Table of Contents

[1.0 Introduction 1](#_Toc212235886)

[2.0 Data description 3](#_Toc212235887)

[3.0 Methodology 6](#_Toc212235888)

[3.1 Data Preprocessing 6](#_Toc212235889)

[3.2 Model Building 15](#_Toc212235890)

[3.3 Evaluation 17](#_Toc212235891)

[4.0 Results 20](#_Toc212235892)

[5.0 Conclusion 25](#_Toc212235893)

[6.0 References 26](#_Toc212235894)

[7.0 Appendix 28](#_Toc212235895)

[7.0 Appendices: peer evaluation 29](#_Toc212235896)

# 1.0 Introduction

In July 2024, global shipping problems were approximately four times worse than their long-term average. This was the worst supplier-delivery decline documented by the PMI survey outside of the pandemic years (Williamson, 2024). Dierker et al. (2022) mentioned that average schedule slippage on the strategically important Far East to North America corridor grew from just 2 days in early 2020 to 12 days by Q4 2021. This is a six-fold increase that is still affecting routing plans in 2024.

Such instability is more than an operational nuisance. Studies have shown that more uncertainty in lead time raises inventory prices, causes stock-outs, and makes businesses spend more on emergency freight (Mohammed & Mandal, 2024). Late delivery make customers less likely to buy again, while early or on-time deliveries make them more likely to buy again. Some industries have seen sales drop by as much as 10% for three to four weeks after even "minor" delays (Matteo Gabellini et al., 2024).

Now, let's talk about our business dilemma. For our high-volume sporting goods store, the question is no longer if an order will be late, but which one it will be and how soon we can see it coming. Every unexpected delay hurts service-level agreements, costs money to switch modes, and makes people less loyal to your business (Harter et al., 2024).

At the same time, our project goal, which is based on these stakes, is to create a supervised machine-learning pipeline that marks high-risk shipments before they leave the warehouse. Specifically, the model will first Predict the probability that a new order will miss its promised date, Second priorities at‑risk shipments for proactive interventions (premium carriers, dynamic routing, customer alerts), and third explain which operational levers—port choice, lane congestion, shipment mode—drive most lateness, giving managers evidence‑based levers for process redesign (Matteo Gabellini et al., 2024).

Why do data mining here? The standards for safety buffers that have been around for a long time can't keep up with networks that are influenced by climate change and are politically unstable. Recent work in applied science reveals that deep learning models that capture macroeconomic cues do a better job of predicting delivery delay risk than standard heuristics. Our method fits perfectly with the first three criteria of the rubric: a clear business problem, SMART analytical objectives, and proven business significance. This sets the stage for an evidence-based analysis that saves money on operations and improves the customer experience.

Why do data mining? Matteo Gabellini et al. (2024) mentioned that the norms for safety buffers that have always worked can't keep up with networks that are influenced by climate change and are politically unstable. Recent research in applied science has shown that deep-learning models that capture macroeconomic cues are better at predicting delivery delays than traditional heuristics. Our method fits perfectly with the first three criteria of the rubric: a clear business problem, SMART analytical objectives, and proven business significance. This sets the stage for an evidence-driven analysis that leads to both operational savings and improvements in the customer experience.

# 2.0 Data description

***Data Sources***

|  |  |
| --- | --- |
| Item | Details |
| Name | “ DataCo SMART Supply Chain for Big Data Analysis ” |
| Source | Kaggle dataset by Shashwat Work |
| Link | <https://www.kaggle.com/datasets/shashwatwork/dataco-smart-supply-chain-for-big-data-analysis/data> |
| Time span | 2015 - 2019 transactions |
| Rows × Cols | 180 519 records × 53 variables |
| File format | .CSV |
| Licence | Public-domain |

***Attribute of Data***

|  |  |  |  |
| --- | --- | --- | --- |
| Variable | Type | Example | Business meaning |
| late\_delivery\_risk | Binary target | 1 | 1 = late delivery is expected |
| days\_for\_shipping\_real | Numerical | 5 | Transit time count by Days |
| days\_for\_shipment\_scheduled | Numerical | 3 | Service Level Agreement promise Days |
| shipping\_mode | Categories | First Class | Service level applied to Shipping |
| order\_date | DATE | 2018-05-03 | Order creation date |
| shipping\_date | DATE | 2018-05-01 | Goods left warehouse |
| customer\_city | Text | Chicago | Location of Buyer |
| customer\_country | Text | United States | Country of Buyer |
| payment\_type | Categories | Dedit | Payment method |
| order\_item\_quantity | Numerical | 2 | Line of unts |
| Product\_category  sales | Text  USD | Sportwear  197 USD | Product Categories  Revenue |

The precise data needs for this analysis are focused on a number of key customer attributes that will enable a thorough comprehension of customer behaviour, transaction history, and delivery performance. Customer demographics, which include variables like **customer\_city** and **customer\_country** that provide light on the regional and geographical elements influencing delivery risk, are crucial in the first place. Although age group data would be useful for further segmentation, it is not included in the data set and will either be estimated via geographical and economic segmentation or not included in the analysis.

To analyse customer purchase patterns and payment preferences, it is crucial to consider transaction history and key attributes such as **sales**, **order\_item\_quantity**, and **payment\_type**. The connection between purchase behaviour and delivery risk may be evaluated with the aid of these attributes. To identify any delays or discrepancies in **delivery schedules**, the **order\_date** and **shipping\_date** variables will be utilised to monitor the period between order placement and shipment.

In order to anticipate late deliveries based on past patterns, our analysis will use the delivery status, which is given by the **late\_delivery\_risk** variable, as the target label. The analysis of operational efficiency will be supported by the **days\_for\_shipping\_real** and **days\_for\_shipment\_scheduled** variables, which will provide information about actual vs promised delivery timings. Together, these data sets provide the framework for examining delivery delays and refining logistical plans.

The chosen dataset is contains detailed customer order transactions with delivery status of a retail sporting goods company from Kaggle, (<https://www.kaggle.com/datasets/shashwatwork/dataco-smart-supply-chain-for-big-data-analysis/data>). It includes customer demographics details, delivery status, payment methods, risk of delivery, etc. The data is used for performance analysis, logistic optimizations. The file format is

# 

# 3.0 Methodology

Based on the analytical questions designed in the project proposal, the project’s primary objective is to predict late delivery risk based on product and customer data, a binary classification problem with binary outcome late (1) or on-time (0). The secondary objective is to analyze the late delivery factors and regions. Based on the initial exploratory data analysis, the observations revealed complex and non-linear relationships between features. Therefore, this project will employ tree-based models to predict outcomes and extract actionable insights.

