

---

# GreenLink Supply Chain Footprint Analyzer

---

*Leveraging Data Warehousing, Machine Learning, and Analytics for Sustainable Supply Chain Management*

**Author:**

Yamannage Sachith Nimesh

**Date:**

2024.11.24

**Institution/Organization:**

University of Ruhuna

## Contents

1. Introduction .....	4
1.1 Background .....	4
1.2 Problem Statement .....	4
1.3 Aim and Objectives .....	4
1.3.1 Aim .....	4
1.3.2 Objectives .....	5
2. Methodology .....	6
2.1 Data Warehousing .....	6
2.1.1 Objective .....	6
2.1.2 Process .....	6
2.1.3 Outcome .....	6
2.2 Data Integration and Transformation .....	6
2.2.1 Objective .....	6
2.2.2 Process .....	6
2.2.3 Outcome .....	7
2.3 Machine Learning Analysis .....	7
2.3.1 Objective .....	7
2.3.2 Process .....	7
2.3.3 Outcome .....	7
2.4 Dashboard Development .....	7
2.4.1 Objective .....	7
2.4.2 Process .....	7
2.4.3 Outcome .....	8
Summary .....	8
3. Results and Findings .....	9
3.1 Creating the Data Warehouse for GreenLink Supply Chain Footprint Analyzer .....	9
3.1.1 Objective .....	9
3.1.2 Step-by-Step Implementation .....	9
3.1.3 Outcome .....	10
3.2 Merging Tables Using Data Factory .....	10
3.2.1 Implementation Steps .....	10
3.2.2 Outcome .....	11
3.3 Analysis and Machine Learning in Fabric Data Science Notebook .....	12

3.3.1 Set Up the Data Science Notebook .....	12
3.3.2 Data Preprocessing.....	12
3.3.3 Train Machine Learning Model .....	13
3.3.4 Visualization.....	13
3.3.5 Outcome.....	13
3.4 GreenLink Supply Chain Footprint Analyzer Dashboard Description .....	14
3.4.1 Key Metrics Overview .....	14
3.4.2 CO <sub>2</sub> Delivery Under Supplier .....	14
3.4.3 Sum of Average CO <sub>2</sub> Per KM by Transportation Mode .....	14
3.4.4 Geospatial Distribution of CO <sub>2</sub> Emissions.....	14
3.4.5 Production Overview .....	14
3.4.6 CO <sub>2</sub> Per Delivery by Supplier .....	15
3.4.7 Conclusion .....	15
4. Conclusion .....	16
Future Suggestions.....	16
Final Thoughts .....	17
5. References.....	18

# 1. Introduction

The global push for sustainability has brought supply chain operations into sharp focus as a significant contributor to carbon emissions. Organizations are increasingly seeking innovative ways to monitor, optimize, and reduce their environmental impact while maintaining operational efficiency. To address these challenges, the **GreenLink Supply Chain Footprint Analyzer** was developed as a comprehensive data-driven platform to track, predict, and visualize carbon emissions across various supply chain activities.

## 1.1 Background

Supply chains are complex systems involving multiple stakeholders, including manufacturers, suppliers, transportation providers, and customers. These systems contribute significantly to global greenhouse gas emissions due to factors such as energy-intensive production processes, long-distance transportation, and inefficient resource utilization. Despite the availability of data, many organizations lack an integrated framework to analyze and act on their emissions data effectively.

Emerging technologies such as data warehouses, machine learning, and advanced visualization tools offer immense potential for tackling these challenges. By integrating these tools into a cohesive framework, organizations can gain actionable insights into their carbon footprint and make informed decisions to optimize their sustainability efforts.

## 1.2 Problem Statement

The primary challenge faced by organizations lies in the lack of a centralized, scalable platform for managing and analyzing supply chain emissions data. Key issues include:

- Fragmented data across production, transportation, and supplier operations.
- Limited visibility into the drivers of emissions and their relative contributions.
- Inability to forecast emissions accurately for strategic planning.
- Absence of actionable insights for implementing low-carbon alternatives.

These limitations hinder organizations from achieving their sustainability goals and responding effectively to regulatory and market pressures.

## 1.3 Aim and Objectives

### 1.3.1 Aim

To develop a robust and scalable framework, the **GreenLink Supply Chain Footprint Analyzer**, which integrates data warehousing, machine learning, and visualization to optimize supply chain sustainability by tracking, predicting, and reducing carbon emissions.

