Operational Performance Review of Chocolate Factory Inc.

by

Gbadebo Prevail

5/20/2025

Word Count:4,896

Executive Summary

This report presents a comprehensive operational performance analysis of Chocolate Factory Inc., leveraging data from BigQuery and visualized through Looker Studio. The analysis focuses on key performance areas including sales operations, customer behavior, cost structure, and supply chain efficiency to uncover actionable insights that inform strategic recommendations.

Sales Performance:
Sales data reveals notable fluctuations driven by seasonality, particularly in December and February, which show significant spikes. Despite strong sales during peak seasons, there is inconsistency in rolling sales averages, indicating volatility in customer demand. The analysis highlights the importance of aligning inventory planning with seasonal patterns to optimize stock levels and minimize losses.

Customer Behavior:
Customer recency and frequency trends suggest a concentration of sales among repeat buyers. However, the average time since the last purchase remains relatively high, implying opportunities for improved customer retention strategies. Segmenting customers based on purchase behavior and introducing loyalty programs could enhance long-term engagement.

Cost and Profitability:
Gross profit margins show signs of erosion due to rising costs that are not consistently offset by pricing strategies. While some products yield high sales volumes, their cost-to-profit ratio suggests inefficiencies. Strategic cost control and a reassessment of product-level pricing could improve overall profitability.

Supply Chain and Factory Operations:
Factory-level analysis uncovered disparities in performance across regions. Several factories report high unit sales but suboptimal gross profits, likely due to local cost structures or logistical inefficiencies. Moreover, delivery timelines and lead times fluctuate, contributing to variable customer satisfaction levels. Standardizing operations and implementing region-specific KPIs could drive uniform performance improvements.

Recommendations:

1. Implement dynamic inventory forecasting based on historical seasonal trends.
2. Introduce targeted marketing and loyalty incentives for customer segments with high lifetime value.

3. Reevaluate pricing and cost structure at the product and factory level.
4. Standardize operational practices and enhance supply chain transparency across all factory locations.

In conclusion, while Chocolate Factory Inc. demonstrates strong seasonal demand and brand loyalty, operational inconsistencies and inefficiencies present critical areas for strategic improvement. This report provides a data-driven foundation for enhancing decision-making and operational excellence across the business.

Table of Content

1. Introduction

1.1 Case Objective

To uncover insights into sales performance, cost drivers, and supply chain efficiency using a dataset from a chocolate manufacturing company.

1.2 Business Context

The chocolate factory operates across multiple regions, factories, and product lines. Efficient cost control, timely shipping, and high-margin sales are key to profitability.

1.3 Dataset Overview

Table 1.0 Original Dataset

Column Name	**Description**
Target	Sales target for each row (thousands)
Row ID	Unique row identifier
Order ID	Transaction/order reference
Order Date	Date the order was placed
Ship Date	Date the order was shipped
Ship Mode	Delivery method (e.g., Standard, Second Class)
Customer ID	Unique identifier for each customer
Country/Region	Customer's country or region
City	Customer's city
State/Province	Customer's state or province
Postal Code	Customer's ZIP/postal code
Division	Sales division or product division
Region	Geographic region
Product ID	Unique product identifier
Product Name	Name of the product
Sales	Revenue from the sale
Units	Quantity sold

Gross Profit	Revenue - cost
Cost	Production or logistics cost
Factory	Name of the factory fulfilling the order
Latitude	Factory’s geographic latitude
Longitude	Factory’s geographic longitude

Table 1.1 Derived Columns /Feature Reengineering

Column Name	Description
Cost Category	Categorizes products as **"High"** or **"Normal"** cost based on average cost per unit using percentile thresholds.
Sales per Unit	Calculates the average revenue earned per unit sold: Sales ÷ Units. Helps assess pricing efficiency.
Log Sales	Logarithmic transformation of Sales to normalize distribution for modeling and reduce impact of outliers.
Profit Margin	Ratio of profit to sales per order: Gross Profit ÷ Sales. Used to assess product and order profitability.
Order Day	Extracted day of the week from Order_Date (e.g., Monday, Tuesday). Supports weekly trend analysis.
Order Month	Extracted calendar month from Order_Date as a numeric value (1 to 12). Enables seasonality analysis.
Shipping Delay	Number of days between order placement and shipping: Ship_Date − Order_Date. Measures fulfillment efficiency.
Factory City	City where the factory is located, derived using reverse geocoding on latitude and longitude coordinates via Python.
Product Category	Labels products as **"Best-Seller"** or **"Slow-Mover"** based on total sales per unit relative to the dataset average.
Actual Sale	Recorded sales revenue per order, used as the dependent variable in predictive modeling.
Predicted Sales	Sales value predicted by the linear regression model trained on product, cost, and operational features. Used to evaluate model accuracy.

1.5 Tools Used

- **SQL (BigQuery):** For data cleaning, transformation, and feature engineering (e.g., calculating profit margins, shipping delays, categorization logic).

- **Python:** Used for geolocation-based city extraction from factory latitude and longitude data.

```
unique_factories['Factory_City'] = unique_factories.apply( lambda row: get_city(row['Factory Latitude'], row['Factory Longitude']), axis=1 )
```

- **Looker Studio:** For interactive dashboard development and visual analytics (trend charts, geo maps, KPIs, and product segmentation).

