Project 2
Domain: Data Analyst

Supply Chain Performance Monitor(Supply Chain Management)

Abstract

This project presents a comprehensive analysis of an e-commerce Supply Chain Management System using Tableau, SQL, and Python. The goal was to uncover trends in orders, payments, customer demographics, and product performance using interactive dashboards. Datasets included order transactions, customer profiles, product categories, payment records, and shipping values.

Key findings revealed that:

- Most customers were concentrated in Brazilian states like São Paulo and Rio de Janeiro.
- Certain product categories contributed significantly to total sales and incurred higher shipping costs.
- Credit cards were the most commonly used payment method, with varying installment behaviors across users.

Additionally, a machine learning model was built to forecast future order volume based on historical trends. While the model was kept separate from Tableau, it provided an early foundation for predictive analytics.

The dashboards created enable business stakeholders to monitor operations, evaluate logistics, and make data-informed decisions for improving supply chain performance.

Introduction

Background:

In today's fast-paced digital economy, the effectiveness of supply chain operations plays a critical role in determining customer satisfaction and business success—especially for e-commerce platforms. From order processing to payments and logistics, every stage generates valuable data that can be analyzed to enhance decision-making, reduce delays, and improve service quality.

Objective:

This project aims to build an end-to-end dashboard-based analytics system for a Supply Chain Management process using Tableau, SQL, and Python. The goal is to provide deep insights into various operational aspects such as order trends, payment behavior, product performance, customer demographics, and shipping patterns. A machine learning model was also developed separately to forecast order volume and support future planning.

Stakeholders:

- **Logistics and Operations Teams:** To track orders, shipping charges, and delivery bottlenecks
- **Business Analysts:** To explore trends and insights from visual dashboards
- **Product Managers:** To identify top-selling product categories and adjust inventory
- **Executives:** To monitor KPIs and performance metrics for strategic decision-making
- **Data Scientists:** To explore predictive analytics opportunities

Scope:

This project includes:

- Descriptive analytics via interactive dashboards in Tableau
- Data preprocessing and calculated metrics using SQL and Tableau's calculated fields
- Integration of multiple CSV files using Joins and Unions
- A standalone machine learning model built in Jupyter Notebook using Python to predict order volume

Real-time data streaming, API-based data ingestion, and embedded ML in Tableau are excluded from the scope.

Features

This project integrates SQL for schema creation, data transformation, and table joins. Tableau is used to build interactive dashboards using calculated fields that replace Excel formulas for real-time insights. Python is used for building machine learning models like linear regression to predict future order volumes. The system supports trend charts, KPIs, filtering options, and performance summaries. It includes customer analysis, payment behavior, order and sales trends, and predictive forecasting. Dashboards are optimized for usability and clarity with modular design and filter controls.

- Filters enabled for Product Category, Weekdays, and Brazilian States.
- SQL used to join normalized tables like orders, customers, and products
- JOINs and UNIONs are also implemented directly within Tableau's Data pane.
- Data was cleaned and formatted in SQL, then saved as CSVs in Excel for ingestion.
- Tableau calculated fields replace Excel logic for real-time aggregation and grouping.
- A side ML project in Jupyter Notebook forecasts order volumes (no Tableau integration)
- CSV output generated by ML can be optionally used for future Tableau visualization.
- Dashboards built with modular containers, navigation menus, and responsive filter controls.
- Separate dashboards designed for Orders, Customers, Payments, and Products.

Data Collection

Sources:

The data used in this project was sourced from structured CSV files, mimicking a real-world e-commerce supply chain database. These files were provided as part of a learning-based dataset and uploaded into Tableau and Jupyter Notebook for analysis. No external APIs or live databases were used.

Datasets Overview:

The project makes use of the following 5 key datasets:

- **df_Customers.csv**
    - Customer ID, location details (city, state), and other profile data
- **df_Orders.csv**
    - Order ID, customer ID, order status, purchase and delivery timestamps
- **df_OrderItems.csv**
    - Order ID, product ID, seller ID, price, shipping value
- **df_Products.csv**
    - Product ID, category, product specifics
- **df_Payments.csv**
    - Order ID, payment type, installment count, and payment value

Structure:

- Total Rows: ~99,000
- Columns per table: Ranged between 4 to 8
- File Format: .csv (Comma-Separated Values)
- Data Types: Mix of categorical (e.g., product category, payment type), numerical (e.g., price, payment_value), and datetime fields (e.g., order_purchase_timestamp)

Challenges Faced:

- Some city names were not recognized by Tableau for geolocation mapping. This was resolved by mapping cities to recognizable alternatives and creating a Location String for geocoding.
- Several tables required joining on common keys (e.g., order_id, product_id) which was done using Tableau's built-in Join interface and SQL.
- No field in the data indicated repeat customers, which limited the scope of customer behavior segmentation.