### 1.3.2 Objectives

**1. Centralize Data Management:**

- Build a star-schema data warehouse to consolidate data from production, transportation, and supplier operations.
- Ensure data integrity and scalability for downstream analysis.

**2. Analyze and Predict Emissions:**

- Perform exploratory data analysis (EDA) to uncover patterns and correlations in emissions data.
- Train machine learning models to predict carbon emissions based on factors such as distance traveled, units produced, and delivery count.

**3. Provide Actionable Insights:**

- Develop a Power BI dashboard to visualize key metrics and predictive insights.
- Highlight emission hotspots and recommend low-carbon alternatives for production and logistics.

**4. Support Strategic Decision-Making:**

- Enable real-time tracking of emissions trends.
- Empower organizations to make data-driven decisions to achieve their sustainability targets.

By combining advanced analytics with intuitive visualization, this project aims to bridge the gap between data and actionable sustainability strategies, creating a transformative impact on supply chain operations.

## 2. Methodology

The **GreenLink Supply Chain Footprint Analyzer** was developed using a systematic approach that combines data engineering, machine learning, and visualization to provide actionable insights into supply chain sustainability. The methodology consisted of four major phases:

### 2.1 Data Warehousing

#### 2.1.1 Objective

To centralize and organize supply chain data in a structured format for efficient analysis and machine learning.

#### 2.1.2 Process

- A **workspace** was created in Microsoft Fabric to manage the project's resources.
- Data from various sources, such as transportation, production, and suppliers, was ingested into a **Lakehouse** and stored as raw CSV files.
- A **data warehouse** was set up to structure the data using a **star schema**:
  - A **fact table** contained key metrics, such as distances traveled, units produced, and delivery counts.
  - **Dimension tables** represented transportation details, production attributes, and supplier information.
- Relationships between the fact and dimension tables were established to facilitate querying and downstream processes.

#### 2.1.3 Outcome

A robust star-schema data warehouse served as the foundation for advanced analytics and predictive modeling.

### 2.2 Data Integration and Transformation

#### 2.2.1 Objective

To merge and clean the data for seamless analysis and feature engineering.

#### 2.2.2 Process

- Using **Dataflow Gen2** in Microsoft Fabric's Data Factory, the tables were merged into a unified dataset called the **Company Merged Table**.
- Data cleaning included:
  - Filling missing values with appropriate defaults.
  - Detecting and removing outliers using statistical techniques.

- Feature engineering was performed to create additional insights from the merged data, ensuring consistency and usability in downstream tasks.

### 2.2.3 Outcome

A comprehensive and clean dataset was prepared for exploratory analysis and machine learning.

## 2.3 Machine Learning Analysis

### 2.3.1 Objective

To develop a predictive model for estimating carbon emissions across various supply chain activities.

### 2.3.2 Process

- **Exploratory Data Analysis (EDA)** was conducted to understand the distribution of key metrics and identify patterns or correlations.
- Multiple machine learning models were trained using the prepared dataset, focusing on features like distance traveled, units produced, and delivery count.
- Hyperparameter tuning and cross-validation were applied to optimize the model's performance.
- The best-performing model was selected based on evaluation metrics, demonstrating the ability to predict emissions with high accuracy.

### 2.3.3 Outcome

The trained machine learning model provided reliable predictions for carbon emissions, supporting actionable insights for improving supply chain sustainability.

## 2.4 Dashboard Development

### 2.4.1 Objective

To visualize the results and insights for stakeholders, enabling data-driven decision-making.

### 2.4.2 Process

- A **Power BI dashboard** was developed, integrating key metrics, geospatial analysis, and ML predictions.
- The dashboard included:
  - An overview of supply chain metrics such as total deliveries, distances, and units produced.
  - Detailed breakdowns of emissions by supplier, transportation mode, and production attributes.

- Predictive insights from the machine learning model, enabling stakeholders to forecast future emissions based on input features.
- Dynamic filters were implemented to allow users to interactively explore the data.

### **2.4.3 Outcome**

The dashboard served as a powerful tool for monitoring and optimizing supply chain operations, combining historical analysis with predictive capabilities.

## **Summary**

This methodology ensured a systematic approach to building the GreenLink Supply Chain Footprint Analyzer. Each phase was designed to address specific challenges, from data centralization and integration to advanced analytics and visualization. The framework supports organizations in making informed decisions to reduce their carbon footprint and improve overall supply chain efficiency.



## 3. Results and Findings

### 3.1 Creating the Data Warehouse for GreenLink Supply Chain Footprint Analyzer

#### 3.1.1 Objective

To establish a star-schema data warehouse for the GreenLink Supply Chain Footprint Analyzer project. This warehouse will act as the centralized data repository, integrating all supply chain-related data for downstream analysis and machine learning.