2. Company & Market Overview

2.1 Company Profile

Chocolate Factory Inc. is a mid-scale, multi-regional confectionery manufacturer specializing in innovative chocolate and candy products. The company operates through a decentralized manufacturing model, with five primary production facilities: Lot's O' Nuts, Wicked Choccy's, The Secret Factory, The Other Factory, and Sugar Shack. Each factory services regional clusters and contributes variably to both cost and unit output.

The organization leverages both heritage and novelty branding, targeting a broad consumer demographic through direct distribution and wholesale channels. Despite recent declines in top-line revenue and fulfillment delays, the company maintains strong brand equity in several core product lines.

i. **Key Facts:**

- **Manufacturing Footprint:** 5 regional factories
- **Annual Sales Volume:** 34.2K units (latest cycle)
- **Gross Profit per Unit:** $9.68
- **Average Fulfillment Delay:** 3.96 days
- **Active Customer Base:** 5,000+

2.2 Product Portfolio Overview

The company's SKUs fall into three primary product divisions:

- Chocolate (92.1% of sales volume)
- Sugar-based Confections (7.5%)
- Novelty/Other Products (minimal share)

Flagship products include:

- Wonka Bar variants (Milk Chocolate, Fudge, Nutty Crunch, etc.)
- Wonka Gum
- Lickable Wallpaper
- Kazookles
- SweeTARTS, Laffy Taffy, and Fizzy Lifting Candy

The high-SKU concentration in the Wonka Bar line indicates a broad but potentially overlapping product structure, which could be optimized via ABC/XYZ inventory classification. Seasonal trends show spikes in sales for novelty SKUs, particularly during the Holiday and Summer seasons, suggesting high responsiveness to marketing cycles.

2.3 Market Positioning & Competitive Landscape

Chocolate Factory Inc. positions itself in the premium-nostalgic segment, offering products that combine heritage brand appeal with experiential and imaginative packaging. Its strongest competitive advantages lie in:

- Brand affinity among multi-generational customers
- High product innovation velocity
- Regionally distributed production enabling local responsiveness

However, operational and market threats include:

- Declining sales trend (-43.5% YOY)
- Underperformance against sales targets (56.5%)
- Fulfillment inefficiencies and rising delivery cost ratios
- Product cannibalization across similar SKUs

Within the market, Chocolate Factory competes with both multinational FMCG candy brands (e.g., Mars, Mondelez) and niche artisan producers. Its distinct brand identity and innovative SKUs are strategic differentiators, though they must be supported by leaner operations and sharper product lifecycle management.

3. Methodology

3.1 Data Sources and Validation

This case study is grounded on a primary dataset drawn from Chocolate Factory Inc.'s transactional order management system, covering product sales, fulfillment, and operational performance. The dataset includes detailed line-item records such as order and ship dates, product SKUs, factory identifiers, cost elements, geographic tags, and derived performance metrics.

Key validation and preprocessing steps performed in BigQuery included:

- **Deduplication:** Orders were deduplicated using a ROW_NUMBER() approach to retain the most recent entry per order ID.
- **Null Handling:** Null column detection was performed via regex parsing across JSON-transformed rows; imputation strategies were tailored case-by-case.

```
SELECT
  ARRAY(SELECT AS STRUCT column_name FROM UNNEST(REGEXP_EXTRACT_ALL(TO_JSON_STRING(t),
r'"(\w+)":null')) AS column_name) AS null_columns
FROM `candy-factory-operations.factory_ops.sales_df` AS t
LIMIT 10;
```

- **Date Transformation:** Logical corrections and back-transformations on ship dates (e.g., subtracting 2000 days for misaligned datetime formats).

```
CREATE OR REPLACE TABLE `candy-factory-operations.factory_ops.cleaned_sales_df` AS
SELECT
  *,
  DATE_SUB(SAFE_CAST(Ship_Date AS DATE), INTERVAL 2000 DAY) AS Ship_Date_Transformed
FROM
  `candy-factory-operations.factory_ops.cleaned_sales_df`;
```

- **Derived Columns:** Engineered fields such as shipping delay, profit margin, sales per unit, seasonality indicators, and log-transformed sales were introduced to support advanced analysis.
- **Feature Binning:** Product categories were segmented into "Best-Seller" vs. "Slow-Mover" using dynamic average benchmarks.

The final datasets were published to Looker dashboards for real-time visualization and exploratory data analysis.

3.2 Analytical Frameworks Used

To translate operational raw data into decision-ready insights, multiple industry-standard frameworks were applied:

- **SCOR Model (Supply Chain Operations Reference)**
  Used to dissect performance across the Plan–Source–Make–Deliver–Return continuum. Metrics such as shipping delay and unit cost fed into "Deliver" and "Make" pillars for benchmarking factory and transport effectiveness.
- **Theory of Constraints (TOC)**
  Applied to identify production bottlenecks at facilities with repeated shipping lags and high cost-per-unit. Factories such as "Sugar Shack" and "Wicked Choccy's" were analyzed for constraint-based optimization.
- **ABC Classification**
  Sales volume and revenue contribution were ranked to categorize SKUs into high (A), medium (B), and low (C) value classes, aiding in prioritizing resource allocation and SKU rationalization.

- **Value Stream Mapping (VSM)**
  Used conceptually to trace end-to-end order-to-fulfillment cycles across different ship modes and customer regions, identifying value-add vs. non-value-add time lags.