## 3.1 Data Preprocessing

The dataset was loaded using pandas with Latin-1 encoding to handle special characters. This project uses a variety of libraries for data preprocessing, visualisation, modelling and evaluation. Pandas and NumPy were imported for data manipulation and numerical operations in preprocessing. Matplotlib and Seaborn libraries were used for data visualisation. The modules in Sciki-Learn library are used for data splitting, encoding, scaling, feature selection, a portion of modelling, and evaluation. Lastly, XGBoost and LightGBM libraries are imported for modelling.

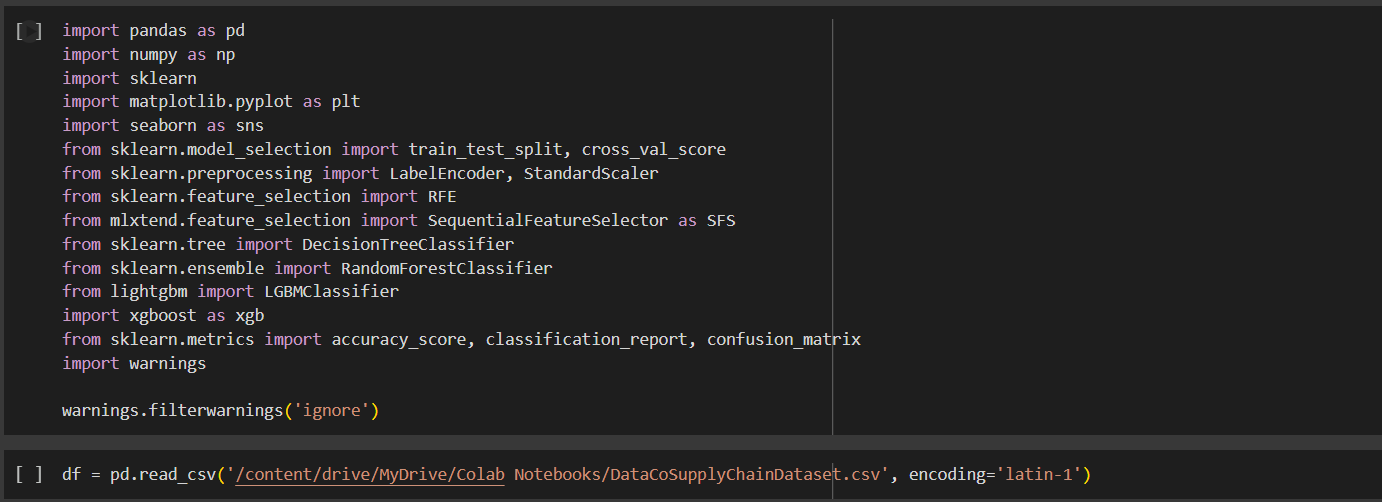


Figure 1: Importing Libraries and Dataset

The data preprocessing begins with data cleaning. To identify missing data and duplicates, Figure 2 screens for rows with null values and duplicates in each column. The result in Figure 3 reveals that “Product Description” and “Order Zipcode” contain a significant portion of missing data, and were dropped subsequently, but no duplicates were found.

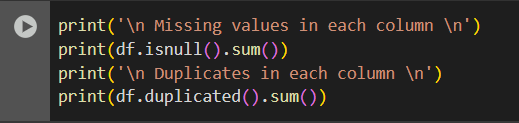


Figure 2: Identify Missing Value and Duplicates

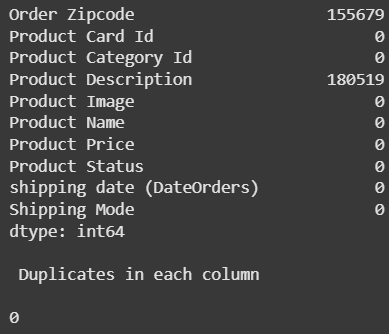


Figure 3: Missing Values and Duplicates Identified

Next, all numerical variables were plotted with boxplot and distribution plot to visualise the outliers and data distribution. Observations revealed that variables like “Days for Shipment (scheduled)”, “Benefit per order”, “Latitude”, “Order Item Profit Ratio”, and “Order Profit Per Order” are heavily right-skewed, while variables like “Sales per customer”, “Customer Zipcode”, “Latitude”, “Order Item Discount”, “Order Item Product Price”, “Order Item Quantity”, “Sales”, Order Item Total”, and “Product Price” are skewed to the left. The right skewness in financial data is normal, as most transactions are small and a few are large.

The following columns are dropped:

1. 'Days for shipping (real)', 'Days for shipment (scheduled)' - Columns directly cause leakage in model prediction; unrealistic to have data of these 2 variables in a real-life scenario when predicting late delivery.
2. 'Delivery Status' – directly leaks to the target variable in text form
3. ’Customer Fname', 'Customer Lname', 'Product Name', 'Product Image', 'Customer Password', 'shipping date (DateOrders)', 'Customer Zipcode', 'Customer Email', 'Product Status', 'Latitude', 'Longitude', – Irrelevant columns
4. 'Customer Id', 'Category Id', 'Department Id', 'Order Customer Id', 'Order Item Cardprod Id', 'Order Item Id', 'Product Card Id', 'Product Category Id', 'Order Id' – Unique identifiers with no predictive power

After dropping unnecessary columns, 27 columns remained from the initial 53 columns

Further data cleaning is carried out to check for dirty data and messy data by checking the unique values in each column. It was discovered that the Customer City column has a misplaced value, as shown in Figure 4. Further investigation of the rows with “CA” in Customer City column shows that these 3 entries have misplaced values in Customer State column, as shown in Figure 5. After verifying the state origin using the Customer Street, which are Elk Grove and El Monte, these two values are used as reference to map the correct values to Customer State and Customer City columns, as demonstrated in Figure 6.