#### 3.1.2 Step-by-Step Implementation

1. **Workspace Creation:**

- A new workspace titled "**GreenLink Supply Chain Footprint Analyzer**" was created in **Microsoft Fabric** to serve as the project's central environment.

2. **Lakehouse Setup:**

- A **Lakehouse** named `SCM_LH` (Supply Chain Management Lakehouse) was created to act as the staging area for data ingestion.
- The following CSV tables were imported into the Lakehouse:
  - **Production Table:** Contains information about materials, energy sources, and CO<sub>2</sub> emissions per unit of production.
  - **Transportation Table:** Details about transport modes, fuel types, and average CO<sub>2</sub> emissions per kilometer.
  - **Supplier Table:** Includes supplier information, locations, and average CO<sub>2</sub> emissions per delivery.
  - **Company Table (Fact Table):** The central fact table containing metrics like distance traveled, units produced, and delivery counts, along with foreign keys linking to dimension tables.

3. **Data Warehouse Creation:**

- A **Data Warehouse** named `SCM` was created to structure and manage the data.
- Using **Dataflow Gen2**, the four tables were transferred from the `SCM_LH` Lakehouse to the `SCM` Data Warehouse for relational data management.

4. **SQL Table Creation:**

- The imported tables were converted into SQL tables within the `SCM` Data Warehouse.
- These tables maintain a structured format, facilitating queries and transformations required for analytics.

5. **Star Schema Design:**

- The tables were linked to create a **star schema**:
  - The **Company Table** acts as the central **fact table**.
  - The **Production Table**, **Transportation Table**, and **Supplier Table** serve as **dimension tables**.
- Relationships between the tables were established via foreign keys:

- ProductionID in the **Fact Table** links to the **Production Table**.
- TransportID in the **Fact Table** links to the **Transportation Table**.
- SupplierID in the **Fact Table** links to the **Supplier Table**.
- The resulting schema enables efficient querying and analysis for business intelligence and machine learning workflows.