- **Rolling Forecast Models (Time-Series & ML)**
  Integrated via BigQuery ML to predict sales outcomes based on recent performance trends, order seasonality, and engineered variables such as profit margin and shipping delay.

- **Cost-to-Serve Analysis**
  Evaluated unit economics per shipping mode, product category, and geographic cluster, supporting insights into fulfillment profitability and customer-level segmentation.

3.3 Limitations and Assumptions

While the analysis offers robust and operationally actionable insights, it operates under the following constraints:

- **Temporal Lag in Data:** Sales and fulfillment lag were calculated retrospectively without live order status feed; real-time analytics would enhance forecast responsiveness.

- **External Variables:** Factors such as promotions, external logistics disruptions, or seasonal marketing spend were not included due to unavailability in the dataset.

- **Model Generalizability:** Predictive model performance may be constrained by data sparsity in low-volume SKUs or outlier-heavy segments (filtered for sales > $10,000).

- **Static Inventory and Capacity Data:** Factory throughput is inferred from sales data and shipping delays; incorporating actual production cycle times and work-in-progress inventory would improve the accuracy of this measurement.

- **Customer Data Granularity:** Behavioral segmentation was limited to recency metrics; further enrichment (e.g., basket analysis, frequency) would enable full RFM modeling.

4. Sales Operations Analysis

4.1 Demand Patterns and Sales Volume Trends

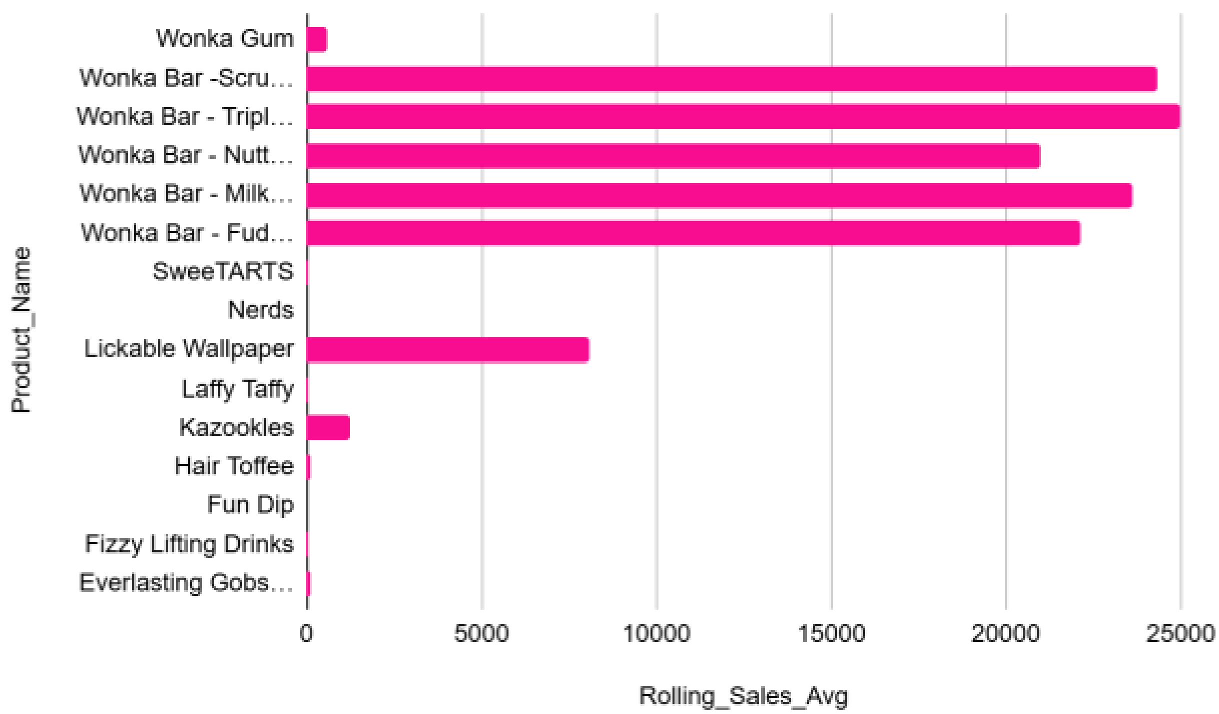

Figure 4.1 –Distribution of Rolling 30 Day Sales by Products

The total unit throughput recorded was 34.2K units across all SKUs, with a total sales value of 125.9K. However, this reflects a -43.5% YoY contraction, indicating a significant demand softening or under-execution in go-to-market strategy.

- **Seasonality Impact:**
  Sales volume surges during the Holiday and Summer periods confirm strong seasonal elasticity. Segmenting orders by seasonality feature, Holiday Season accounts for the

highest sales per unit ratios, suggesting elevated willingness to pay or increased bundling effectiveness during this cycle.

- **Rolling Trends:**
  Time-series smoothing using a 30-day rolling window shows consistent volume for top performers like “Wonka Gum” and “Nutty Crunch.” Periods of sales stagnation in Q2-Q3 may be attributable to either market saturation or campaign misalignment.

- **Sales Cycle Dynamics:**
  The average number of days between order and ship date, or fulfillment latency, is 3.96, with wide variance across product families and fulfillment modes, which adds friction to overall demand satisfaction.