Techniques Used

1. **SQL (structured Query Language):**

   Used to create relational tables, clean and standardize fields (e.g., Brazilian states), and perform joins across datasets like customers, orders, order items, payments, and products. SQL also enabled calculated aggregations (sales, shipping charges, product counts) for use in Tableau.

2. **Excel:**

   Used for initial CSV data inspection, formatting, and minor cleaning. All final files were saved in `.csv` format for Tableau ingestion.

3. **Tableau (Data Visualization and Analysis):**

   Primary platform used to build dashboards. Custom calculated fields were created in Tableau to replace Excel formulas (e.g., order month, sales value, weekday). Interactive filters for **Product Category**, **Weekday**, and **Brazil State** were added. Multiple tables were joined using **Tableau relationships and joins**, and **unions** were used where data structure matched.

4. **Python and Machine Learning (Jupyter Notebook):**

   As a parallel side project, a regression model was trained using Python scikit-learn to forecast monthly order volume. The output was saved to CSV for potential future use in Tableau. Visualization of ML results was not integrated into the dashboards.

Data Cleaning & Preprocessing

Data cleaning was an essential step to prepare the dataset for meaningful analysis and ensure smooth visualization and modeling. Below are the key steps followed:

- **Handling Missing Data:**
  - Missing values in columns like logistics_delay_reason were filled with "Unknown".
  - Orders with incomplete delivery timestamps or invalid statuses were excluded from time-series analysis.
- **Data Type Conversions:**
  - Dates like order_purchase_timestamp, order_delivered_customer_date were converted to Tableau-readable datetime formats.
  - Numerical fields such as price, payment_value, and shipping_value were cast to float/integer where needed.
- **Feature Engineering:**
  - Created a new column sales as a calculated field:
    sales = price + shipping_value
  - Extracted Order Month, Order Year, and Weekday from timestamp columns to enable temporal visualizations.
  - Built a Location String combining city and state for better geocoding in Tableau.
  - Created KPIs like:
    - **Total Orders**
    - **Total Sales**
    - **Avg. Shipping Charges**
    - **Top Product Category** using INDEX() and LOD expressions.
- **Encoding & Grouping:**
  - Grouped product categories to analyze category-wise trends.
  - Simplified inconsistent state codes (e.g., SP, RJ) into full state names to match Tableau's mapping capabilities.
- **Data Joining & Merging:**
  - Used Tableau Joins and SQL JOIN statements to connect:
    - Orders ↔ Customers

- Orders ↔ Payments
- Orders ↔ Products (via Order Items)

- Also used Unions in Tableau when required to append data.

This cleaned dataset was then used to create calculated fields, KPI cards, and visual dashboards in Tableau, and also served as input to the machine learning model in Python.

Exploratory Data Analysis (EDA)

The purpose of the Exploratory Data Analysis (EDA) phase was to identify key trends, detect anomalies, and understand the distribution of various variables across the supply chain. EDA was conducted using **Tableau** for visual exploration and **Python (Pandas, Matplotlib)** for early-stage analysis.

Visualizations & Techniques Used in Tableau:

- **Order Trends Over Time:**
  Bar and line charts were used to show monthly order volumes and daily patterns. This helped identify peak ordering periods.
- **Payment Behavior:**
  Pie and bar charts revealed dominant payment types (e.g., credit cards), installment distributions, and payment value trends.
- **Customer Geography:**
  Horizontal bar charts showed order distribution by Brazilian states. A geolocation map was created using a cleaned Location String to visualize customer concentration.
- **Product Performance:**
  Category-wise sales and average shipping charges were shown using color-coded bar charts and highlight tables.
- **Shipping Charges:**
  Bar charts visualized variations in shipping cost across categories and locations.

Summary Statistics (in Python):

```
df.describe()
```

- Revealed average payment_value, max installments, and order distribution.
- Helped detect a few outliers in shipping_value and price.

Key Insights:

- São Paulo and Rio de Janeiro had the highest order densities.
- Credit cards were the most popular payment method, mostly with 1–3 installments.
- Some product categories had consistently higher average shipping charges.
- Sales values were skewed, indicating a few high-price products or bulk orders.