### 3.1.3 Outcome

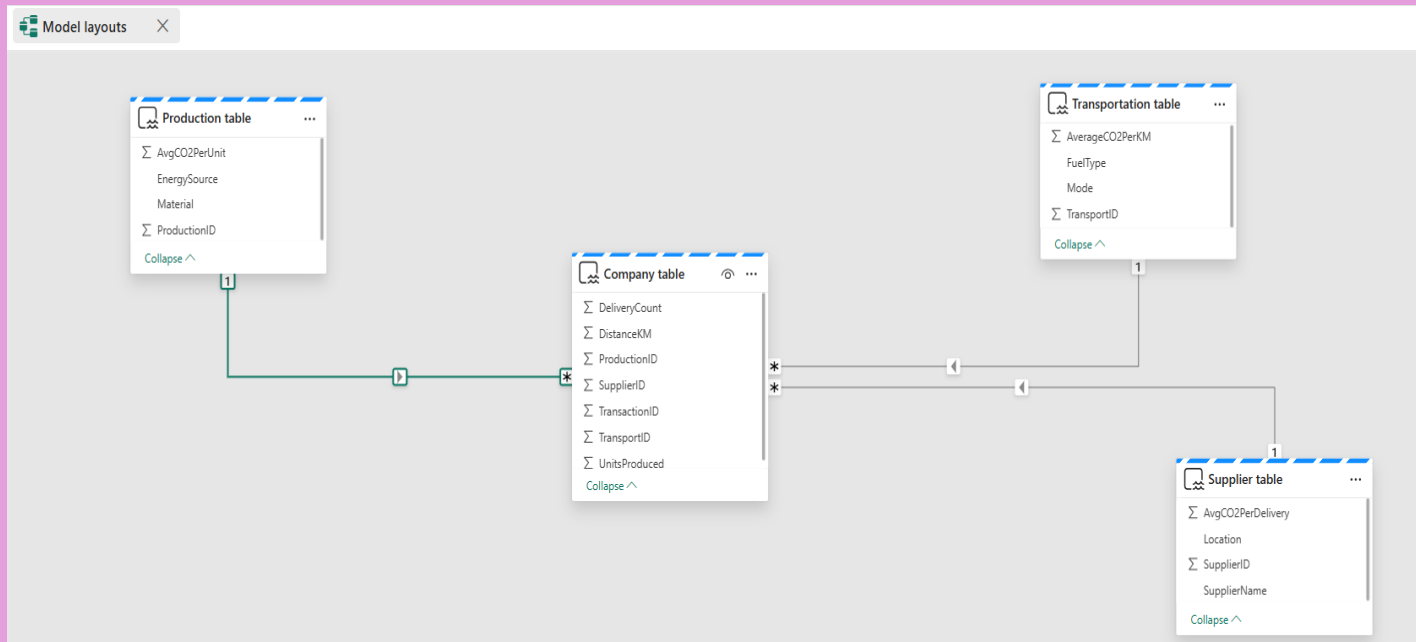


Figure 1 Star schema Datawarehouse

## 3.2 Merging Tables Using Data Factory

To facilitate advanced analytics and machine learning, the dimension and fact tables were merged into a single table, consolidating all key attributes required for the GreenLink Supply Chain Footprint Analyzer. This phase utilized **Dataflow Gen2** in Microsoft Fabric's Data Factory for seamless table merging.

### 3.2.1 Implementation Steps

1. **Setup in Data Factory:**
  - Navigated to **Dataflow Gen2** in **Microsoft Fabric Data Factory**.
  - Established a new data flow specifically for merging the dimension and fact tables.
2. **Table Import:**
  - Imported the four tables—**Production Dimension Table**, **Supplier Dimension Table**, **Transportation Dimension Table**, and **Fact Table**—from the **Lakehouse**.

- Ensured that the column headers and data types were correctly interpreted during the import process.
3. **Merging Process:**
- Sequentially merged the tables to create a unified dataset:
    - Merged the **Fact Table** with the **Production Dimension Table** on the `ProductionID` key.
    - Joined the resulting table with the **Supplier Dimension Table** on the `SupplierID` key.
    - Finally, merged with the **Transportation Dimension Table** on the `TransportID` key.
  - The resulting table was named **Company Merged Table**, encapsulating metrics such as:
    - Production details (e.g., material, energy source, CO<sub>2</sub> per unit).
    - Supplier information (e.g., location, CO<sub>2</sub> per delivery).
    - Transportation details (e.g., fuel type, CO<sub>2</sub> per kilometer).
    - Fact metrics like distance traveled and units produced.
4. **Data Validation and Transformation:**
- Applied transformation steps such as renaming columns for consistency, standardizing data types, and handling missing or inconsistent data.
  - Verified the merged table for data integrity and alignment with the project requirements.
5. **Output Destination:**
- Saved the **Company Merged Table** back into the **Lakehouse**, enabling downstream processes such as machine learning and visualization.

