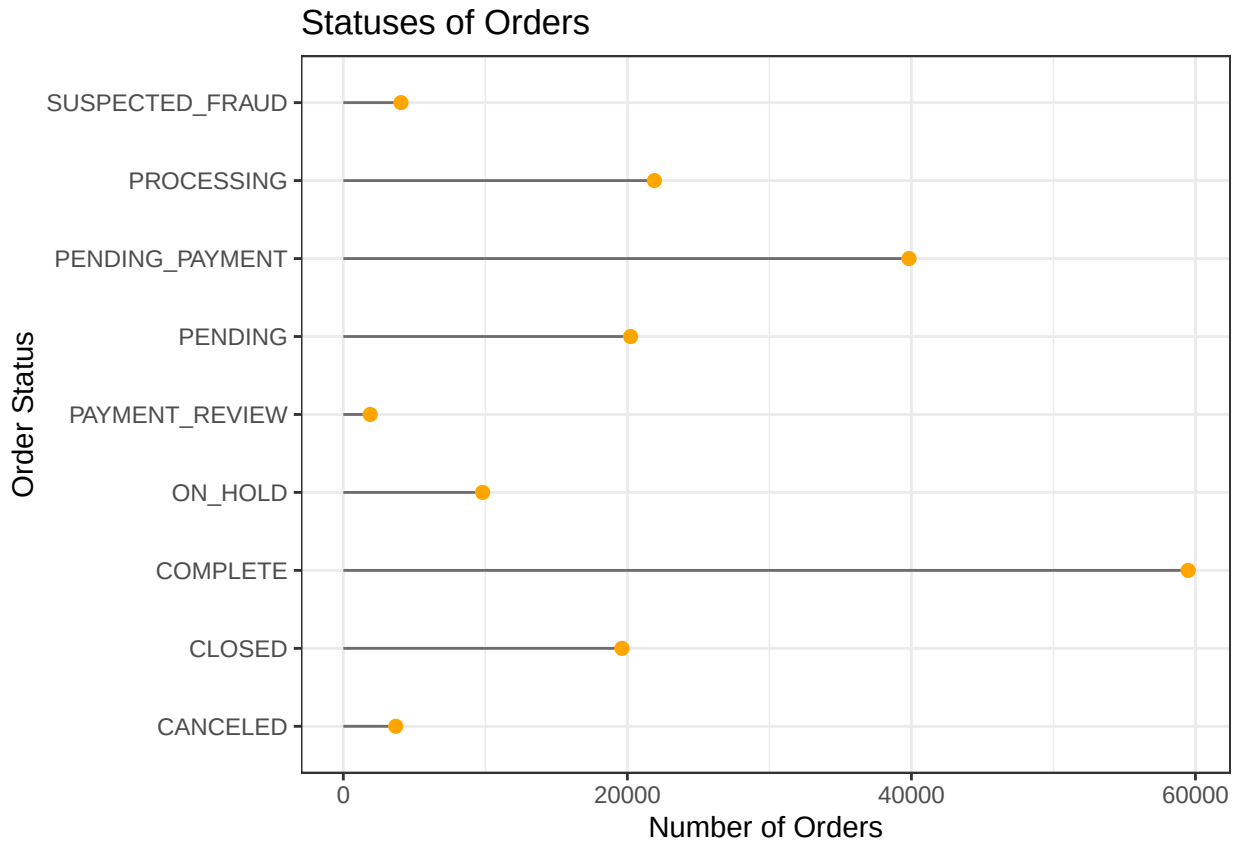
The most common shipping mode is the **Standard Class** mode followed by **Second Class**, **First Class** and **Same Day** Deliveries.

1.5. In which month(s) of the year are there higher orders?:



The highest number of orders are made in January. This can be caused by new year discounts or by the fact that a lot of companies launch and put their new products on the market at the beginning of the year.

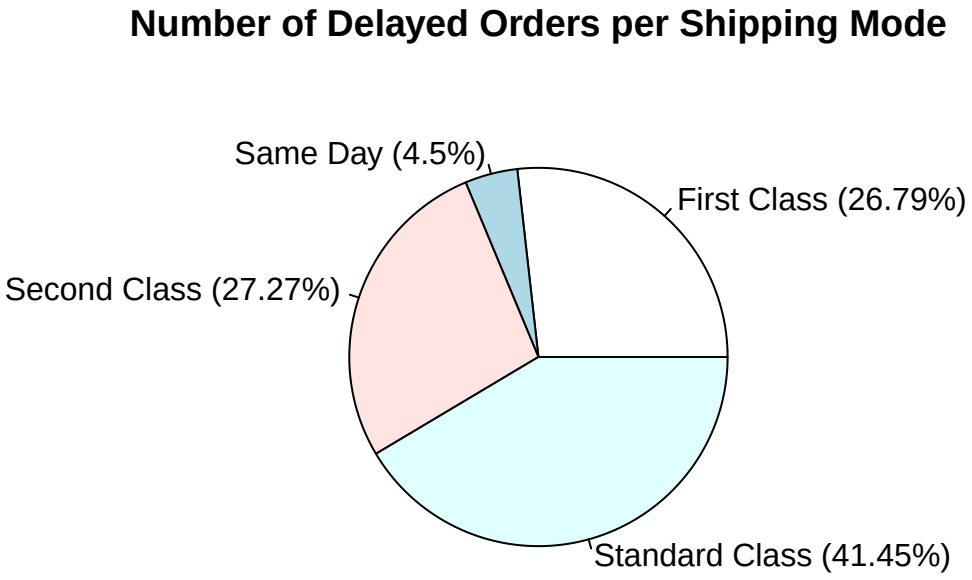
1.6. What are the most common statuses of orders?:



Most orders are **Complete** while a relatively important number of orders are **Pending Payment**. This opens the floor for many assumptions:

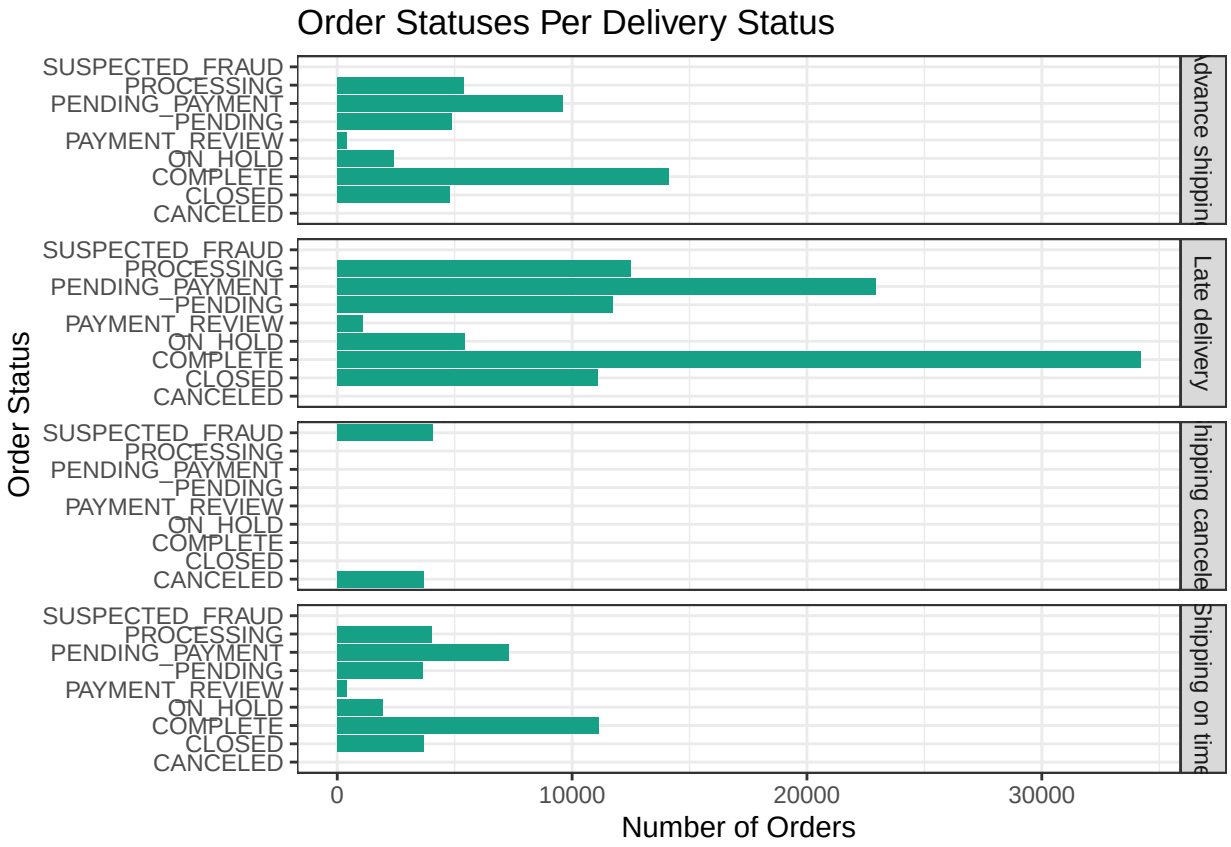
- Buyers hesitate to complete their purchases,
- Buyers want to maximize products in their carts in order to minimize orders and consequently minimize shipping fees,
- Some credit cards are rejected,
- Clients' banks consider those transactions as fraudulent,
- The company's online payment system is defective (unacknowledged transactions, security breaches, etc.).

1.7. Which shipping modes suffer from late deliveries the most?:



Standard Class and **Second Class** deliveries have the most number of late deliveries among other shipping modes, while **Same Day** and **First Class** deliveries have the least number of late deliveries. This is totally understandable since priority classes generally benefit from premium services for additional fees.

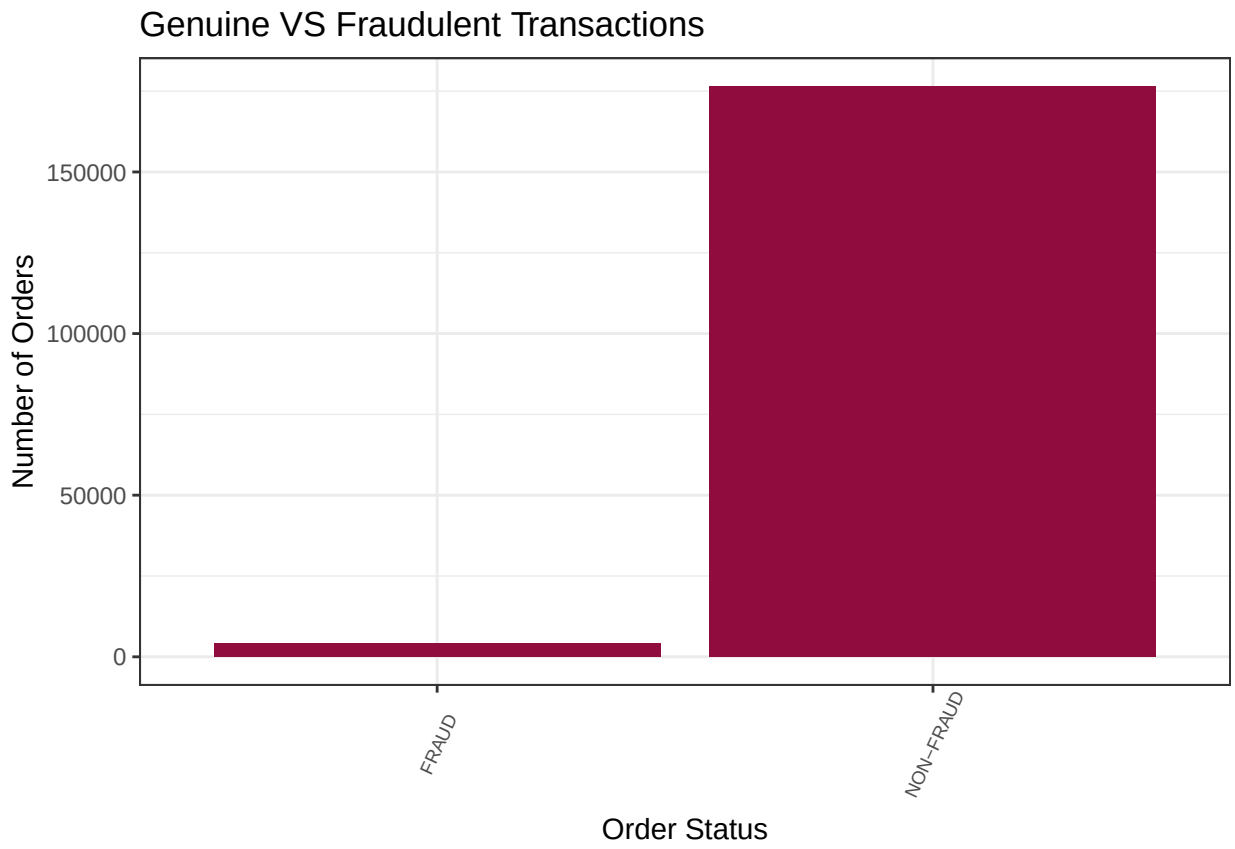
1.8. What are late deliveries caused by?:



34.55% of late deliveries are **Complete** and 23.16% of them are **Pending Payment**. *DataCo Global* need to put their focus on their payment system, since they have a lot of pending payments. **Advance Shippings** and **On-time Shippings** also have a good number of pending payments.