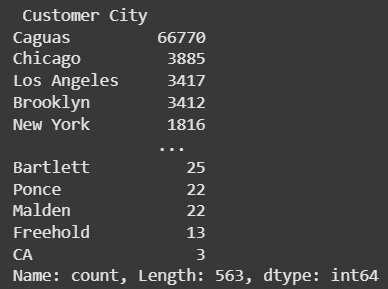


Figure 4: Misplaced Values in Customer City Column

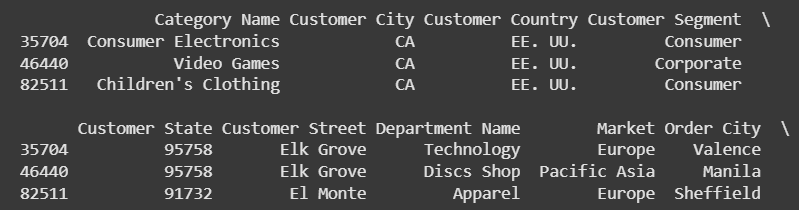


Figure 5: Misplaced Values in Customer State

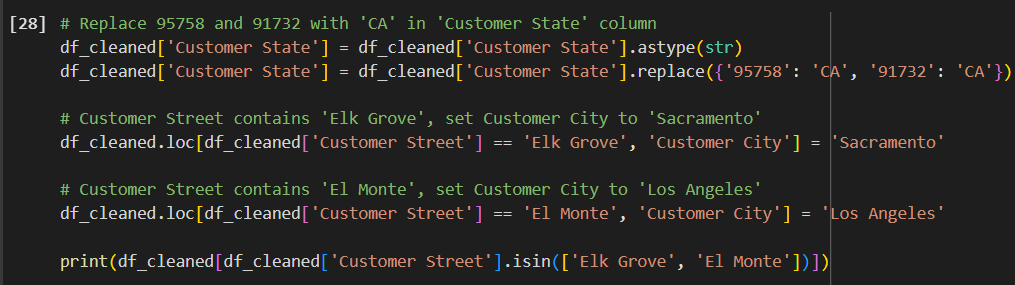


Figure 6: Correcting Misplaced Values in Customer Street and Customer City columns

Furthermore, “order date (DateOrders)” column was in the wrong data type; it was converted to datetime data type with correct format. However, raw datetime objects cannot be directly used in most machine learning models. Therefore, Datetime decomposition was performed to the Order date column to extract numerical features such as year, month, day and hours, enabling the models to discover time-based patterns.

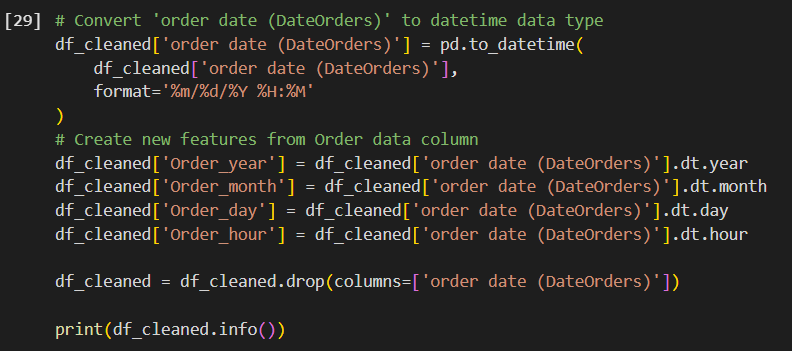


Figure 7: Datetime Decomposition

To ensure no redundant text classes in categorical columns, text cleaning is applied to categorical columns. The .strip() function removes the leading and trailing whitespace for all columns with strings. The result shows a reduction of 505 data variations in the Customer Street column.

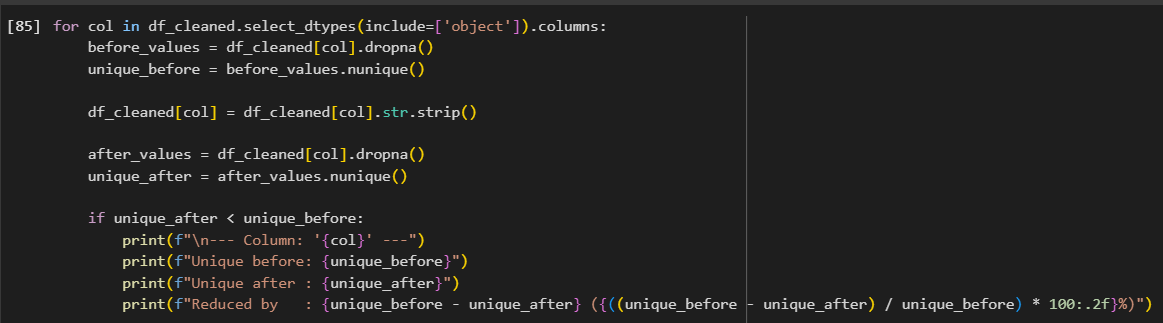


Figure 8: Text Cleaning

Due to the wide dimensionality of this dataset, visualisation and modelling require long computation time and high memory usage. Thus, memory optimization is applied to prevent runtime crashing by downcasting the numerical columns to the smallest integer or float types without information loss, ranging between 8, 16 and 32 bits; the float and integer bit are determined based on the max number of digits in each column. Mokin’s utility function is adapted which works by iterating through all numerical columns of “df\_cleaned” dataframe (Mokin, 2024) The memory reduction function reduced memory usage by 36%, the code includes calculation to measure improvement.