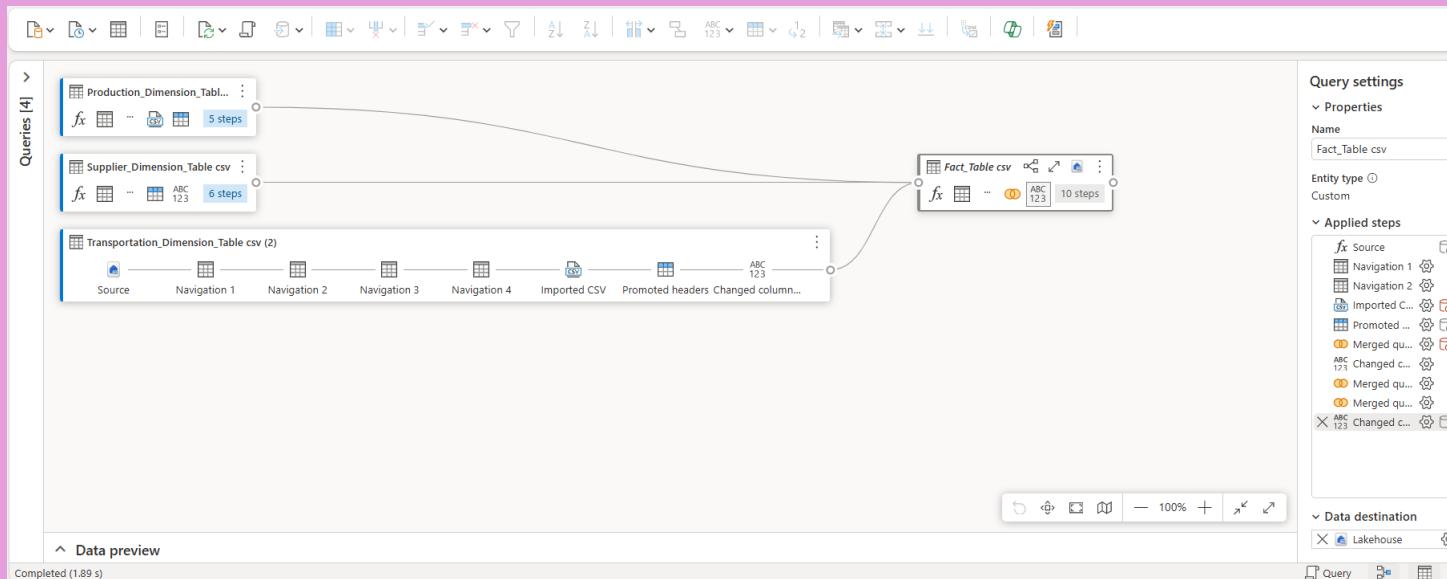


Figure 2 Dataflow model

### 3.2.2 Outcome

The **Company Merged Table** serves as a comprehensive dataset that integrates all supply chain dimensions. This unified structure simplifies querying, facilitates efficient feature engineering, and provides the foundation for machine learning models and dashboard development.

## 3.3 Analysis and Machine Learning in Fabric Data Science Notebook

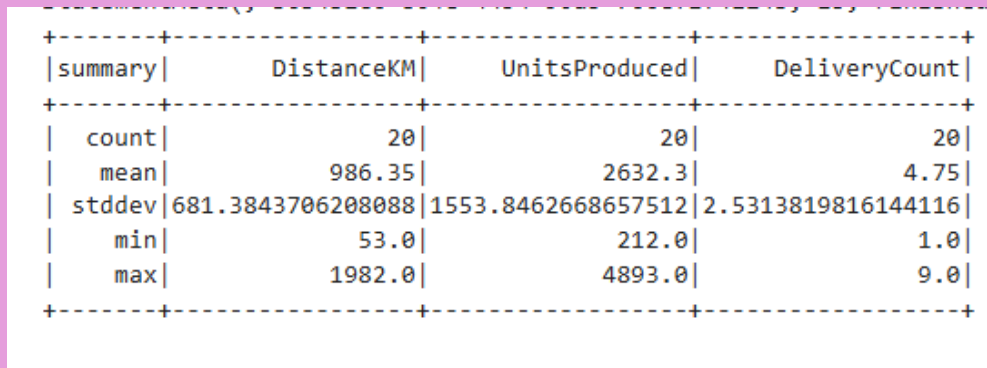
### 3.3.1 Set Up the Data Science Notebook

The analysis was performed in the **Fabric Data Science Notebook**. The `MergedFactTable` from the data warehouse was connected and used as the primary dataset for this phase. This setup allowed seamless integration with Spark-based tools and machine learning libraries for preprocessing, analysis, and model training.

### 3.3.2 Data Preprocessing

#### 1. Exploratory Data Analysis (EDA):

- **Distribution Analysis:** The dataset's key numerical features, including `DistanceKM`, `UnitsProduced`, and `DeliveryCount`, were analyzed for distributions. The statistics revealed:
  - Average distance traveled: 986.35 km.
  - Average units produced: 2632.3 units.
  - Average delivery count: 4.75.



summary	DistanceKM	UnitsProduced	DeliveryCount
count	20	20	20
mean	986.35	2632.3	4.75
stddev	681.3843706208088	1553.8462668657512	2.5313819816144116
min	53.0	212.0	1.0
max	1982.0	4893.0	9.0

Figure3 : Summary Statistics

- **Correlation Matrix:** Relationships between features were studied, indicating weak correlations:
  - `DistanceKM` vs. `UnitsProduced`: -0.386.
  - `DistanceKM` vs. `DeliveryCount`: -0.198.
  - `UnitsProduced` vs. `DeliveryCount`: -0.242.
- 2. **Handling Missing and Outlier Values:**
  - Missing values in key columns were filled with defaults (e.g., 0 for numeric fields).
  - Outliers in `DistanceKM` were removed using z-scores with a threshold of 3.
- 3. **Feature Normalization:**
  - A **VectorAssembler** combined `DistanceKM`, `UnitsProduced`, and `DeliveryCount` into a feature vector.
  - Features were standardized using **StandardScaler** for improved model performance.

### 3.3.3 Train Machine Learning Model

#### 1. Model Selection and Training:

- A **Random Forest Regressor** was chosen for its ability to handle non-linear relationships and interpret feature importance.
- The dataset was split into **80% training** and **20% testing** sets.
- Hyperparameter tuning was performed using a parameter grid for the number of trees.