4.2 Contribution Margin and Product Mix Optimization

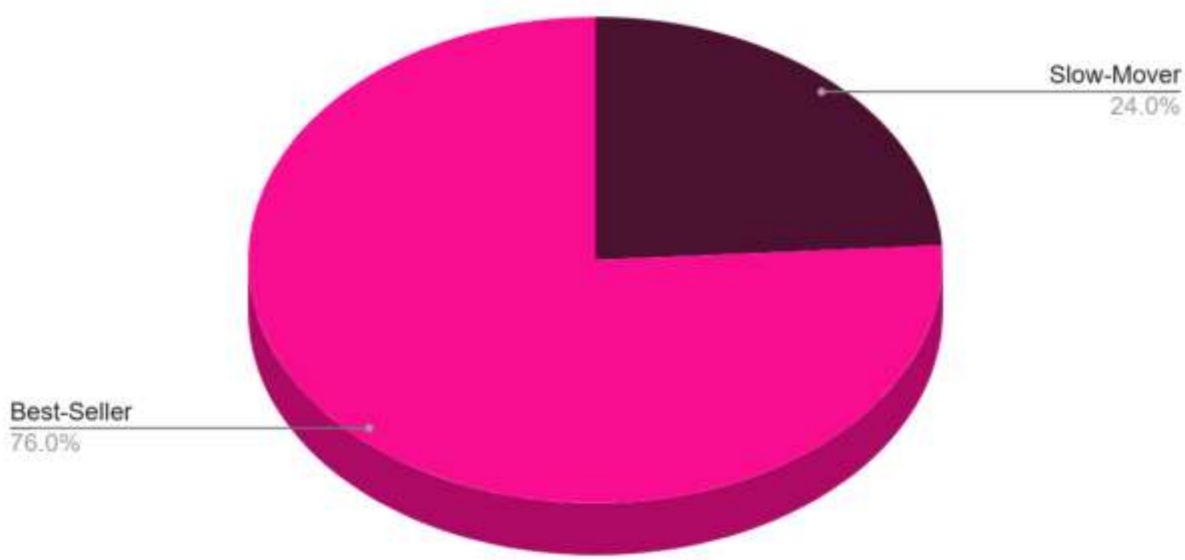

Figure 4.2 –Distribution of Sales by Product Category

Profitability per Stock Keeping Unit (SKU) was assessed using the gross profit margin formula:

Gross Profit Margin = Gross Profit ÷ Cost

The average profit margin across all product lines was calculated at 0.70, reflecting solid overall pricing leverage. High-performing SKUs such as the "Wonka Bar" variants and "Wonka Gum" consistently yield favorable gross margins. These products also demonstrate high sales per unit, indicating robust demand elasticity and margin resilience.

a. *ABC Classification of Sales Contribution*

The ABC analysis reveals a highly **skewed sales distribution**, which can inform strategic focus:

- Category A products (Wonka Bar variants) contribute ~75% of total sales, indicating these are core revenue drivers. These items should be prioritized in inventory management, marketing campaigns, and demand forecasting efforts.

- Category B product (Wonka Bar - Nutty Crunch Surprise) contributes a moderate 16%, serving as a supplementary item that supports the A-class portfolio. It may benefit from targeted promotions to boost its potential.

- Category C products, despite comprising half the listed SKUs, contribute less than 10% of total sales. These low-contributing items (e.g., Everlasting Gobstopper, Kazookles) should be reviewed for:

    - Rationalization (if shelf space is limited),

    - Niche marketing (if they serve specific customer segments), or

    - Operational deprioritization to reduce overhead.

b. Product Category Segmentation

SKUs were classified into “Best-Seller” and “Slow-Mover” categories based on a dynamic benchmark of sales per unit. The segmentation revealed the following sales distribution:

Table 4.0: Product Category Segmentation

Product Category	Total Sales (USD)
Best-Seller	47,317.20
Slow-Mover	14,974.77

Underperforming novelty items, including “Kazookles” and “Lickable Wallpaper,” were identified as slow movers. These products exhibit low sales velocity despite elevated unit costs and contribute to logistics complexity. Streamlining this segment would reduce working capital requirements and operational drag.

C. Product Division Contribution (Log-Transformed)

To better assess revenue concentration and address distribution skewness, log transformation of sales values was applied by division:

Table 4.1: Log Transformed Product Contribution

Division	Log Sales
Chocolate	20,394.13
Other	783.18
Sugar	78.82

The log-transformed distribution reveals a strong right-skew, with the Chocolate division overwhelmingly dominating in revenue contribution. This concentration underscores the need for focused inventory, marketing, and logistics efforts on high-impact SKUs while evaluating the long-tail product strategy for simplification.