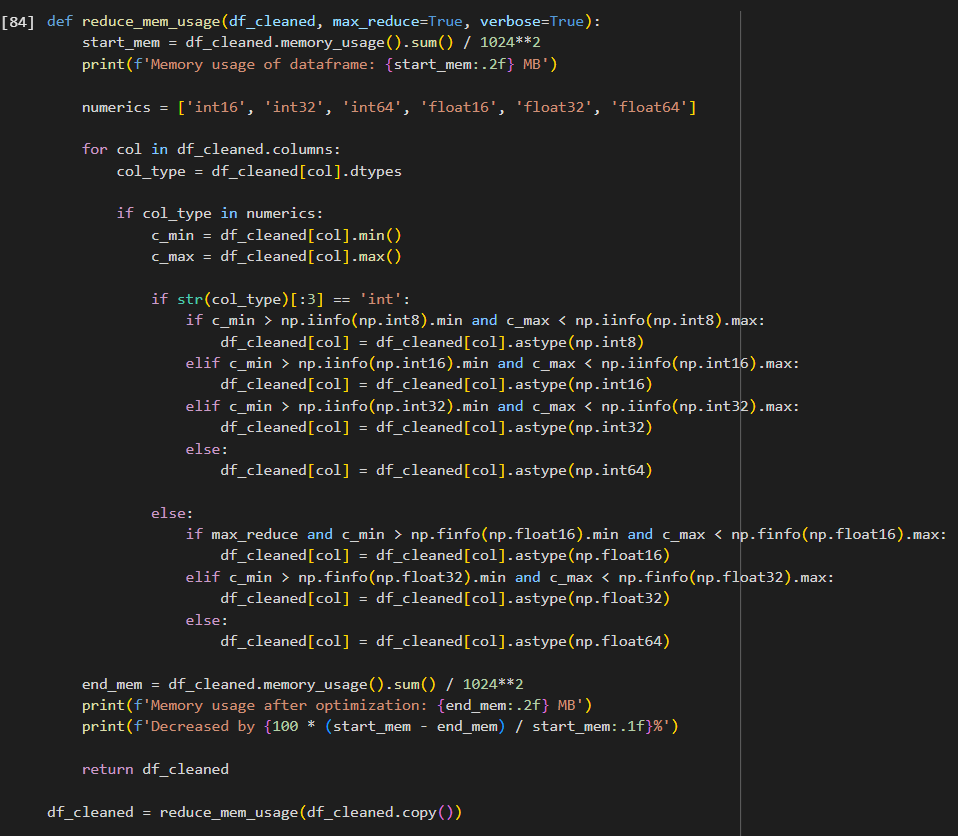


Figure 9: Memory optimisation

After data cleaning, the dataframe is split into train and test sets with a (70:30) ratio before performing scaling and encoding to avoid data leakage. The parameters include stratification based on the target variable to ensure class distribution is maintained.

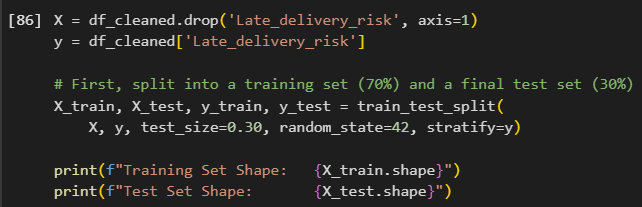


Figure 10: Data Splitting

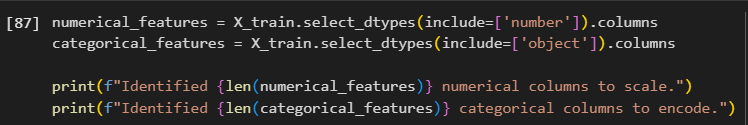


Figure 11: Categorical and Numerical Features

For Data Transformation, Label Encoding method was selected to convert categorical features into integers. This method was chosen over One-Hot Encoding mainly to avoid the high dimensionality that would result from the high cardinality in most categorical features. Most importantly, the chosen tree-based models are not influenced by the artificial orders created by Label Encoding. To preventing data leakage, the encoder was fitted on the training set exclusively, and the learned mapping was then applied to transform the test set (Scikit-learn, 2025).

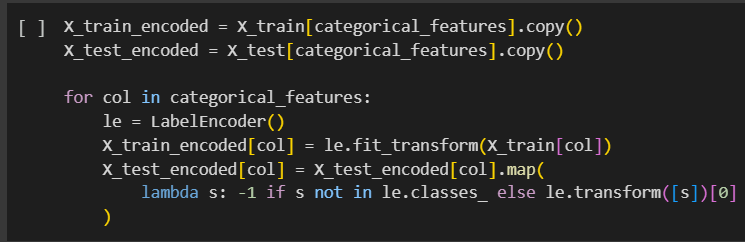


Figure 12: Data Encoding

Standardization was applied to all numerical features using Scikit-Learn’s StandardScaler module. Although outliers were identified during EDA, they were not removed because they represent valid business scenarios. Also, the chosen models are tree-based, so the models are inherently robust and less sensitive to extreme values compared to linear models.

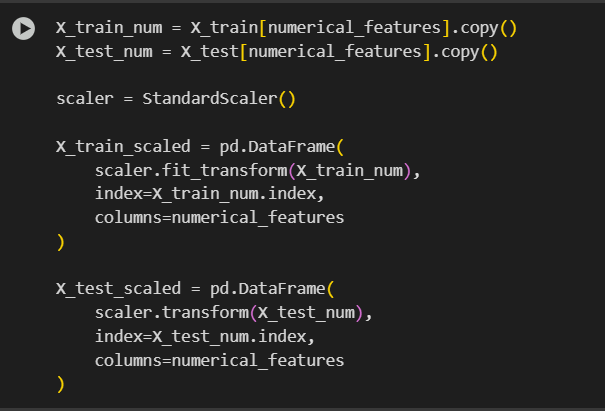


Figure 13: Data Scaling

After Encoding and Standardization, the two resulting dataset were merged back together based on their initial index. This function ensures categorical features and numerical features in each row are aligned correctly, creating the final training and test set for modelling. To further optimise for performance, the memory optimization function is applied once again to downcast the float64 data types produced during encoding and standardization.

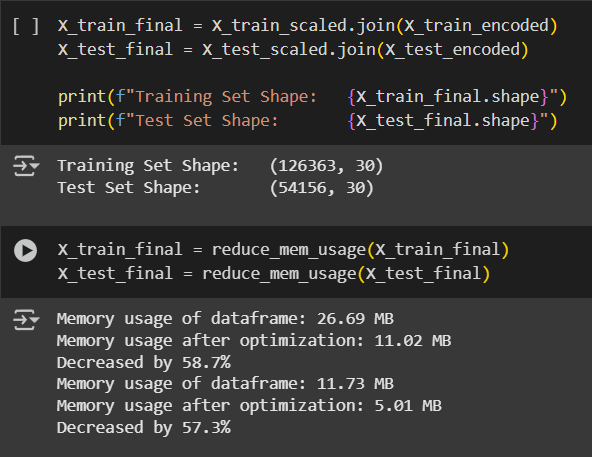


Figure 14: Data Merging

.

To ensure the models perform accurately and reduce noise, a wrapper method was used for feature selection. Due to the computational cost and time of this method on the dataset, a smaller representative sample was created from the training set using stratified sampling. Then, forward selection was performed on this sample using the Sequential Feature Selection module with a Decision Tree as the estimator. The estimator evaluates features using F1-score and achieves 0.7113 in cross-validation score. The best 15 features identified in this process were then used to train the final models.