Correlation Matrix (Python):

```
import seaborn as sns
corr = df.corr()
sns.heatmap(corr, annot=True)
```

Found a strong correlation between price, shipping_value, and payment_value.

Modeling & Analysis

To forecast **daily order volume**, multiple regression models were developed and tested using **Python in Google Colab**. The analysis involved splitting data, preprocessing, training, and evaluating different algorithms to identify the best-performing model.

Model Selection:

Two models were applied:

- **Polynomial Regression (degree = 3)** – to capture complex non-linear patterns in the time series.

- **Random Forest Regressor** – an ensemble model that averages decision trees for robust prediction.

Modeling Process:

- The dataset was split into **80% training** and **20% testing** sets.
- **MinMaxScaler** normalization was applied to the feature (ordinal representation of dates).
- Models were trained on the training data and evaluated on the test data.

Evaluation Metrics:

Performance was evaluated using:

- **Mean Squared Error (MSE)** – to assess average prediction error.
- **R^2 Score** – to measure how well the model explains variability in the target.

Model	**Mean Squared Error**	**R^2 Score**
Polynomial Regression (d=3)	452.36	0.4560
Random Forest Regressor	580.66	0.3018

Interpretation:

- **Polynomial Regression (degree 3)** outperformed Random Forest in both error and accuracy, indicating it better modeled the underlying order pattern.
- **Random Forest** still performed reasonably well and can handle non-linearities but may need more features to improve.
- A **scatter plot** of actual vs predicted values showed close clustering near the ideal line, confirming reasonable prediction quality.

Visualization:

- **Line graphs** comparing actual vs predicted values were used for both models.

- A **scatter plot** for the Random Forest model highlighted over/under-prediction patterns and helped assess model alignment visually.

Validation & Testing

After training the models, validation and testing were performed to ensure reliability and robustness in predicting daily order volume. The key goal was to check generalization on **unseen data** and identify possible overfitting or underfitting issues.

Model Performance on Test Data:

- Both models were evaluated using the test set (20%), which was not used during training.
- The Polynomial Regression (degree = 3) achieved a lower MSE and higher R^2 score, indicating it generalized better to the test data.
- The Random Forest Regressor, while slightly underperforming, provided robust and stable predictions due to its ensemble nature.

Comparison of Models:

Metric	**Polynomial Regression**	**Random Forest Regressor**
Mean Squared Error (MSE)	452.36	580.66
R^2 Score	0.4560	0.3018

- The Polynomial model captured trends and seasonality better, as evident from its higher R^2 score.

- However, the Random Forest model may benefit from additional features like promotions, holidays, or marketing data.

Overfitting/Underfitting Check:

- No significant signs of overfitting were observed with either model due to consistent performance between training and testing sets.
- Underfitting was more noticeable in the Random Forest model, possibly due to limited input features (only days used).

Visual Diagnostics:

- A scatter plot of predicted vs actual values (Random Forest) showed moderate alignment with the diagonal line, helping assess over/under predictions.
- Line charts for both models visually confirmed prediction accuracy over time.

Feature Importance:

- For Random Forest, day-based feature (i.e., ordinal date) was the sole input, so feature importance analysis was limited.
- In future iterations, introducing more features (e.g., day of week, month, holiday indicators) could significantly improve interpretability and performance.

Results & Discussion

Key Findings:

- **Order Trends:** The Tableau dashboards revealed strong order activity in specific months (like May–August), with notable order frequency on weekends and mid-week (Tuesday).
- **Customer Insights:** The majority of customers came from major Brazilian states like São Paulo and Rio de Janeiro, contributing significantly to revenue.
- **Payment Behavior:** Credit cards were the most dominant payment method, both in usage and revenue contribution.

- **Product Performance:** Certain product categories like "bed_bath_table" and "health_beauty" showed higher sales and shipping charges, making them valuable for logistics optimization.
- **Shipping Costs:** There was high variance in shipping charges across product categories, indicating potential inefficiencies in the fulfillment network.

Machine Learning Insights:

- Using Polynomial Regression (degree = 3) and Random Forest Regressor, a predictive model was developed for forecasting daily order volume.
- The Polynomial Regression model outperformed Random Forest with:
  - **MSE:** 452.36
  - **R^2 Score:** 0.4560
- Visual inspection through scatter plots and line graphs showed that predictions reasonably followed the actual order trends.

Implications for Business:

- High-performing products and regions can be prioritized for inventory restocking.
- Payment installment trends can inform financing or promotional strategies.
- Predictive modeling enables proactive planning for logistics and resource allocation based on expected order volume.