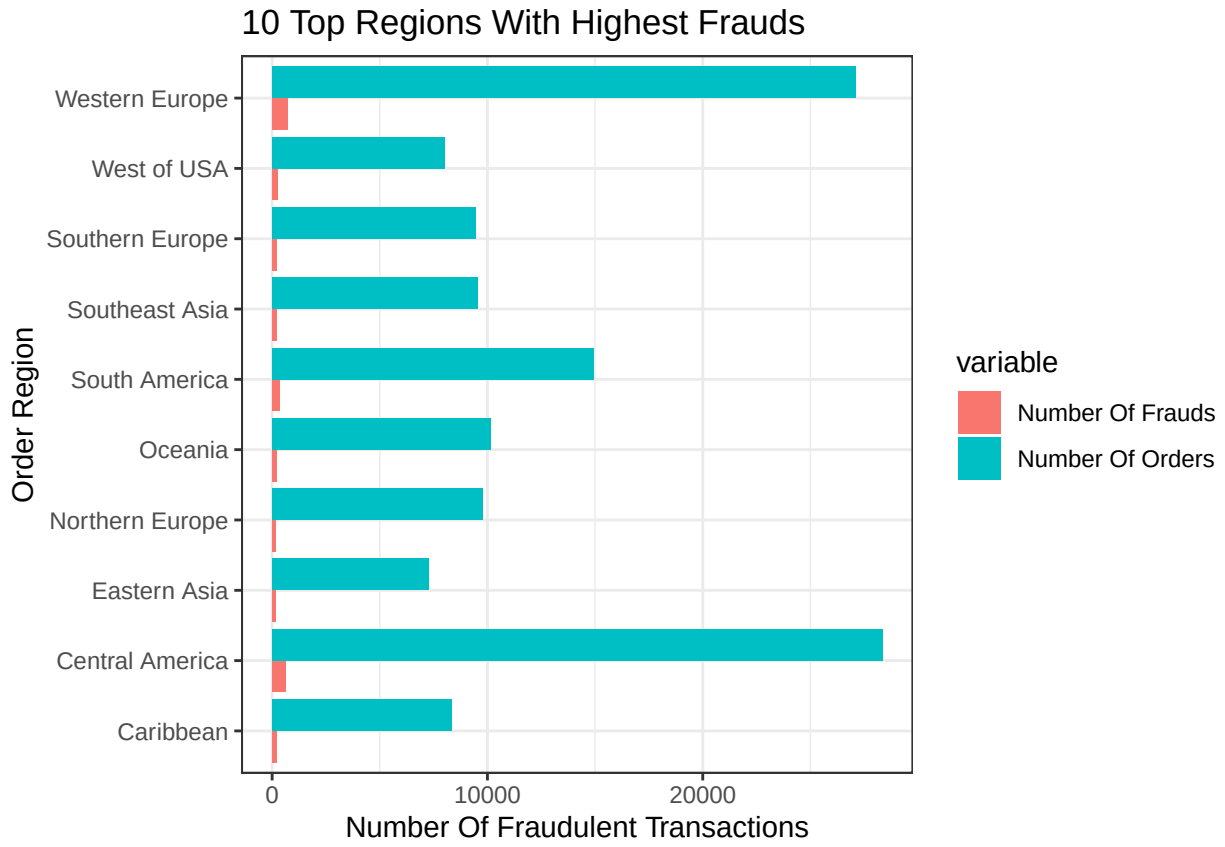
We can also see that orders with **Suspected Frauds** are automatically canceled.

1.9. How many fraudulent transactions are there, compared to genuine ones?:



Fraudulent transactions account for 2.3% of total transactions, which is somehow high and can cause the company huge losses in the long term.

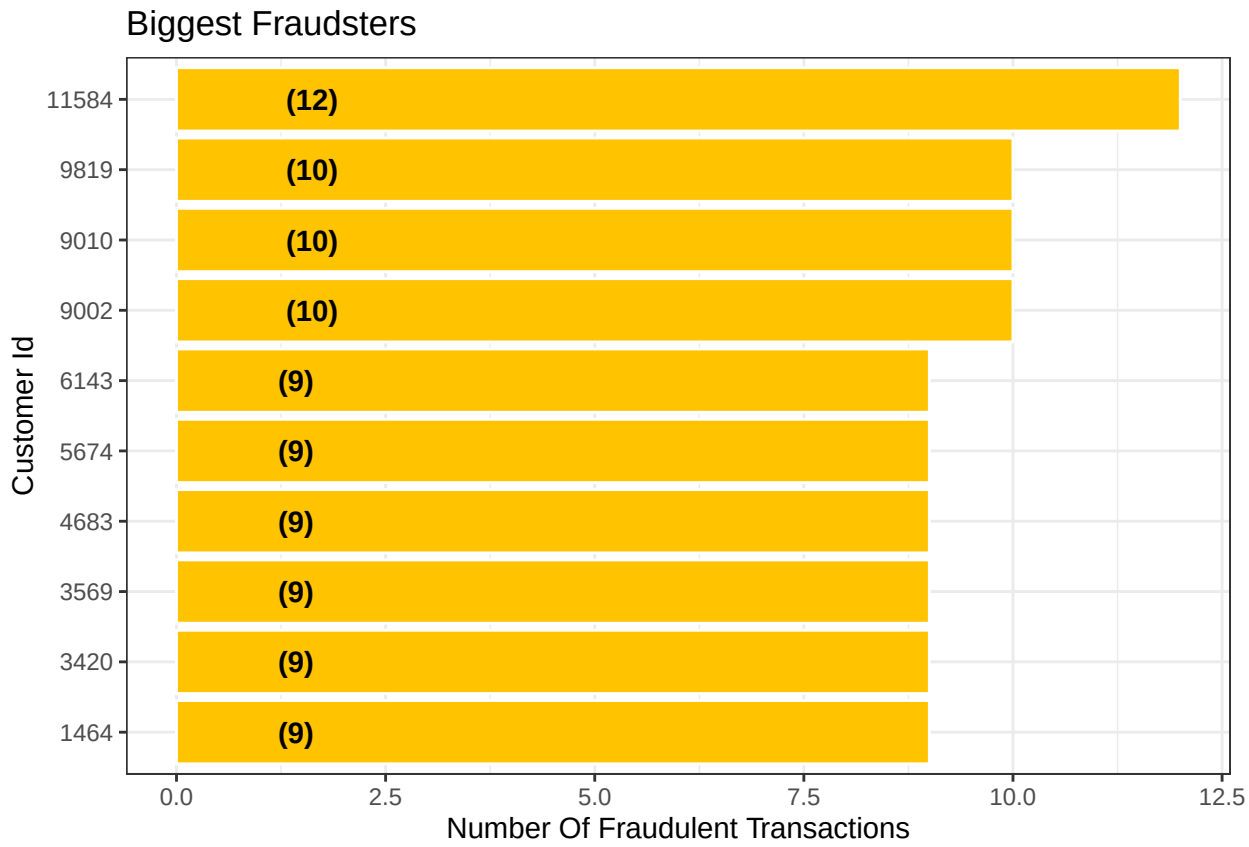
1.10. Where are those frauds coming from?:



We can see that the main source of frauds is Western Europe, with 17.36% of total frauds, followed by Central America (15.53%) and South America (8.89%). That is understandable since they have the most orders. Orders from those regions have to be checked carefully and thoroughly.