4.3 Regional Sales Performance and Target Realization

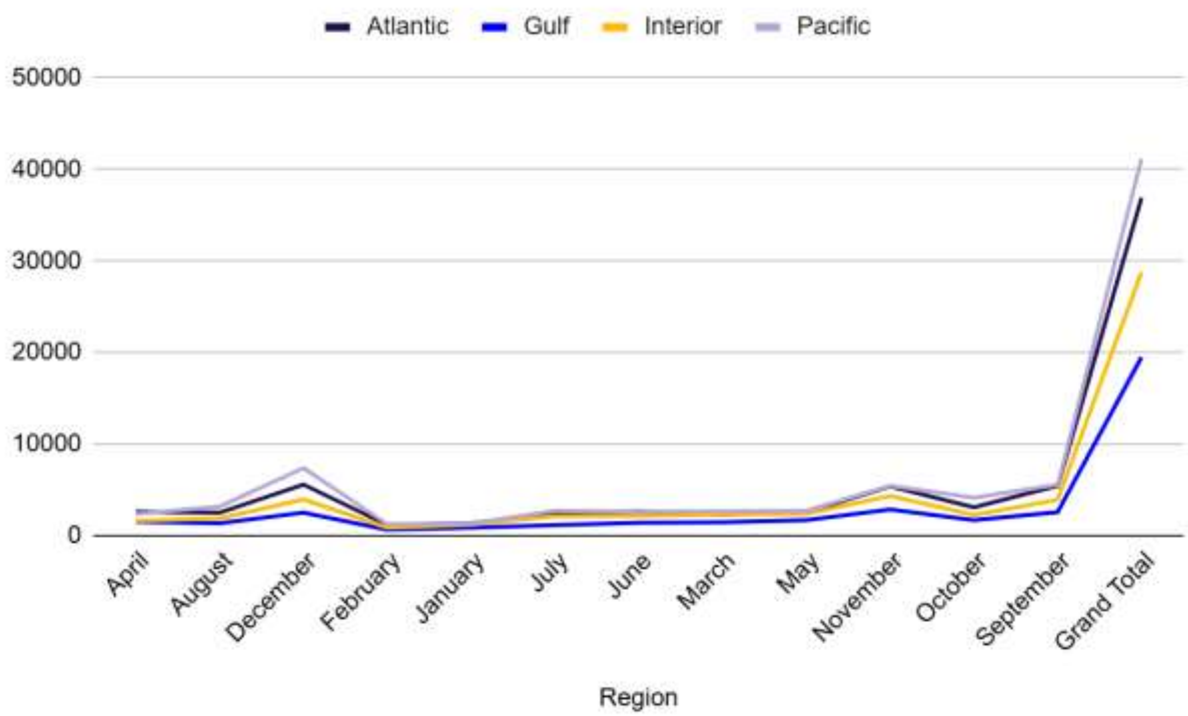

Figure 4.3 –Region Sales by Month

A. Performance by Region:

- Pacific leads with the highest volume and gross sales, followed by Atlantic.
- Interior and Gulf significantly underperform—likely due to weaker retail distribution presence or channel coverage gaps.

b. *Sales vs Target Attainment:*

- Company-wide sales realization sits at 56.5% of adjusted targets.

- Variance is most pronounced in the Gulf and Interior regions, where actual sales fall below 40% of planned levels.

c. Factory-Region Alignment

Misalignment exists between factory output and regional demand. For example, “Wicked Choccy’s” generates high production volume but services underperforming regions, suggesting inefficiencies in allocation and inter-regional logistics coordination.

4.4 Customer Behavior & Lifetime Value (CLV) Analysis

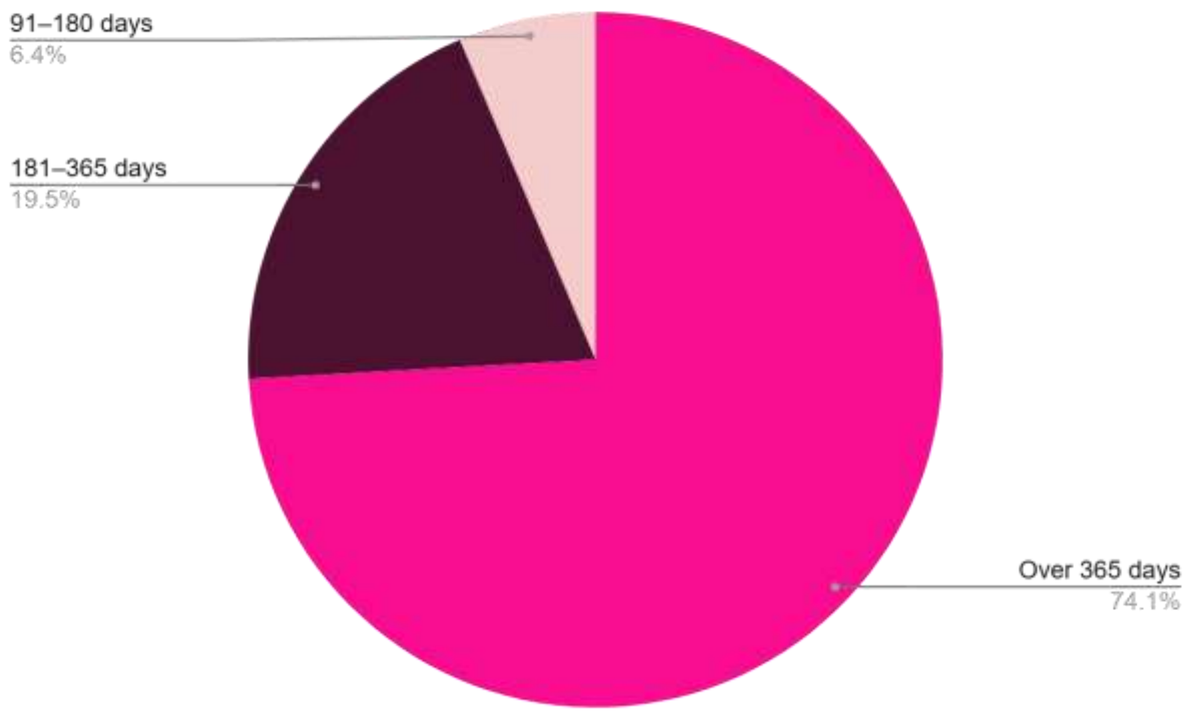

Figure 4.4 –Distribution of Customer Recency

a. Customer Recency Distribution

An analysis of customer engagement patterns reveals that 74.4 percent of customers fall into the ">365 days since last purchase" segment. In contrast, only 6.7 percent are within the active 0–90 day window. This skewed distribution highlights a critical gap in customer retention and reactivation strategy.