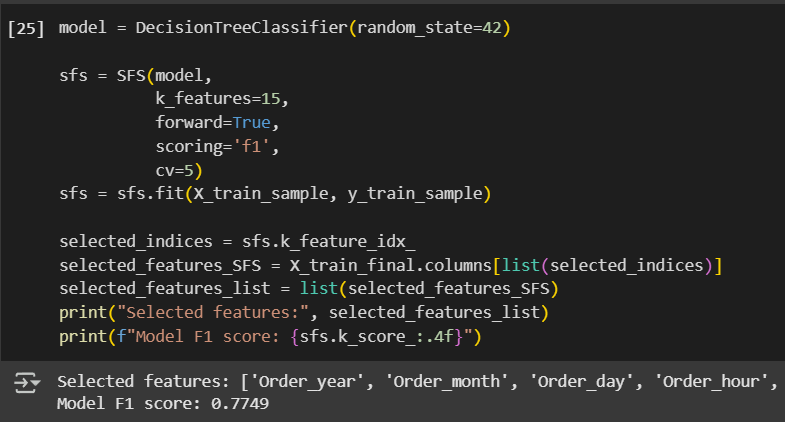


Figure 15: Feature Selection

## 

## 

## 3.2 Model Building

Given the financial and operational implication of late deliveries, this project requires a modelling approach with high predictive accuracy and actionable insights. A baseline Decision Tree will be used to provide initial performance with simple and understandable rules.To meet the primary objective of accurate prediction with complex features, ensemble models were chosen: Random Forest, LightGBM Classifier, and XGBoost Classifier. These models are most suitable for this complex problem as they can capture non-linear relationships between features in large tabular datasets where traditional methods fall short.

To ensure a fair and consistent model comparison, all models were trained and evaluated using an identical workflow. However, LightGBM Classifier’s verbosity parameter was set to -1, to suppress the logging output to maintain a clean output. Each model was fitted on the preprocessed training data (X\_test\_fs), then the trained model’s performance was evaluated by generating predictions on the unseen test (X\_test\_fs) using the .predict() method. Each model prediction is evaluated using classification metrics, including the confusion matrix, overall accuracy score and the classification report.

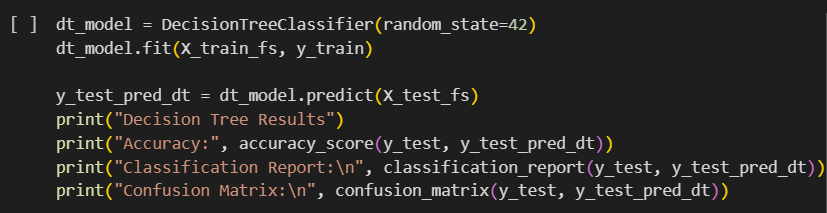


Figure 16: Decision Tree Model Building

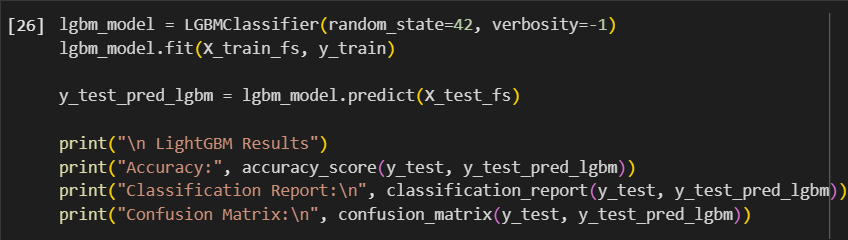


Figure 17: LightGBM Classifier Model Building

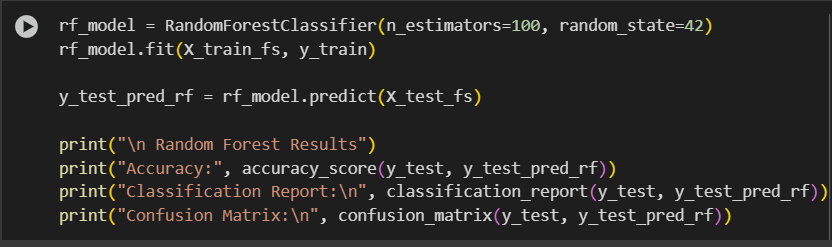


Figure 18: Random Forest Model Building

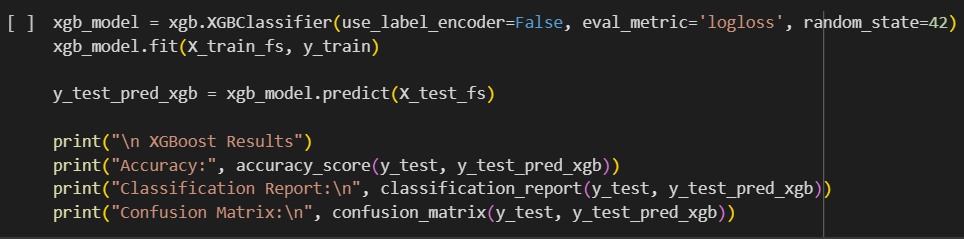


Figure 19: XGBoost Classifier Model Building

## 

## 3.3 Evaluation

Firstly, Decision Tree is a supervised learning algorithm used for classification and regression method that data is split into branches such as yes or no answer. It has the best performance among all evaluated models, achieving the highest accuracy of (94.13%). It demonstrates a balanced performance for both classes, with a precision and recall of (0.94 & 0.95) for the “late” class (class 1) and “not late” (class 0). High recall means fewer false negatives and high precision indicates the correctly predicted cases. This means it correctly identified most late and not late cases while minimizing false positives. Hence, Decision Tree is a reliable and accurate choice for delay prediction.

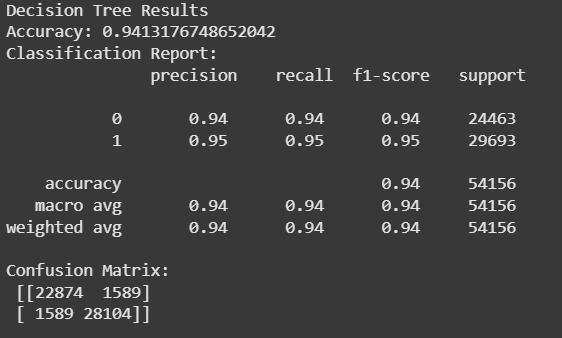


Figure 20: Decision Tree Results

Secondly, LightGBM (Light Gradient Boosting Machine) is a high-performance gradient boosting framework developed by Microsoft, designed for speed and efficiency to deal with large datasets with many features. In this case, it has the worst performance compared to other three models, with an **Accuracy - 74.29%** , slightly lower than XGBoost (74.5%). Based on the confusion matrix, it shows a high recall of (0.91), meaning most of on time deliveries were identified, but precision rate is low (0.65) meaning most of “not late” cases were misclassified as “late”. However, for class 1 (late) LightGBM achieves higher precision of (0.89) and low recall (0.60), meaning 40% of late deliveries were missed. This indicates overfitting and imbalance of data, suggesting a poor overall predictive performance.