1.11. Who are the biggest fraudsters?:

Here we will display the biggest fraudsters' unique IDs, since two fraudsters can have the same first and last names, and using their names might lead to false statistics.



Jeffrey Wilcox, **Crystal Smith**, **David Donovan** and **Mary Mullins** are on the top of the blacklist, followed by many other fraudsters. Their transactions have to be watched carefully. Otherwise, if they persist with fraud attempts, they should be banned from the platform.

IV. Classification:

In this section, we will implement five classification models in order to predict the likelihood of **Fraudulent Transactions** for new orders (i.e. Whether a transaction is expected to be fraudulent or not). Consequently, e-commerce businesses can try to do the necessary to avoid cancellations, since we have discovered previously that orders which are suspected as fraudulent are automatically cancelled.

1. Preprocessing (2nd Leg):

As mentioned previously, a number of nominal attributes have been kept for the purpose of doing meaningful visualizations. However, some of them are irrelevant to our predictions, others are duplicated and some others need to be encoded into ordinal variables.

Based on the correlation matrix (Kindly refer to the *Correlations1.xlsx* file), a lot of features have a correlation of **0.99**, which means that we have the same information with different metadata.

The initial supply chain data set will be transformed as follows:

1. *Benefit Per Order* and *Order Profit Per Order* are the same. Thus, one of them will be dropped. The same procedure applies to:
    - *Sales Per Customer*, *Sales* and *Order Item Total*

- *Category ID*, *Product Category ID*, *Order Customer ID*, *Order Item Category ID* and *Product card ID*
- *Order Item Product Price* and *Product Price*

2. Category Names (Nominal) will be removed and Category IDs (Numerical) will be kept.
3. Transaction Types, Delivery Statuses, Customer Segments, Markets, Order Cities & Countries and Shipping Modes will be encoded into ordinal variables (e.g. 1 for Standard Class, 2 for First Class, 3 for Second Class and 4 for Same Day orders).
4. Dates, which are stored as Strings, will be parsed as POSIXct objects with the "yyyy-mm-dd UTC" format and then divided as four features for years, months, days and hours.
5. The target feature *(Fraud)* will be added based on Order Statuses (1 for the Suspected Fraud status and 0 for other statuses)
6. The *Product Status* feature, which is single-valued, will not affect our models in any way. Thus, it will be dropped too.

2. Subset Selection:

If we look at the correlation matrix of our data set (Kindly refer to the *Correlations2.xlsx* file), we can see that the strongest correlation between the *Fraud* target and other features equals 0.39. This leaves us unsure about which features can be considered as good predictors to build our models. To break this uncertainty, we will use the **Best Subset Selection** approach to trim our data to the features which are deemed important.

Considering that the optimal number of predictors is unknown, let's have a look at all possible N-Variable models (N ranges between 1 and the number of features of our data set, which is 28 in this case).

```
## Subset selection object
## Call: regsubsets.formula(fraud ~ ., new_data, nvmax = ncol(new_data) -
##     1)
## 28 Variables  (and intercept)
##                                Forced in Forced out
## Type                               FALSE      FALSE
## `Days.for.shipment.(scheduled)`    FALSE      FALSE
## Benefit.per.order                  FALSE      FALSE
## Sales.per.customer                 FALSE      FALSE
## Delivery.Status                    FALSE      FALSE
## Category.Id                        FALSE      FALSE
## Customer.Id                        FALSE      FALSE
## Customer.Segment                   FALSE      FALSE
## Department.Id                      FALSE      FALSE
## Latitude                           FALSE      FALSE
## Longitude                          FALSE      FALSE
## Market                             FALSE      FALSE
## Order.City                         FALSE      FALSE
## Order.Country                      FALSE      FALSE
## Order.Item.Discount                FALSE      FALSE
## Order.Item.Discount.Rate           FALSE      FALSE
## Order.Item.Profit.Ratio            FALSE      FALSE
## Order.Item.Quantity                FALSE      FALSE
## Product.Price                      FALSE      FALSE
## Shipping.Mode                      FALSE      FALSE
## order_days                         FALSE      FALSE
## order_months                       FALSE      FALSE
## order_years                        FALSE      FALSE
```