#### 2. Evaluation:

- The trained model was evaluated on the test set, yielding an RMSE (Root Mean Squared Error) of **0.3369**, indicating good predictive accuracy.

### 3.3.4 Visualization

Below is a visualization of the model evaluation metric:

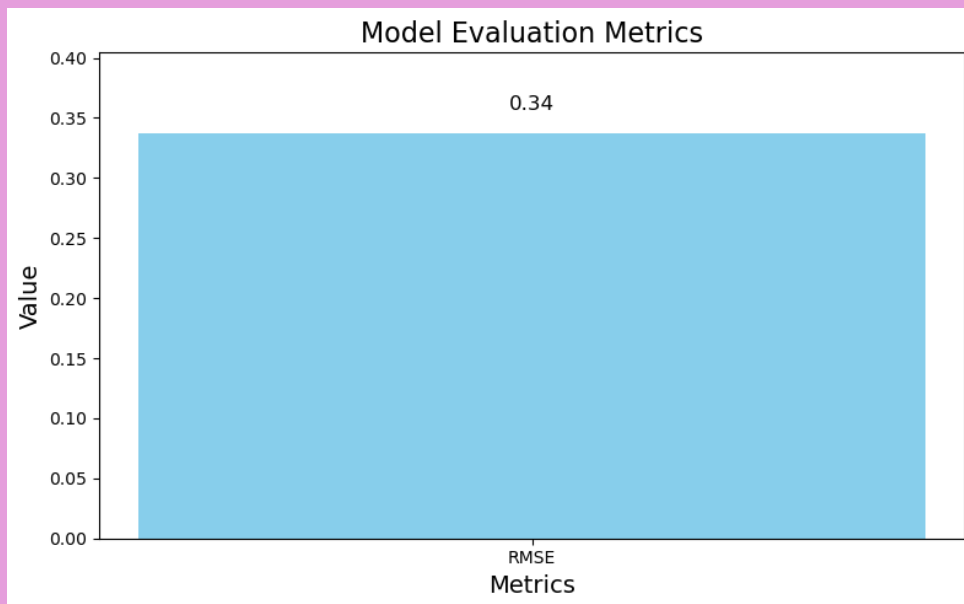


Figure 3 Evaluation plot

### 3.3.5 Outcome

The analysis revealed a reliable model for predicting CO<sub>2</sub> emissions based on features such as distance traveled, units produced, and delivery count. The trained **Random Forest Regression Model** with optimized parameters demonstrated strong performance with low error rates.

This structured approach ensures robust predictions, enabling actionable insights for sustainability improvements in supply chains.

## 3.4 GreenLink Supply Chain Footprint Analyzer Dashboard Description

The **GreenLink Supply Chain Footprint Analyzer** dashboard provides a comprehensive overview of key metrics and insights related to the sustainability of supply chain activities. It integrates descriptive and predictive analytics, enabling stakeholders to monitor and optimize carbon emissions across various dimensions of the supply chain.

### 3.4.1 Key Metrics Overview

- **Delivery Count (Top Left):** Displays the total number of deliveries made across the supply chain (e.g., 95).
- **Distance Traveled (Top Center):** Highlights the total kilometers traveled by all transportation modes (e.g., 20K km).
- **Units Produced (Top Right):** Indicates the total units produced by manufacturing activities (e.g., 53K units).
- **Average CO<sub>2</sub> Emissions Per Unit (Top Far Right):** Shows the average carbon emissions generated per unit of production (e.g., 1.98 CO<sub>2</sub>/unit).

### 3.4.2 CO<sub>2</sub> Delivery Under Supplier

- **Visualization Type:** Horizontal Bar Chart
- **Purpose:** Shows the contribution of different suppliers to CO<sub>2</sub> emissions per delivery.
- **Insight:** Helps identify suppliers with the highest carbon footprint, allowing focused interventions.

### 3.4.3 Sum of Average CO<sub>2</sub> Per KM by Transportation Mode

- **Visualization Type:** Bar Chart
- **Purpose:** Displays the average CO<sub>2</sub> emissions per kilometer for different transportation modes such as Plane, Truck, Ship, Train, and Drone.
- **Insight:** Enables analysis of the most eco-friendly and least eco-friendly transportation options.

### 3.4.4 Geospatial Distribution of CO<sub>2</sub> Emissions

- **Visualization Type:** Map
- **Purpose:** Displays the geographical locations of suppliers and their respective CO<sub>2</sub> delivery contributions.
- **Insight:** Helps identify emission hotspots across different continents (e.g., North America, Europe, Asia).