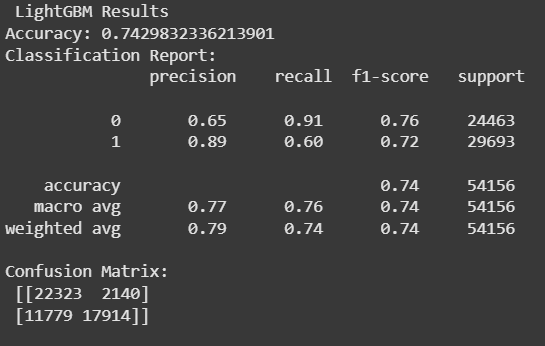


Figure 21: LightGBM

Thirdly, XGBoost (Extreme Gradient Boosting) is an algorithm used in machine learning that build model sequentially, where each new model focuses on correcting errors of the previous ones. Suitable to minimise a loss function, reduce overfitting, speed, and memory efficiency. The accuracy of (77.5%) is slighter higher than LightGBM (74.2%). However, it is still relatively low. Recall for

“late” class is (0.86), the precision is only (0.70), indicating a high number of false positives. For “not late” class precision low (0.70) with high recall (0.87), meaning it correctly identifies on time deliveries. The model favored predicting delays, which can be good in high-risk scenarios but reduces trust in predictions due to lower precision. XGBoost is generally powerful for structured data, but in this case, it did not outperform simpler models.

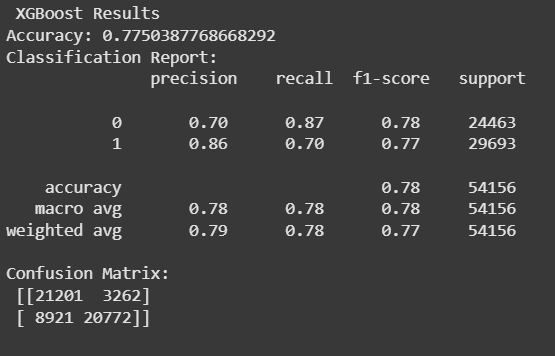


Figure 22: XGBoost Results

Lastly, random forest performs the second best and has a high accuracy of (94.9%). Random Forest is an ensemble learning method based on bagging that builds multiple decision trees and aggregates their output based on majority which help reduce overfitting and improve generalization compared to Decision Tree. It performs well in identifying class 1 “late” with a high recall of (0.97) and F1-score of (0.95), indicating most actual class 1 were correctly predicted. This model confusion results show a balance of recall and precision for both classes, this means most cases were correctly identified and predicted. Random Forest typically performs well on complex datasets due to its ensemble nature, therefore there may be overfitting issues in the dataset.

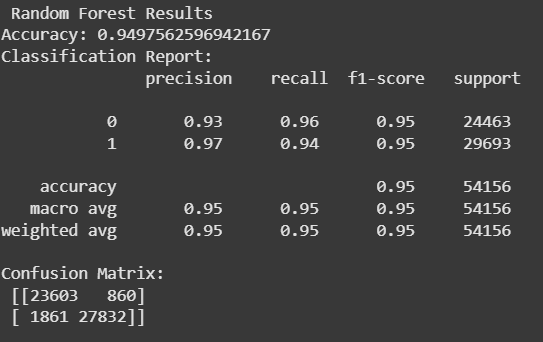


Figure 23: Random Forest Results

# 

# 4.0 Results

**Result**

Based on the objective and analytics questions set earlier for this project, a model that could predict whether order will be delayed should be developed. Eventually, this report has successfully developed models that could predict the probability of delaying the order through analysing the historical data like past order details, shipment modes, customer locations etc. There are 4 models being developed in this report, which are Decision Tree, LightGBM, Random Forest and XGBoost. **Random Forest** will be the best model for conducting prediction in this case as it has the highest accuracy rate of **94.98%** compared to other 3 models.

Besides, there are 53 features or called as variables in the datasets, in order to get more accurate prediction, the model has identified 15 features that could affect outcome the most which are Order year, Order month, Order day, Order Hour, Type, Customer Country, Customer Segment, Customer State, Customer Street, Market, Order City, Order Country, Order State, Order Status, Shipping Mode. Through learning the historical data trends from the above features, the random forest data could predict an accurate outcome by conducting analysis through the 15 features above.   
Below is the chart indicating the ranking of top 15 features that affect the outcome the most.

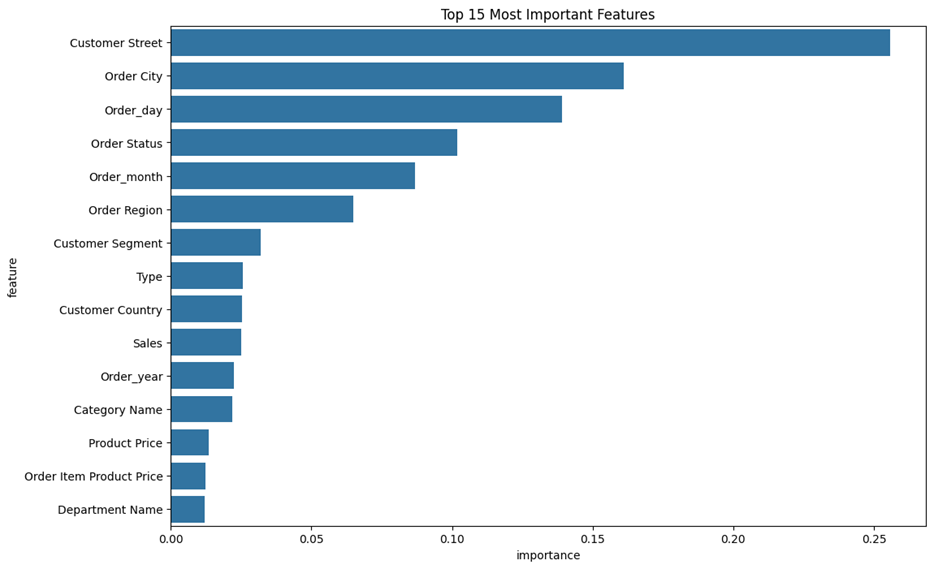


Figure 24

Below are the 7 graphical charts discovering key findings and trends from the dataset.