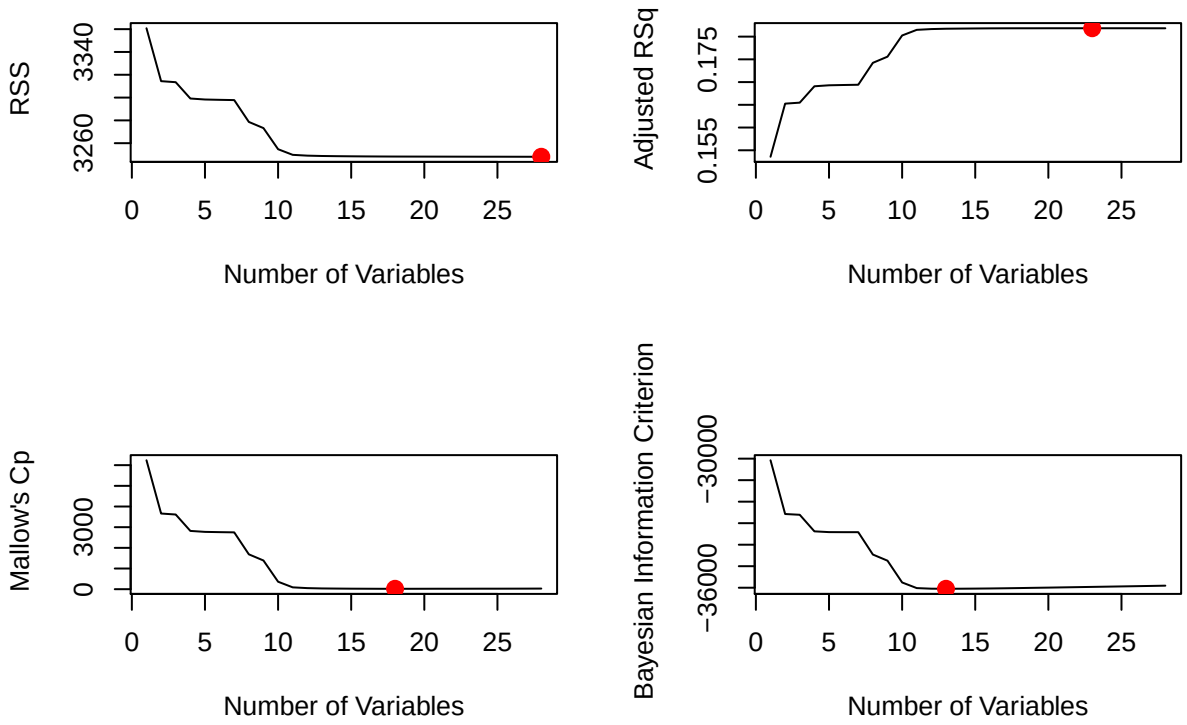
```
## order_hours                          FALSE      FALSE
## shipping_days                        FALSE      FALSE
## shipping_months                      FALSE      FALSE
## shipping_years                       FALSE      FALSE
## shipping_hours                       FALSE      FALSE
## 1 subsets of each size up to 28
## Selection Algorithm: exhaustive
##           Type `Days.for.shipment.(scheduled)` Benefit.per.order
## 1  ( 1 )  " "  " "                              " "
## 2  ( 1 )  " "  " "                              " "
## 3  ( 1 )  "*"  " "                              " "
## 4  ( 1 )  " "  " "                              " "
## 5  ( 1 )  "*"  " "                              " "
## 6  ( 1 )  "*"  " "                              " "
## 7  ( 1 )  "*"  " "                              " "
## 8  ( 1 )  " "  "*"                              " "
## 9  ( 1 )  " "  "*"                              " "
## 10  ( 1 ) " "  "*"                              " "
## 11  ( 1 ) " "  "*"                              " "
## 12  ( 1 ) "*"  "*"                              " "
## 13  ( 1 ) "*"  "*"                              " "
## 14  ( 1 ) "*"  "*"                              " "
## 15  ( 1 ) "*"  "*"                              " "
## 16  ( 1 ) "*"  "*"                              " "
## 17  ( 1 ) "*"  "*"                              " "
## 18  ( 1 ) "*"  "*"                              " "
## 19  ( 1 ) "*"  "*"                              " "
## 20  ( 1 ) "*"  "*"                              " "
## 21  ( 1 ) "*"  "*"                              " "
## 22  ( 1 ) "*"  "*"                              " "
## 23  ( 1 ) "*"  "*"                              " "
## 24  ( 1 ) "*"  "*"                              " "
## 25  ( 1 ) "*"  "*"                              " "
## 26  ( 1 ) "*"  "*"                              " "
## 27  ( 1 ) "*"  "*"                              "*"
## 28  ( 1 ) "*"  "*"                              "*"
##           Sales.per.customer Delivery.Status Category.Id Customer.Id
## 1  ( 1 )  " "                "*"             " "         " "
## 2  ( 1 )  " "                "*"             " "         " "
## 3  ( 1 )  " "                "*"             " "         " "
## 4  ( 1 )  " "                "*"             " "         " "
## 5  ( 1 )  " "                "*"             " "         " "
## 6  ( 1 )  " "                "*"             " "         "*"
## 7  ( 1 )  " "                "*"             " "         "*"
## 8  ( 1 )  " "                "*"             " "         " "
## 9  ( 1 )  " "                "*"             " "         " "
## 10  ( 1 ) " "                "*"             " "         " "
## 11  ( 1 ) " "                "*"             " "         " "
## 12  ( 1 ) " "                "*"             " "         " "
## 13  ( 1 ) " "                "*"             " "         "*"
## 14  ( 1 ) " "                "*"             " "         "*"
## 15  ( 1 ) " "                "*"             " "         "*"
## 16  ( 1 ) " "                "*"             " "         "*"
## 17  ( 1 ) " "                "*"             " "         "*"
```

```
## 18  ( 1 ) " "                "*"             "*"          "*"
## 19  ( 1 ) " "                "*"             "*"          "*"
## 20  ( 1 ) " "                "*"             "*"          "*"
## 21  ( 1 ) " "                "*"             "*"          "*"
## 22  ( 1 ) "*"                "*"             "*"          "*"
## 23  ( 1 ) "*"                "*"             "*"          "*"
## 24  ( 1 ) "*"                "*"             "*"          "*"
## 25  ( 1 ) "*"                "*"             "*"          "*"
## 26  ( 1 ) "*"                "*"             "*"          "*"
## 27  ( 1 ) "*"                "*"             "*"          "*"
## 28  ( 1 ) "*"                "*"             "*"          "*"
##           Customer.Segment Department.Id Latitude Longitude Market Order.City
## 1  ( 1 )  " "              " "           " "      " "       " "    " "
## 2  ( 1 )  " "              " "           " "      " "       " "    " "
## 3  ( 1 )  " "              " "           " "      " "       " "    " "
## 4  ( 1 )  " "              " "           " "      " "       " "    " "
## 5  ( 1 )  " "              " "           " "      " "       " "    " "
## 6  ( 1 )  " "              " "           " "      " "       " "    " "
## 7  ( 1 )  " "              " "           " "      " "       " "    " "
## 8  ( 1 )  " "              " "           " "      " "       " "    " "
## 9  ( 1 )  " "              " "           " "      " "       " "    " "
## 10  ( 1 ) " "              " "           " "      " "       " "    " "
## 11  ( 1 ) " "              " "           " "      " "       " "    " "
## 12  ( 1 ) " "              " "           " "      " "       " "    " "
## 13  ( 1 ) " "              " "           " "      " "       " "    " "
## 14  ( 1 ) " "              " "           " "      " "       " "    "*"
## 15  ( 1 ) " "              " "           " "      "*"       " "    "*"
## 16  ( 1 ) "*"              " "           " "      "*"       " "    "*"
## 17  ( 1 ) "*"              " "           " "      "*"       "*"    "*"
## 18  ( 1 ) "*"              " "           " "      "*"       "*"    "*"
## 19  ( 1 ) "*"              " "           " "      "*"       "*"    "*"
## 20  ( 1 ) "*"              " "           " "      "*"       "*"    "*"
## 21  ( 1 ) "*"              " "           " "      "*"       "*"    "*"
## 22  ( 1 ) "*"              " "           " "      "*"       "*"    "*"
## 23  ( 1 ) "*"              " "           " "      "*"       "*"    "*"
## 24  ( 1 ) "*"              " "           "*"      "*"       "*"    "*"
## 25  ( 1 ) "*"              " "           " "      "*"       "*"    "*"
## 26  ( 1 ) "*"              " "           "*"      "*"       "*"    "*"
## 27  ( 1 ) "*"              " "           "*"      "*"       "*"    "*"
## 28  ( 1 ) "*"              "*"           "*"      "*"       "*"    "*"
##           Order.Country Order.Item.Discount Order.Item.Discount.Rate
## 1  ( 1 )  " "           " "                 " "
## 2  ( 1 )  " "           " "                 " "
## 3  ( 1 )  " "           " "                 " "
## 4  ( 1 )  " "           " "                 " "
## 5  ( 1 )  " "           " "                 " "
## 6  ( 1 )  " "           " "                 " "
## 7  ( 1 )  " "           " "                 " "
## 8  ( 1 )  " "           " "                 " "
## 9  ( 1 )  " "           " "                 " "
## 10  ( 1 ) " "           " "                 " "
## 11  ( 1 ) " "           " "                 " "
## 12  ( 1 ) " "           " "                 " "
## 13  ( 1 ) " "           " "                 " "
```