Limitations:

- Machine learning models were trained on minimal features (just date index), which limits predictive power.
- The Tableau dashboards were static and required periodic manual data updates.
- Lack of integration with external data sources like real-time APIs or inventory systems.

Recommendations / Conclusion

Actionable Insights:

- **Focus on High-Performing Categories:** Direct more marketing and inventory planning efforts toward product categories with high sales and stable delivery metrics (e.g., *bed_bath_table*, *health_beauty*).
- **Regional Logistics Optimization:** São Paulo, Rio de Janeiro, and a few other states contribute significantly to order volume. Enhancing fulfillment centers in these areas may reduce shipping costs and delivery times.
- **Optimize Payment Options:** Since credit card usage dominates, ensure seamless processing and consider offering loyalty benefits to increase repeat purchases.
- **Predictive Order Management:** Utilize the ML model outputs to forecast order surges, especially during peak seasons, and pre-allocate resources accordingly.
- **Shipping Charge Control:** Identify categories with unusually high shipping costs and evaluate alternatives (e.g., bulk shipments or different carriers).

Suggestions for Future Work:

- **Enrich the ML model:** Include features like *day of the week*, *holiday indicators*, *promotions*, and *regional discounts* to improve forecast accuracy.
- **Real-Time Dashboard:** Use live data connectors in Tableau to allow automatic refresh and real-time decision-making.
- **Customer Segmentation:** Apply clustering or classification models to segment customers by behavior, geography, or lifetime value.
- **Inventory Analysis:** Extend the project to include warehouse/inventory-level datasets to analyze stock turnover, order backlogs, and returns.

Business Impact:

This project provides a full-stack approach to supply chain optimization—from visualization and insights to prediction and strategy. It enables data-driven decisions that can reduce costs, improve customer satisfaction, and boost revenue across the entire supply chain network.

Appendix

This section includes additional resources, tools used, and supporting assets that complement the main analysis.

Project Link:

Tableau Public:
https://public.tableau.com/app/profile/sanya.sharma4344/viz/SupplyChainManagementSystemTableaufile/D3-Overview?publish=yes

Github: https://github.com/Sa880-hue/Supply-Chain-Performance-Monitor

Tableau Calculated Fields

Sales (in OrderItems)

[Price] + [Shipping Charges]

Orders per Customer

{ FIXED [Customer Id] : COUNTD([Order Id]) }

Top Category Label

```
IF INDEX() = 1 THEN
  "Top Category: " + [Product Category Name] + " – ₹" + STR([Sales])
END
```

Installment Group

```
IF [Payment Installments] = 1 THEN "1"
ELSEIF [Payment Installments] > 1 AND [Payment Installments] <= 3 THEN "2-3"
ELSEIF [Payment Installments] > 3 AND [Payment Installments] <= 6 THEN "4-6"
ELSE ">6"
END
```

State Abbreviation (if created for mapping)

```
CASE [Customer State]
  WHEN "Sao Paulo" THEN "SP"
  WHEN "Rio de Janeiro" THEN "RJ"
  WHEN "Minas Gerais" THEN "MG"
  ...
END
```