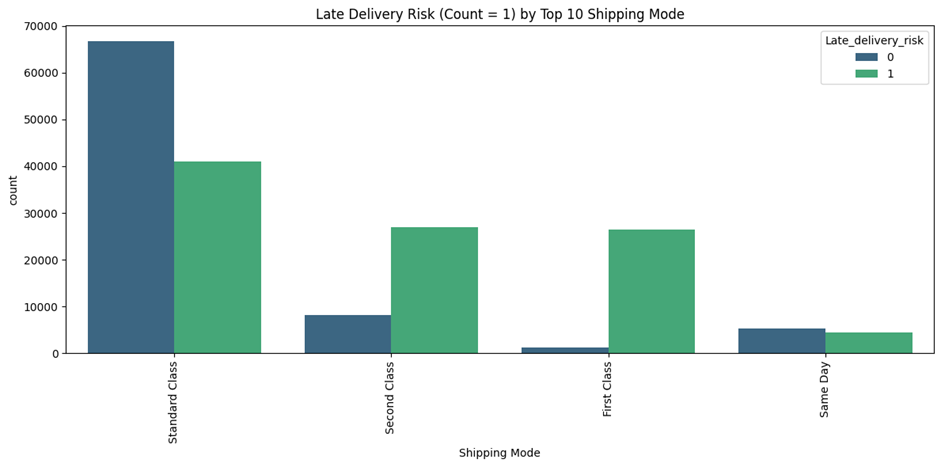


Figure 25

* The diagram above shows different classes of shipping mode could affect late deliveries
* Standard Class has the highest number of late deliveries, while second class and first have not much difference but Same Day shipping mode has the lowest late deliveries.



Figure 26

* The Diagram above shows the trend of late deliveries by months in a year.
* January is the month with the highest number of late deliveries.
* Starting from February until September has a comparatively steady trend.
* The last quarter of the year has the lowest numbers of late deliveries throughout the whole year.

A graph of a bar chart

AI-generated content may be incorrect.

Figure 27

* The Diagram above shows that Latam (Latin America) is the market with the highest amount of late delivery in the world, followed by Europe and Pacific Asia while Africa is the market with the least late delivery.

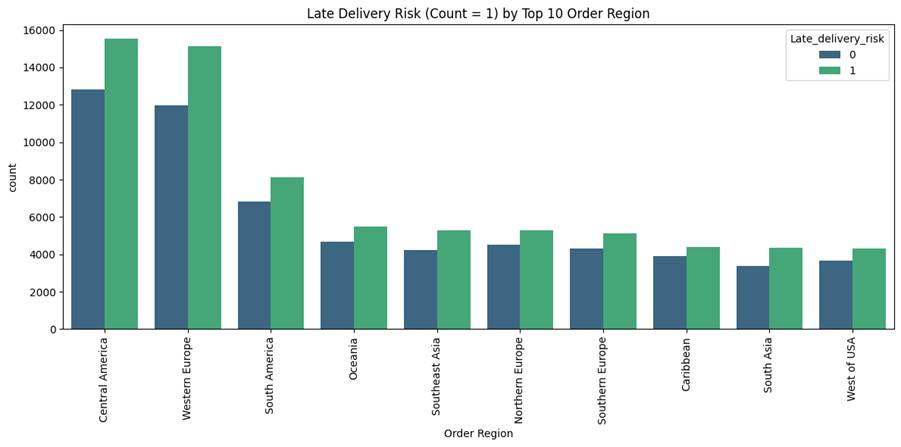


Figure 27

* The Diagram above shows that Central America have the highest number of late deliveries followed by Western Europe, South America etc, indicating the main region experiencing late delivery is around America.

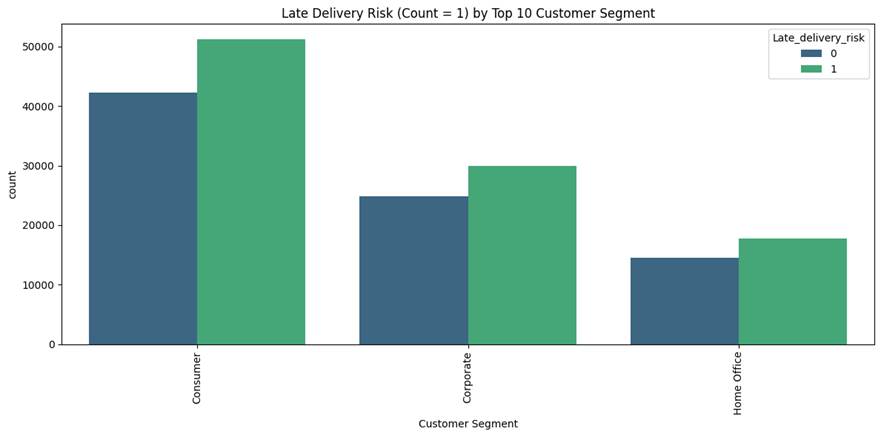


Figure 28

* The diagram above shows customers who are in consumer segment tend to experience more late delivery than customers who are in corporate and home office segment.

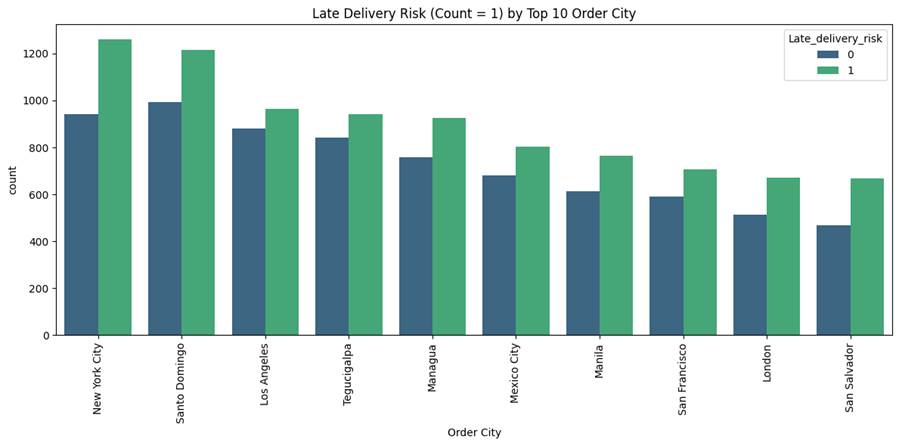


Figure 29

* The above diagram indicates the top 10 cities with the highest number of late deliveries in the world.
* New York city tends to have more late delivery followed by Santa Domingo, Los Angeles etc.