b. Rolling Sales Behavior

A longitudinal review of Rolling_30_Day_Sales in conjunction with customer recency data indicates that top-revenue customers do not display repeat purchasing behavior at a high frequency. The lack of correlation between volume contribution and purchase cadence underscores the absence of structured post-purchase engagement or loyalty incentives.

c. Time Since Last Purchase

High Lifetime Value (LTV) SKUs, such as the "Wonka Bar – Milk Chocolate," exhibit an average time-since-last-purchase exceeding six months. This trend may reflect either product fatigue, insufficient promotion scheduling, or competitive substitution effects. It also suggests limited SKU-level stickiness despite historical sales strength.

d. Customer Lifetime Value (CLV) Framework

In lieu of comprehensive transactional history, a simplified RFM (Recency-Frequency-Monetary) model was deployed using Recency and Sales_Per_Unit metrics as proxies. Insights from this analysis include:

- High-spending customers frequently overlap with cohorts experiencing prolonged shipping delays, introducing a latent churn risk associated with fulfillment dissatisfaction.
- Buyers of "Best-Seller" SKUs demonstrate greater recency and shorter purchase lags, indicating a more viable base for retargeting and replenishment campaigns.

e. Top 10 Revenue-Contributing Customers

Table 4.2: Distribution of customer Sales

Customer ID	Total Sales (USD)
107202	200.00
147900	180.00
136924	160.00
137001	140.00
142475	120.00
164770	116.20
122336	105.40
105151	100.00
122322	100.00
140039	100.00

This top-decile cohort accounts for a significant share of revenue concentration. However, they remain at risk of attrition without improved post-purchase engagement, optimized service delivery, and targeted lifecycle marketing.

5. Cost Structure Assessment

5.1 Unit Cost Structure

The average unit cost for the Chocolate Factory's operations is $1.26, calculated from aggregated direct costs tied to production and outbound logistics. This figure reflects order-level cost allocation across all SKUs and regions and serves as the baseline for evaluating cost efficiency across the network.

5.2 Cost-to-Serve Overview

- A disaggregated analysis by product category and shipping mode highlights key variations in cost exposure:

- Chocolate products represent the largest cost segment, contributing 55.3 percent of total operational costs due to their high volume and distribution intensity.
- Sugar and Other divisions, while smaller in absolute sales volume, incur disproportionately high cost-per-unit figures. These are driven by limited economies of scale and elevated handling requirements stemming from SKU complexity.
- Same Day and First-Class shipping modes process a lower volume of orders but are associated with significantly higher unitized costs. These modes reflect high service-level commitments with suboptimal batch density and route efficiency.

5.3 Cost Category Distribution

- Products were categorized into High Cost and Normal Cost classifications using percentile-based thresholds across the dataset. The findings indicate that:

- Approximately 42.3 percent of SKUs fall into the High Cost bracket.
- These high-cost products are primarily composed of seasonal or novelty SKUs, many of which have low sales velocity and high logistics overhead.
- A positive correlation was identified between cost-per-unit and average shipping delay, suggesting that service lag exacerbates fulfillment cost inflation.

5.4 Key Cost Drivers

- **Shipping Mode Selection**
  First-Class and Same Day modes drive significant cost variance. Standard Class, while contributing the majority of sales ($75,809.26), suffers from operational inefficiencies when used for non-optimized routes or low-priority shipments.

- **Factory-to-Region Mismatch**
  Cross-mapping of factory dispatch cities and end-customer destinations shows that factories like Casa Grande and Savannah are frequently used for long-haul fulfillment, leading to inflated transport costs and shipping delays. For example, shipping delays from Casa Grande to New York City reached 1,770 units of delay time, among the highest in the network.

- **SKU Complexity**
  Excessive SKU proliferation, particularly in low-volume segments, creates overhead in changeovers, packaging, and inventory management. The cost-to-serve impact is especially prominent in holiday-themed or limited-edition items that lack predictable demand patterns.