### 3.4.5 Production Overview

- **Visualization Type:** Pie Chart
- **Purpose:** Displays the proportion of production under different IDs.
- **Insight:** Provides a high-level breakdown of production activities.

3.4.6 CO<sub>2</sub> Per Delivery by Supplier

- **Visualization Type:** Line Chart
- **Purpose:** Displays the CO<sub>2</sub> emissions per delivery across different suppliers.
- **Insight:** Highlights how different suppliers compare in terms of carbon efficiency, allowing optimization.

3.4.7 Conclusion

This dashboard enables users to analyze and interpret sustainability metrics across the supply chain. By combining geospatial, categorical, and predictive visualizations, it provides actionable insights for reducing the carbon footprint of supply chain operations.

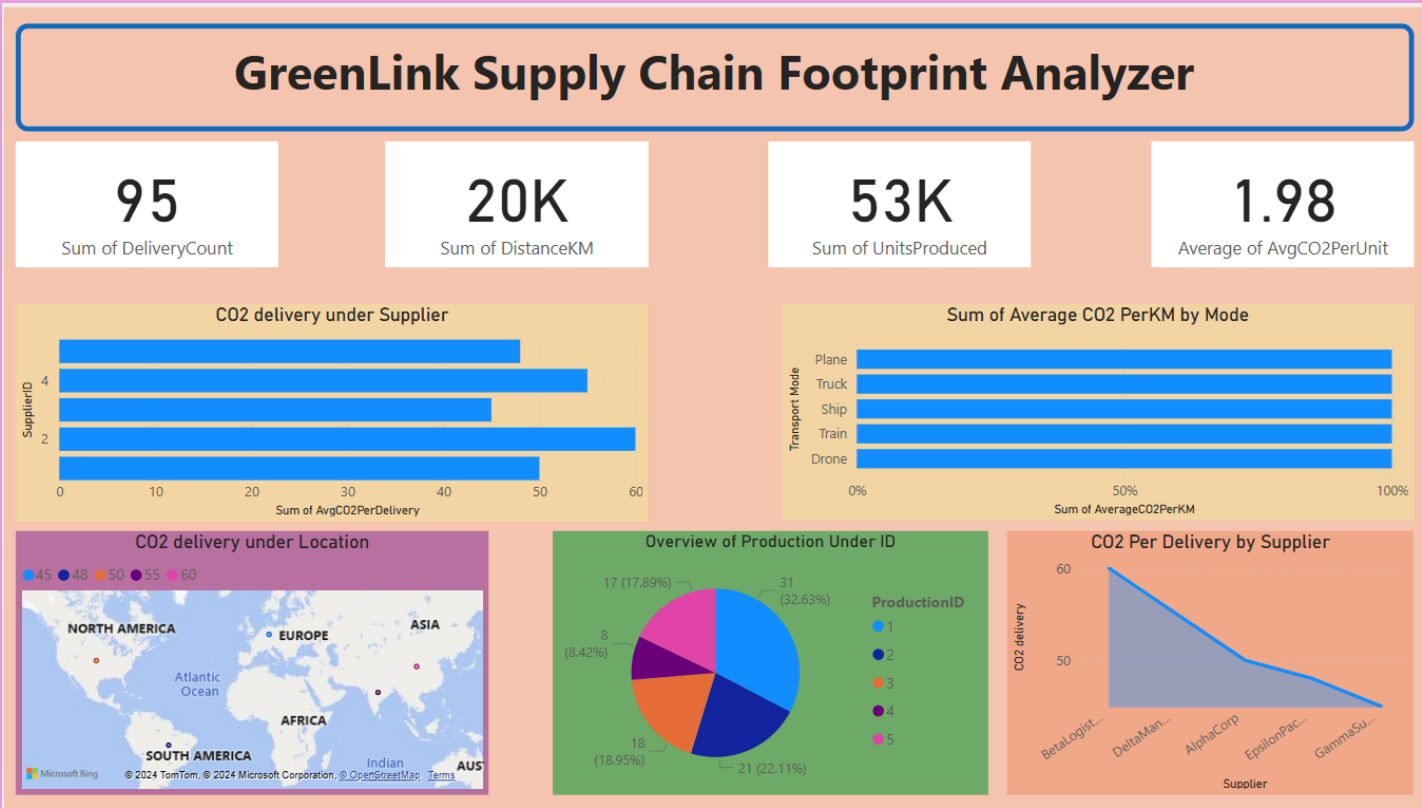


Figure 4 Dashboard

## 4. Conclusion

The **GreenLink Supply Chain Footprint Analyzer** successfully addressed the critical need for a robust, data-driven framework to track, analyze, and predict carbon emissions across supply chain operations. By integrating advanced data warehousing, machine learning, and visualization techniques, the project provided actionable insights to drive sustainability initiatives while optimizing supply chain efficiency.