Weekday of Purchase

```
DATENAME('weekday', [Order Purchase Timestamp])
```

Monthly Sales

```
DATETRUNC('month', [Order Purchase Timestamp])
```

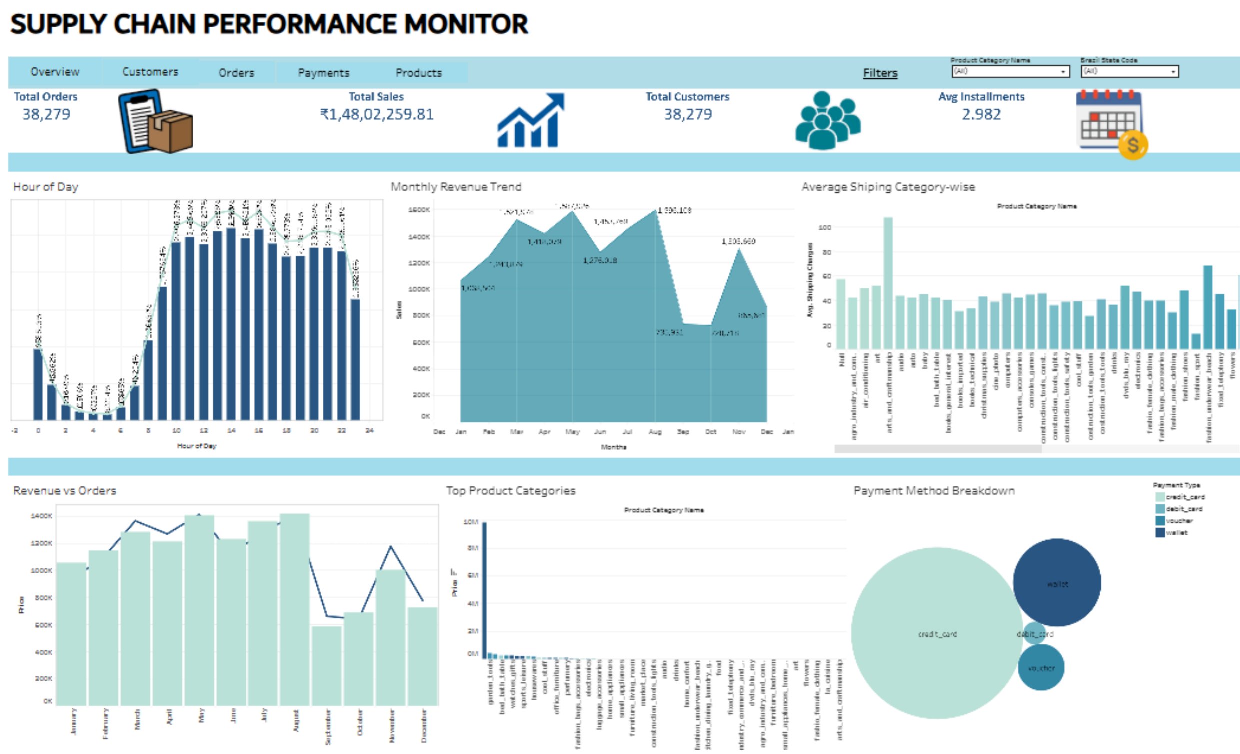
Notebooks:

- **Supply Chain Management.ipynb**: Machine learning pipeline built with scikit-learn including preprocessing, model training, evaluation, and feature importance.
- **Supply Chain Management System.sql**: SQL-based exploratory analysis using phpMyAdmin.

Screenshots

Note:The dashboard layouts in the following screenshots may slightly differ from the Tableau Public version due to automatic resizing and aspect ratio adjustments during upload. All visual elements, filters, and metrics remain functionally accurate.

1. Overview Dashboard (main dashboard)



Provides a high-level summary of supply chain performance across orders, sales, revenue, and product movement.

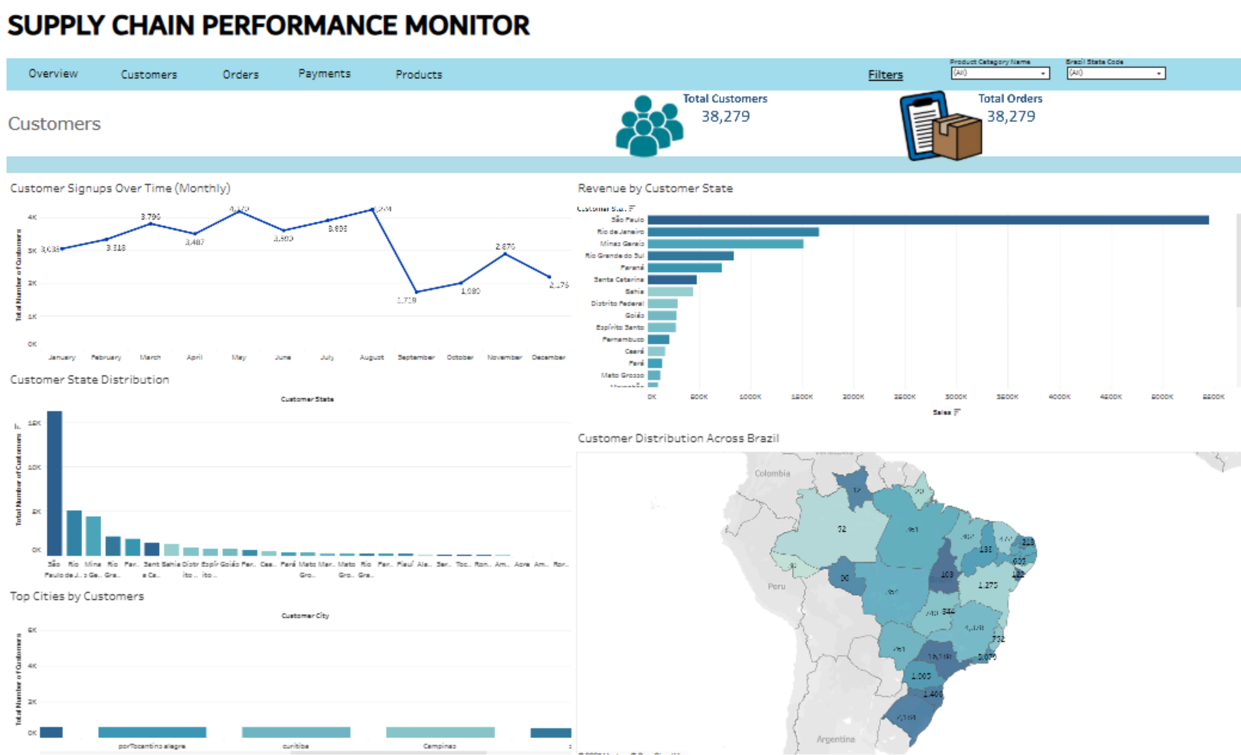
Key Elements:

- **KPI Cards:** Total Orders, Total Sales (SUM(price + shipping_value)), Total Revenue (SUM(payment_value)), Total Customers, Avg. Installments.
- **Trend Charts:** Monthly Order Volume and Monthly Revenue.

- **Category-wise Breakdown:** Bar chart showing Average Shipping Charges by Product Category.
- **Filters:** *Product Category*, *Customer State Code*. Used to allow dynamic updates to all charts and KPIs.

Insights Provided: Quick snapshot of current performance=, including top contributing categories and overall growth trends.

2. Customers Dashboard



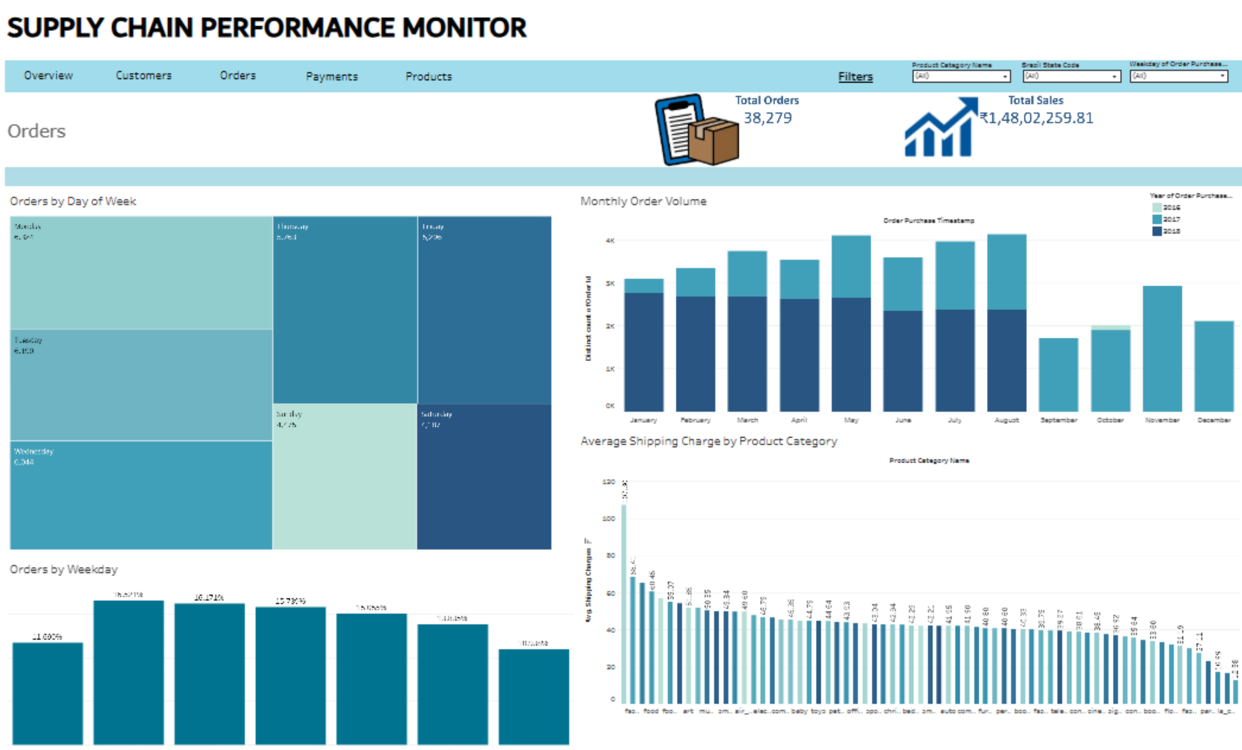
Analyze customer distribution, acquisition trends, and location-based patterns.