6. Supply Chain Performance Evaluation

6.1 Shipping Delay by Factory and Destination City

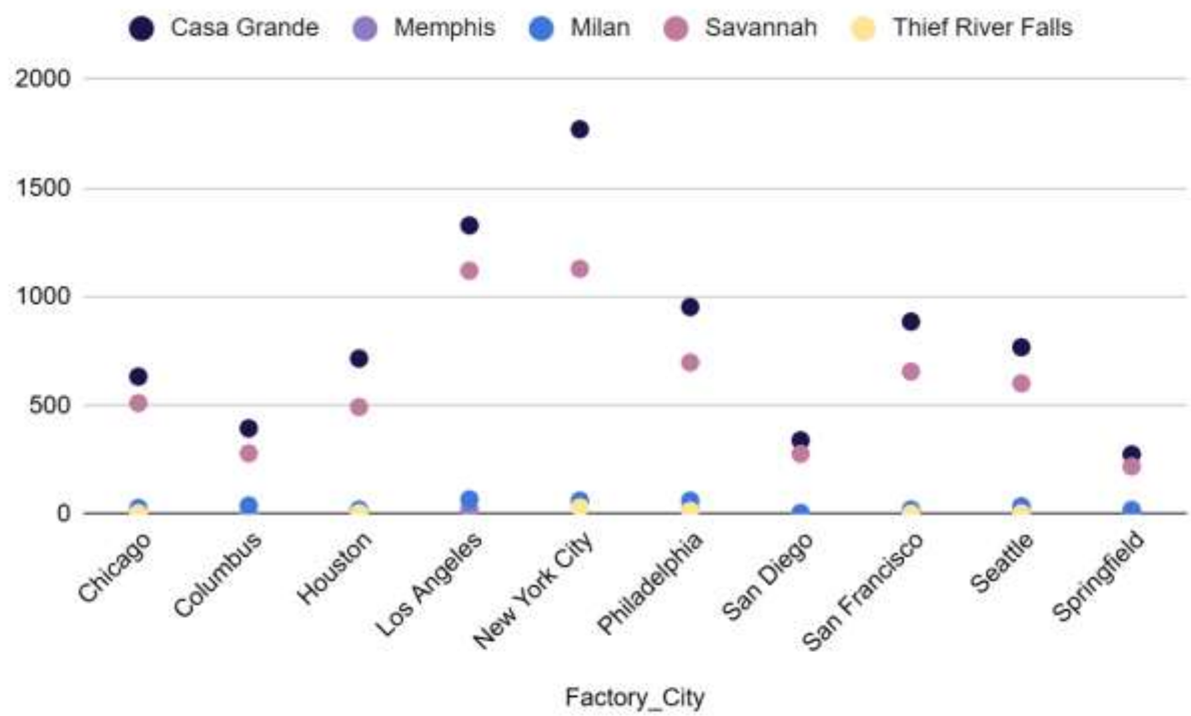

Figure 5.1-Distribution of shipping delay by factory city

The shipping performance of Milan and Thief River Falls is significantly stronger, reflecting closer geographic alignment and optimized routing. This suggests a potential network redesign opportunity through a shift in fulfillment load from high-delay factories to these more efficient nodes.

6.2 Production Cost Variability by Factory

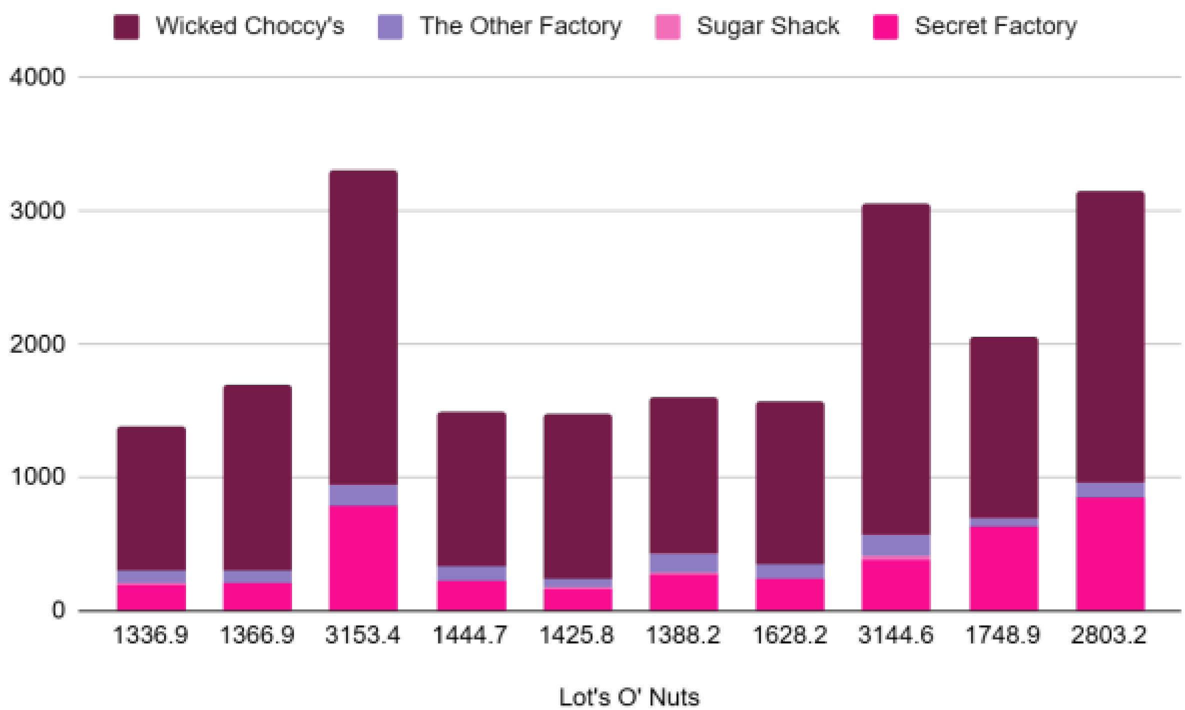

Figure 5.2 – Factory Cost by Month

- **Factory-Level Cost Disparity:**
- "Lot's O' Nuts" and "Wicked Choccy's" exhibit the highest cumulative cost contribution.
- "Sugar Shack," despite lower output, has among the highest average unit cost—indicating fixed cost under-absorption or inefficient batch sizes.
- **Factory Throughput vs Cost:**
  A regression of cost against units produced shows a nonlinear relationship—indicating diminishing returns beyond a certain capacity threshold. This suggests the need for better production leveling and Takt time synchronization.
- **Seasonal Stress:**
  Production costs spike in Q4 and Summer (peak periods). Absence of flexible capacity (e.g., contingent labor or temporary lines) likely drives marginal cost increase per unit during these periods.