```
## 14  ( 1 ) " "           " "                 " "
## 15  ( 1 ) " "           " "                 " "
## 16  ( 1 ) " "           " "                 " "
## 17  ( 1 ) " "           " "                 " "
## 18  ( 1 ) " "           " "                 " "
## 19  ( 1 ) " "           " "                 " "
## 20  ( 1 ) " "           " "                 "*"
## 21  ( 1 ) "*"           " "                 "*"
## 22  ( 1 ) " "           " "                 " "
## 23  ( 1 ) "*"           " "                 " "
## 24  ( 1 ) "*"           " "                 " "
## 25  ( 1 ) "*"           "*"                 "*"
## 26  ( 1 ) "*"           "*"                 "*"
## 27  ( 1 ) "*"           "*"                 "*"
## 28  ( 1 ) "*"           "*"                 "*"
##           Order.Item.Profit.Ratio Order.Item.Quantity Product.Price
## 1  ( 1 )  " "                     " "                 " "
## 2  ( 1 )  " "                     " "                 " "
## 3  ( 1 )  " "                     " "                 " "
## 4  ( 1 )  " "                     " "                 " "
## 5  ( 1 )  " "                     " "                 " "
## 6  ( 1 )  " "                     " "                 " "
## 7  ( 1 )  " "                     " "                 " "
## 8  ( 1 )  " "                     " "                 " "
## 9  ( 1 )  " "                     " "                 " "
## 10  ( 1 ) " "                     " "                 " "
## 11  ( 1 ) " "                     " "                 " "
## 12  ( 1 ) " "                     " "                 " "
## 13  ( 1 ) " "                     " "                 " "
## 14  ( 1 ) " "                     " "                 " "
## 15  ( 1 ) " "                     " "                 " "
## 16  ( 1 ) " "                     " "                 " "
## 17  ( 1 ) " "                     " "                 " "
## 18  ( 1 ) " "                     " "                 " "
## 19  ( 1 ) "*"                     " "                 " "
## 20  ( 1 ) "*"                     " "                 " "
## 21  ( 1 ) "*"                     " "                 " "
## 22  ( 1 ) "*"                     "*"                 "*"
## 23  ( 1 ) "*"                     "*"                 "*"
## 24  ( 1 ) "*"                     "*"                 "*"
## 25  ( 1 ) "*"                     "*"                 "*"
## 26  ( 1 ) "*"                     "*"                 "*"
## 27  ( 1 ) "*"                     "*"                 "*"
## 28  ( 1 ) "*"                     "*"                 "*"
##           Shipping.Mode order_days order_months order_years order_hours
## 1  ( 1 )  " "           " "        " "          " "         " "
## 2  ( 1 )  "*"           " "        " "          " "         " "
## 3  ( 1 )  "*"           " "        " "          " "         " "
## 4  ( 1 )  "*"           " "        " "          " "         "*"
## 5  ( 1 )  "*"           " "        " "          " "         "*"
## 6  ( 1 )  "*"           " "        " "          " "         "*"
## 7  ( 1 )  "*"           "*"        " "          " "         "*"
## 8  ( 1 )  " "           "*"        "*"          "*"         " "
## 9  ( 1 )  "*"           "*"        "*"          "*"         " "
```

```
## 10  ( 1 ) " "           "*"         "*"           "*"          "*"
## 11  ( 1 ) "*"           "*"         "*"           "*"          "*"
## 12  ( 1 ) "*"           "*"         "*"           "*"          "*"
## 13  ( 1 ) "*"           "*"         "*"           "*"          "*"
## 14  ( 1 ) "*"           "*"         "*"           "*"          "*"
## 15  ( 1 ) "*"           "*"         "*"           "*"          "*"
## 16  ( 1 ) "*"           "*"         "*"           "*"          "*"
## 17  ( 1 ) "*"           "*"         "*"           "*"          "*"
## 18  ( 1 ) "*"           "*"         "*"           "*"          "*"
## 19  ( 1 ) "*"           "*"         "*"           "*"          "*"
## 20  ( 1 ) "*"           "*"         "*"           "*"          "*"
## 21  ( 1 ) "*"           "*"         "*"           "*"          "*"
## 22  ( 1 ) "*"           "*"         "*"           "*"          "*"
## 23  ( 1 ) "*"           "*"         "*"           "*"          "*"
## 24  ( 1 ) "*"           "*"         "*"           "*"          "*"
## 25  ( 1 ) "*"           "*"         "*"           "*"          "*"
## 26  ( 1 ) "*"           "*"         "*"           "*"          "*"
## 27  ( 1 ) "*"           "*"         "*"           "*"          "*"
## 28  ( 1 ) "*"           "*"         "*"           "*"          "*"
##           shipping_days shipping_months shipping_years shipping_hours
## 1  ( 1 )  " "           " "             " "            " "
## 2  ( 1 )  " "           " "             " "            " "
## 3  ( 1 )  " "           " "             " "            " "
## 4  ( 1 )  " "           " "             " "            "*"
## 5  ( 1 )  " "           " "             " "            "*"
## 6  ( 1 )  " "           " "             " "            "*"
## 7  ( 1 )  " "           " "             " "            "*"
## 8  ( 1 )  "*"           "*"             "*"            " "
## 9  ( 1 )  "*"           "*"             "*"            " "
## 10  ( 1 ) "*"           "*"             "*"            "*"
## 11  ( 1 ) "*"           "*"             "*"            "*"
## 12  ( 1 ) "*"           "*"             "*"            "*"
## 13  ( 1 ) "*"           "*"             "*"            "*"
## 14  ( 1 ) "*"           "*"             "*"            "*"
## 15  ( 1 ) "*"           "*"             "*"            "*"
## 16  ( 1 ) "*"           "*"             "*"            "*"
## 17  ( 1 ) "*"           "*"             "*"            "*"
## 18  ( 1 ) "*"           "*"             "*"            "*"
## 19  ( 1 ) "*"           "*"             "*"            "*"
## 20  ( 1 ) "*"           "*"             "*"            "*"
## 21  ( 1 ) "*"           "*"             "*"            "*"
## 22  ( 1 ) "*"           "*"             "*"            "*"
## 23  ( 1 ) "*"           "*"             "*"            "*"
## 24  ( 1 ) "*"           "*"             "*"            "*"
## 25  ( 1 ) "*"           "*"             "*"            "*"
## 26  ( 1 ) "*"           "*"             "*"            "*"
## 27  ( 1 ) "*"           "*"             "*"            "*"
## 28  ( 1 ) "*"           "*"             "*"            "*"
```