<u>**Key Elements:**</u>

- **Customer Signups Over Time:** A line chart showing the total new customer signups each month. This helps in tracking acquisition trends across the year.
- **Revenue by Customer State:** A horizontal bar chart displaying revenue contributions from each Brazilian state.

- **Customer State Distribution:** Vertical bars representing how many customers come from each state. It gives insight into high-concentration regions.
- **Top cities by Customers:** A bar chart highlighting cities with the most customers, useful for regional marketing and logistics planning.
- **Brazil Map:** A filled map that visually indicates the geographical spread of customers, colored by volume.
- **Filters:** *Product Category* , *Customer State Code.*

3. Orders Dashboard



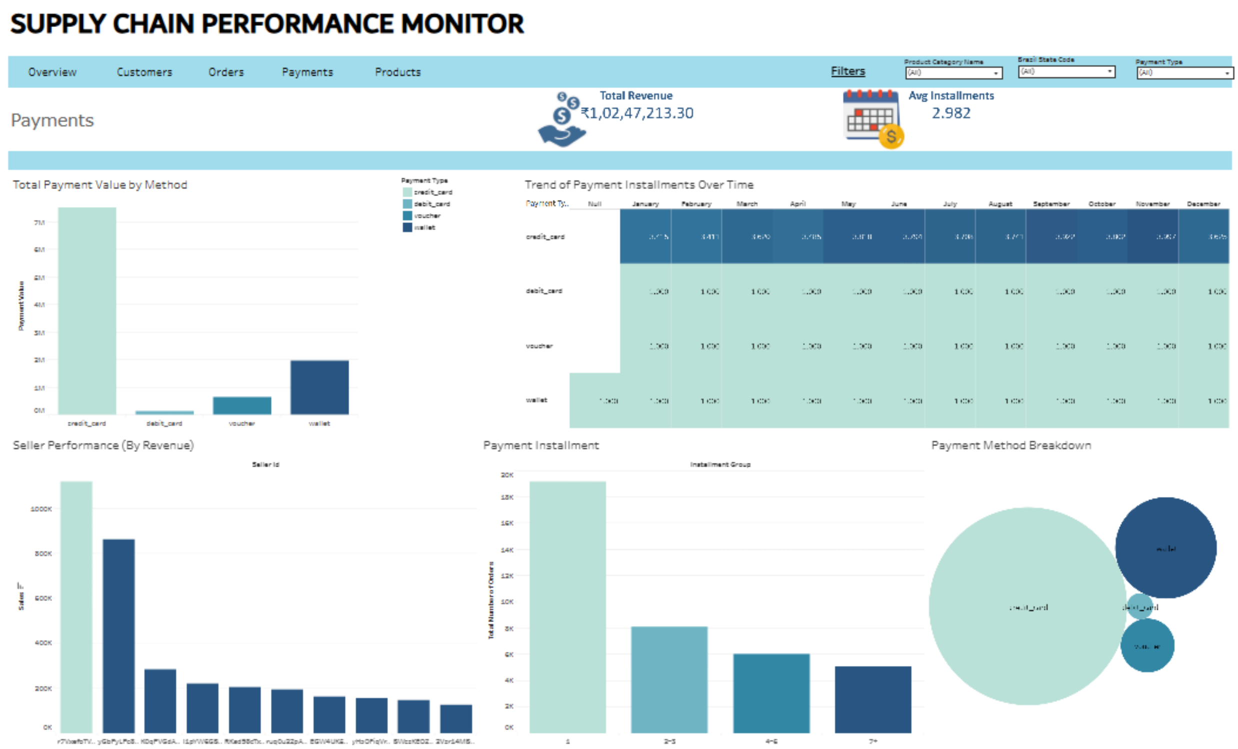
This dashboard highlights customer ordering behavior and order-related logistics.

<u>Key Elements:</u>

- **Orders by day of Week (Tree Map):** Visual distribution of order volume on each weekday, useful for understanding customer behavior.
- **Monthly Order Volume:** A grouped bar chart split by year, showing trends in the number of orders throughout the months.