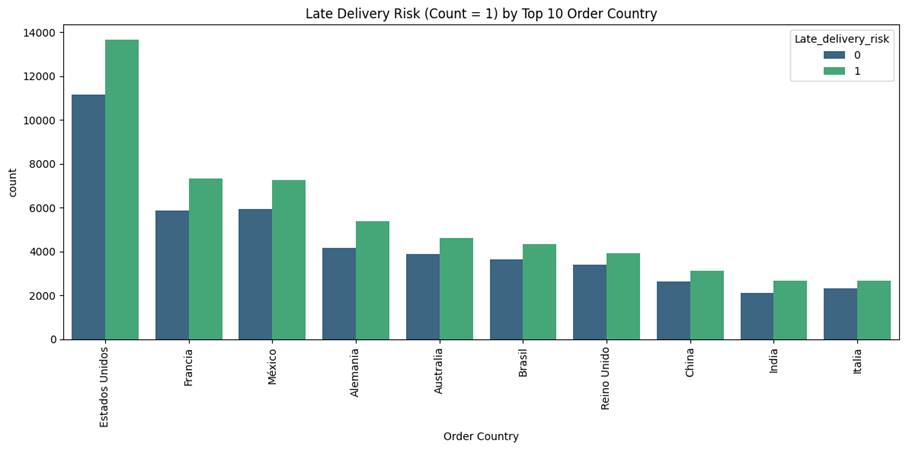


Figure 30

* The diagram above shows the top 10 countries with the highest number of late deliveries in the world.
* Orders from Estados Unidos tend to have more late deliveries followed by Francia, Mexico, etc.

**Insights & Recommendations**

After going through the analysis of the 15 features, the Random Forest model basically could be able to predict whether the new order will be late delivery or not. Through this prediction model, businesses are not only able to identify potential late delivery orders but also are able to take proactive interventions to avoid late delivery issues.

For instance, if there is a new order that is predicted that it will be late delivery, businesses could identify which are the main causes of the late delivery through the model and from there take the right action like changing the shipping modes, or having dynamic routing based on situations to keep the order delivered on time. Besides, if the order is unavoidable to be late delivered, businesses could at least inform the customer in advance for further compensation in order to retain customer satisfaction.

# 5.0 Conclusion

With the completion of this project, we have successfully addressed critical supply chain issues such as late deliveries by developing a predictive analytics model using historical order and shipment data. The main objective of this project was to anticipate which orders are at risk of not being delivered on time. We achieved this by building and evaluating four classification models. For instance, Decision Tree,Random Forest, LightGBM and XGBoost. Among these, the Random Forest model outperformed the other models. It achieved an accuracy rate of 94.9% as well as both a strong recall and precision score. Therefore, this model was determined to be the most efficient in correctly identifying both the delayed and on-time deliveries. Through the utilisation of feature selection , we have identified 15 key variables that significantly influenced delivery outcomes which includes factors such as order timing,customer segment,shipping modes and regional factors.

Based on our analysis, we have uncovered valuable insights. For instance, Latin America and Central America had the highest instances of late deliveries. The consumer customer segment was the most affected compared to the corporate and home office segments.Moreover, January stood out as the month with the highest number of delivery delays. This suggests that there is a seasonal pattern. These insights have provided us with a strong foundation for making recommendations that businesses can implement to improve delivery performance.

Moving on, organizations can use this model to identify the high-risk shipments and intervene early. For example, they can upgrade to premium shipping options or adjust delivery routes. High-risk regions such as Latin America can be improved with better logistics planning. Customer satisfaction can be preserved through automated alerts and transparency when delays are anticipated. It is also recommended that the model be updated regularly to adapt to changing patterns in supply chain behaviour.

In a nutshell, this project demonstrates how data mining can be used effectively to solve real-world operational problems, helping businesses improve their decision-making , reduce delivery risks as well as improving the overall customer experience.

# 6.0 References

Dierker, D., Greenberg, E., Saxon, S., & Tiruneh, T. (2022, March 14). *Navigating the current disruption in containerized logistics*. McKinsey & Company.<https://www.mckinsey.com/industries/logistics/our-insights/navigating-the-current-disruption-in-containerized-logistics>

Harter, A., Stich, L., & Spann, M. (2024). The Effect of Delivery Time on Repurchase Behavior in Quick Commerce. Journal of Service Research, 28(2). <https://doi.org/10.1177/10946705241236961>

Matteo Gabellini, Civolani, L., Calabrese, F., & Bortolini, M. (2024). A Deep Learning Approach to Predict Supply Chain Delivery Delay Risk Based on Macroeconomic Indicators: A Case Study in the Automotive Sector. Applied Sciences, 14(11), 4688–4688. <https://doi.org/10.3390/app14114688>

Mohammed, I., & Mandal, J. (2024). The Impact of Lead Time Variability on Supply Chain Management. International Journal of Supply Chain Management, 8(2), 41–55. <https://doi.org/10.47604/ijscm.3075>

Mokin, V. (2024). *Merging FE & Prediction—Xgb, lgb, logr, linr*. Kaggle. <https://kaggle.com/code/vbmokin/merging-fe-prediction-xgb-lgb-logr-linr>

Scikit-learn. (2025). *11. Common pitfalls and recommended practices*. Scikit-Learn. <https://scikit-learn/stable/common_pitfalls.html>

Williamson, C. (2024, August 2). Shipping delays impact global supply chains and exports. S&P Global Market Intelligence. <https://www.spglobal.com/market-intelligence/en/news-insights/research/shipping-delays-impact-global-supply-chains-and-exports-jul24>

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 7.0 Appendix

Video presentation link: <https://youtu.be/EshDQ5H4CQ4?si=k1XW3Mj5OFAnqetl>

# 7.0 Appendices: peer evaluation