In order to choose the optimal number of predictors, let's plot the Residual Sum of Squares, Adjusted R^2, Mallow's Cp and Bayesian Information Criterion metrics for each number of variables:



Our aim is to minimize the RSS, Cp and BIC and to maximize the adjusted R^2. If we look at the RSS metric alone, we would select all 28 features, since they give the minimal RSS score (14 variables would give the same results according to Occam's Razor principle though). However, if we consider the Cp, BIC and adjusted R^2 in addition to RSS, we would opt for a 13-variable model, which minimizes the RSS, Cp and BIC and maximizes the adjusted R^2 at the same time. Hence, we will use 13-variable models in the next sections.

Regsubsets suggests the following 13 predictors, which we will use to train our models later on: *Type, Scheduled Days for Shipment, Delivery Status, Customer ID, Shipping Mode, Order Days, Oder Months, Order Years, Order Hours, Shipping Days, Shipping Months, Shipping Years, Shipping Hours.*

3. Data Splitting:

As an initial step, we will divide our initial data set into two subsets:

- Training Set, which consists of 70% samples of the original data set, selected randomly.
- Test Set, which represents the remaining 30% of the initial data set.

The training set is a key element in the learning phase. The more diverse and balanced it is, the better the accuracy, sensitivity and specificity of the trained models are.

Let's have a look at the proportions of our target variable in the training set:

0	1
0.9774	0.02262

The training set is clearly unbalanced. 97.74% of the training samples represent non fraudulent transactions while only 2.26% represent fraudulent transactions. This is a common problem in fraud detection, as fraud is the rarest event.

If we keep the training set as it is, it will be biased towards recognizing genuine transactions (great Specificity but very bad Sensitivity). And eventhough the accuracy could be high, the trained model would still perform poorly.

Thus, we will use the SMOTE function to generate synthetic fraudulent samples and ideally get as many fraudulent samples as non-fraudulent ones.

New SMOTEd training set's proportions:

0	1
0.5	0.5

As we can see, the training set is now perfectly balanced and ready to be used to train our models.

4. K-Nearest Neighbors:

As a first algorithm, we will implement the KNN which is a simple and non-parametric algorithm. It does not depend on features and only calculates Euclidean distances between data points. However, since every feature is considered as a coordinate and since K distances are calculated for each data point, the KNN algorithm is computationally expensive. This will be demonstrated later in the learning phase.

We will test our KNN model using 10 odd values of K to avoid any tie on the majority class among the K neighbors, since we have a binary problem. The performance metric which will be used is the accuracy, based on which the best K value will be selected.

k	Accuracy	Kappa
5	0.6198	0.2395
7	0.6052	0.2105
9	0.598	0.1959
11	0.5893	0.1786
13	0.5825	0.165
15	0.5854	0.1707
17	0.5871	0.1742
19	0.5858	0.1716
21	0.5794	0.1588
23	0.5733	0.1466

As we can see, the best Cross Validation accuracy was achieved using 5 neighbors. The accuracy seems to be decreasing the more K increases. Let's see if a K value of 3 or 1 can achieve a higher accuracy.

k	Accuracy	Kappa
1	0.6694	0.3383
3	0.6105	0.2203

One neighbor seems to be enough in this case, since it gives the highest accuracy among other K values. This means that there are no significant groups within the data (the whole data can be seen as one group).

Now let's see how our **1NN** model will perform on the test set:

Confusion Matrix:

	0	1
0	32527	118
1	20425	1086

Model Results:

	x
Sensitivity	0.9019934
Specificity	0.6142733
Pos Pred Value	0.0504858
Neg Pred Value	0.9963854
Precision	0.0504858
Recall	0.9019934
F1	0.0956196
Prevalence	0.0222321
Detection Rate	0.0200532
Detection Prevalence	0.3972044
Balanced Accuracy	0.7581333

The accuracy is quite low (62.07%). The sensitivity is not very bad but the model is obviously not specific (It tends to misclassify more than half of non-fraudulent transactions). The KNN algorithm seems to be a bad choice for this problem.

5. Logistic Regression:

Now, let's try to do a logistic regression:

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	34.1483251	2305.0127012	0.0148148	0.9881799
Type	-0.0784143	438.7999575	-0.0001787	0.9998574
'Days.for.shipment.(scheduled)'	0.2431534	0.1813550	1.3407596	0.1799985
Delivery.Status	24.6061994	513.5429659	0.0479146	0.9617843
Customer.Id	0.0000104	0.0000274	0.3787605	0.7048657
Shipping.Mode	0.0503837	0.2363469	0.2131768	0.8311891
order_days	0.1941382	0.0920882	2.1081772	0.0350157
order_months	5.4324880	2.7723102	1.9595527	0.0500481
order_years	66.5799780	33.2873911	2.0001561	0.0454834
order_hours	0.1136239	0.0417121	2.7240057	0.0064495
shipping_days	-0.1817780	0.0911465	-1.9943496	0.0461139
shipping_months	-5.5627628	2.7750572	-2.0045579	0.0450103
shipping_years	-66.6434130	33.2903962	-2.0018810	0.0452975
shipping_hours	-0.1178016	0.0431026	-2.7330496	0.0062751

Confusion Matrix:

	0	1
0	51889	0
1	1063	1204

The logistic regression model gave a 98.04% accuracy in 22 iterations, which is excellent. It also achieved an good AUC score 0.7655492, which means that the trade-off between its precision and recall is well managed. The model is also very sensitive (100%) and specific (97.99%). There is only a few number of false positives.

The logistic regression model also suggests that the strongest predictors are order and shipping dates and times. Let's verify if this is true:

Confusion Matrix:

	0	1
0	29189	684
1	23763	520

Those assumptions are definitely not correct as the accuracy dropped significantly (From 98.04% to 54.86%). This means that a model's performance cannot always be judged based on low P-Values of its predictors.