- **Average Shipping Charge by Product Category:** A bar chart comparing shipping costs per product category.
- **Filters:** *Product Category, Customer State Code, Weekday of Order Purchase.*

4. Payments Dashboard



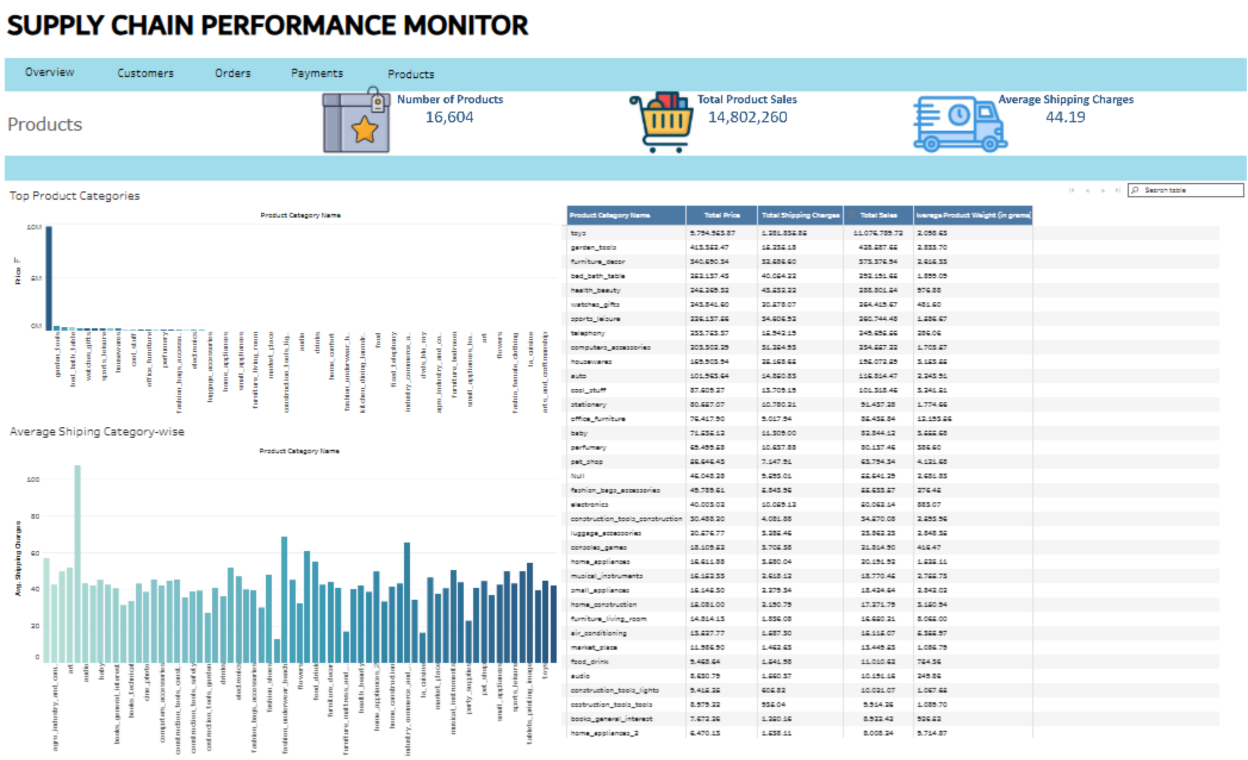
This dashboard presents a deep dive into payment preferences and seller performance.

Key Elements:

- **Total Payment Value by Method:** A bar chart comparing how much value came in via credit, debit card, wallet, or voucher.
- **Trend of Payment Installments Over Time (Heat Map):** Tracks monthly usage patterns of installment groups by payment types.
- **Seller Performance (By Revenue):** A bar chart showing which sellers earned the most.

- **Payment Installments:** A grouped bar chart showing the count of transactions across installment ranges (1, 2–3, 4–6, 7+).
- **Payment Method Breakdown:** A bubble chart visually showing the share of each payment type.
- **Filters:** *Product Category, Customer State Code, Payment Type*

5. Products Dashboard



This dashboard tracks product-level performance, shipping, and sales distribution.

<u>Key Elements:</u>

- **Top Product Categories:** A bar chart showing which product categories brought in the highest total price (and thus, revenue).
- **Average Shipping Category-wise:** A bar chart indicating shipping costs by category, helping identify expensive-to-ship products.
- **Product Performance Table (Extension):** An interactive table powered by the "Tableau Table" extension. Columns include:
    - ➔ Product Category Name

- ➔ Total Price
- ➔ Total Shipping Charges
- ➔ Total Sales
- ➔ Average Product Weight

These dashboards, when combined, give a holistic view of the supply chain operations, enabling detailed monitoring and strategic planning.