The project demonstrated the potential of leveraging modern tools such as Microsoft Fabric, machine learning models, and Power BI dashboards to centralize data, uncover hidden patterns, and predict emissions effectively. The use of a star-schema data warehouse ensured scalable and structured data management, enabling seamless querying and integration. Machine learning models, particularly the Random Forest Regressor, exhibited strong predictive capabilities for CO<sub>2</sub> emissions based on features such as distance traveled, units produced, and delivery count. The interactive dashboard further empowered stakeholders with a comprehensive view of emissions data, including historical analysis, real-time tracking, and future projections.

## Future Suggestions

While the project achieved its objectives, there are several areas where further enhancements can be implemented to increase the scope and impact of the solution:

1. **Integration of Additional Data Sources:**
  - Incorporate real-time IoT sensor data for transportation and production.
  - Add weather, fuel price trends, and route optimization data to refine predictions.
2. **Advanced Machine Learning Techniques:**
  - Experiment with deep learning models, such as neural networks, for more complex relationships in emissions data.
  - Explore ensemble learning techniques for enhanced predictive accuracy.
3. **Scenario Analysis and Simulation:**
  - Implement "what-if" scenarios in the dashboard to allow users to simulate the impact of changes in supplier selection, transportation modes, or production methods on emissions.
4. **Automation and Feedback Loops:**
  - Automate data ingestion pipelines for real-time updates to the data warehouse and dashboard.
  - Integrate feedback loops where predictions are validated against actual emissions and used to retrain models.
5. **Geospatial and Supply Chain Optimization:**
  - Use geospatial analysis to identify optimal supplier locations and logistics routes for reducing emissions.
  - Implement supply chain optimization algorithms to balance cost, emissions, and delivery efficiency.
6. **Sustainability Metrics Integration:**
  - Expand the scope to include other sustainability metrics such as water usage, energy efficiency, and waste generation.
7. **Scalability and Deployment:**



- Deploy the system as a cloud-based solution to support larger datasets and multi-regional supply chains.
- Extend the solution to integrate with enterprise resource planning (ERP) systems for end-to-end visibility.

## **Final Thoughts**

The **GreenLink Supply Chain Footprint Analyzer** sets the foundation for leveraging data science and advanced analytics to achieve supply chain sustainability. By building on the existing framework and adopting the proposed enhancements, organizations can not only reduce their carbon footprint but also gain a competitive advantage by aligning with global sustainability goals. This project underscores the importance of integrating technology and data-driven insights into addressing critical environmental challenges.

## 5. References

1. Microsoft. (n.d.). *Data Warehouse Overview in Fabric*. Retrieved from <https://learn.microsoft.com/en-us/fabric/data-warehouse>
2. Microsoft. (n.d.). *Dataflows Gen2 in Microsoft Fabric*. Retrieved from <https://learn.microsoft.com/en-us/fabric/dataflow>
3. Microsoft. (n.d.). *Data Science in Fabric*. Retrieved from <https://learn.microsoft.com/en-us/fabric/data-science>
4. Microsoft. (n.d.). *Using Power BI to Create Dashboards and Reports*. Retrieved from <https://learn.microsoft.com/en-us/power-bi/create-reports/service-interactive-dashboards>
5. Scikit-learn. (n.d.). *RandomForestRegressor Documentation*. Retrieved from <https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html>
6. Microsoft Azure. (n.d.). *Azure Machine Learning: Build and Deploy Machine Learning Models*. Retrieved from <https://learn.microsoft.com/en-us/azure/machine-learning>
7. Microsoft Power BI. (n.d.). *Geospatial Visualization in Power BI*. Retrieved from <https://learn.microsoft.com/en-us/power-bi/visuals/power-bi-map>
8. Python Software Foundation. (n.d.). *StandardScaler in scikit-learn*. Retrieved from <https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html>
9. Microsoft Fabric. (n.d.). *Lakehouse Architecture in Fabric*. Retrieved from <https://learn.microsoft.com/en-us/fabric/lakehouse>
10. Microsoft. (n.d.). *Introduction to Machine Learning with Fabric Notebooks*. Retrieved from <https://learn.microsoft.com/en-us/fabric/notebooks>