6. Support Vector Machines:

Now let's build an SVM model using a Linear Kernel with 10 default levels of tuning parameters and select the best parameter values using a 10-fold cross validation:

C	Accuracy	Kappa	AccuracySD	KappaSD
1	0.9906	0.9811	0.002011	0.004023

The model achieved a perfect cross validation accuracy using the first fold only. Let's see how it performs on our test set:

	0	1
0	51889	0
1	1063	1204

It still performs very well on a different set with a few false positives again and 100% true positives. We can conclude that our data set is linearly separable.

7. Random Forest:

Now we will build a random forest model with 3 default levels of tuning parameters and select the best parameter values using a 10-fold cross validation:

mtry	Accuracy	Kappa	AccuracySD	KappaSD
2	0.991	0.982	0.002798	0.005597
7	0.9915	0.983	0.002888	0.005776
13	0.9913	0.9825	0.003086	0.006173

The best cross validation accuracy was achieved using 7 random variables at each split.

Again, let's see how the tuned model performs on our test set:

	0	1
0	51961	4
1	991	1200

The model performs very well with an accuracy of 98% and a neglectable number of false positives and false negatives. Nevertheless, 4 false negatives in fraud detection is too much and can cost the company hundreds of thousands of dollars.

8. Neural Networks:

The last and the main algorithm is the Neural Networks algorithm. Since we have seen that our data is linearly separable, we will use one hidden-layer only and let the algorithm fine-tune hyper-parameters using a random search. The only stopping criterion that will be used here is the maximum number of iterations, which is intially fixed at 50.

size	decay	Accuracy	Kappa	AccuracySD	KappaSD
1	0	0.9906	0.9813	0.002581	0.005162
1	1e-04	0.9871	0.9741	0.01022	0.02043
1	0.1	0.9905	0.9809	0.002691	0.005382
3	0	0.9905	0.9809	0.002754	0.005508
3	1e-04	0.9906	0.9811	0.002605	0.00521
3	0.1	0.9906	0.9811	0.002733	0.005467
5	0	0.9904	0.9808	0.002546	0.005093
5	1e-04	0.9904	0.9808	0.002479	0.004957
5	0.1	0.9906	0.9811	0.002733	0.005467

The highest cross validation accuracy was achieved using one neuron. Let's see how the model performs on the test set:

	0	1
0	51889	0
1	1063	1204

Accuracy (98.04%), sensitivity (100%) and specificity (97.99%) are all excellent with the auto fine-tuned hyperparameters of the train function.

Let's try to reduce the training time by dropping the number of iterations to 30:

size	decay	Accuracy	Kappa	AccuracySD	KappaSD
1	0	0.9906	0.9813	0.002581	0.005162
1	1e-04	0.987	0.9739	0.01051	0.02102
1	0.1	0.9906	0.9813	0.002741	0.005482
3	0	0.9899	0.9799	0.003011	0.006022
3	1e-04	0.9906	0.9813	0.002741	0.005482
3	0.1	0.9906	0.9811	0.002733	0.005467
5	0	0.9906	0.9811	0.002572	0.005144
5	1e-04	0.9904	0.9808	0.002644	0.005289
5	0.1	0.9904	0.9808	0.00274	0.005479

	0	1
0	51888	0
1	1064	1204

The model's performance and architecture are the same with 30 iterations.

Will 20 iterations break the rule?:

size	decay	Accuracy	Kappa	AccuracySD	KappaSD
1	0	0.99	0.9801	0.003256	0.006512

size	decay	Accuracy	Kappa	AccuracySD	KappaSD
1	1e-04	0.982	0.964	0.01536	0.03072
1	0.1	0.9855	0.9709	0.01627	0.03253
3	0	0.9881	0.9762	0.005053	0.01011
3	1e-04	0.9903	0.9806	0.003262	0.006524
3	0.1	0.9884	0.9767	0.005118	0.01024
5	0	0.9904	0.9808	0.002676	0.005351
5	1e-04	0.9905	0.9809	0.002659	0.005318
5	0.1	0.9904	0.9808	0.002771	0.005541

	0	1
0	51888	2
1	1064	1202

In 20 iterations, the model misclassifies two fraudulent transactions with a slightly reduced accuracy of 98.03% compared to the two previous models. In addition, it uses 5 neurons in the hidden layer, which means more calculations and subsequently, a higher training time.

Now let's try to set the initial hyperparameter values manually and see which method is the most effective. We will use two repetitions with a repeated 10-Fold Cross Validation, a decay of 3.16e-3, 50000 maximum weights and 100 maximum iterations:

	0	1
0	51926	17
1	1026	1187

The accuracy improves slightly (98.07)%, but the 17 false negatives on their own are enough to say that the model performs poorly.

We can conclude that initial hyperparameter values are hard to set. They put us in a long "randomly set-and-test" loop, which does not always lead to satisfying results.

The 30-iteration and 1 neuron model wins the battle in this case. Hence, it will be used for comparison with the aforementioned algorithms.

7. Conclusion:

Throughout this project, we have used the **Caret** package which lets us train different kinds of models using different sampling methods.

The approach we have adopted consists of defining a fixed tuning length and letting the **Train** function do a random search by using default initial hyper-parameter values and fine-tuning those values using a 10-fold cross validation. The best set of values correspond to the highest accuracy.

The following table shows the accuracies, sensitivities and specificities of the five models built:

Model	Accuracy	Sensitivity	Specificity	Training Time
KNN	62.07	90.2	61.43	17.86 secs
Logistic Regression	98.04	100	97.99	0.485 secs
SVM	98	100	97.99	15.78 secs
Random Forest	98	99.67	98.13	143 secs
Neural Networks	98.04	100	97.99	38.28 secs

As we can see, the highest accuracy was given by the **Logistic Regression** and the **Neural Networks** models, with the same perfect sensitivity. However, the Logistic Regression is significantly faster than the Neural Network. Thousands, if not millions, of E-Commerce transactions happen daily, thus causing a continuous data increase. The more samples we have, the model needs to be updated and its learning time becomes crucial. Since there is a simple and effective solution, there is no need to go for complex alternatives (according to Occam's Razor).

Hence, if we were to implement a fraud detection system using this data set, we would definitely use the **Logistic Regression**.

V. Future work (assumptions and potential extensions):

1. Feature engineering: This helps to extract more information from existing data. Feature engineering is highly influenced by hypotheses generation. Good hypothesis result in good features. That is why, it is always fruitful to invest quality time in hypothesis generation.
2. A domain expert can help us further extract insights from our data set.
3. Product price, product category and market region are significant factors that can aid in our fraud detection process.
4. Ensemble modeling: This technique simply combines the result of multiple weak models and produce better results. This can be achieved through many ways:
    - Bagging (Bootstrap Aggregating)
    - Boosting

VI. References:

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