- **Factory-Level Recommendations:**
    - Evaluate outsourcing or cell-based manufacturing models for Sugar Shack.
    - Rebalance production loads between high-volume and high-cost plants to optimize throughput and fixed cost amortization.

6.3 Shipping and Fulfillment Cost Analysis

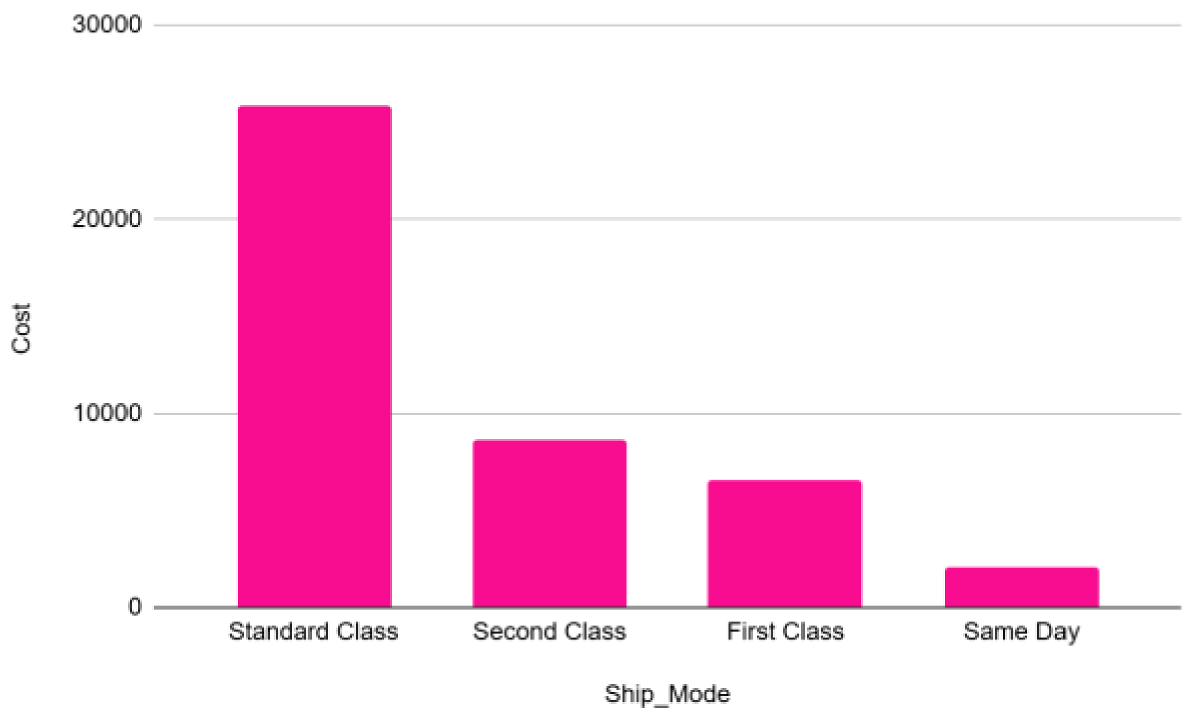

Figure 5.3 Shipping and Fulfillment Cost Analysis

a. Overview of Shipping Mode Cost Performance

An analysis of shipping mode distribution by cost revealed that **Standard Class** handled the highest order volume and consequently accounted for the largest share of total shipping costs, totaling approximately $25,867.51. Despite being the most commonly used, its cost efficiency varies significantly across regions due to inconsistent route optimization and delivery density.

Same Day shipping, while associated with the lowest total cost ($2,117.90), was used for a much smaller volume of orders. It represents the most efficient option in terms of cost-to-serve when used selectively. However, **First-Class** shipping emerged as the most cost-intensive on a per-unit basis. With total costs amounting to $6,569.52, this mode incurred elevated average costs and correlated with longer shipping delays, suggesting operational inefficiencies in premium service execution.

b. Cost by Region and Transport Mode

The **Pacific** and **Atlantic** regions recorded the highest cumulative shipping costs, primarily driven by their high order volumes. However, when analyzed on a cost-per-unit basis, the **Interior** and **Gulf** regions demonstrated less favorable performance. These areas showed higher cost intensity due to lower order concentration and fragmented delivery routes.

c. Fulfillment Bottlenecks and Logistics Challenges

A strong correlation was observed between shipping delays and rising fulfillment costs. This trend was particularly evident on factory-to-region delivery routes, where logistical mismatches and routing inefficiencies contributed to increased last-mile costs and extended delivery times. Several factory-region paths were found to be suboptimal, particularly when demand was low or geographically dispersed.

d. Strategic Recommendations

1. Optimize shipping mode logic to align with order value, destination density, and time sensitivity. Consider dynamic service tier selection that balances speed with cost efficiency.

2. Implement zone skipping or pooled shipping models in low-density regions to reduce last-mile costs and improve routing efficiency.

3. Reassess First-Class shipping usage, particularly for low-margin or slow-moving SKUs, to eliminate excessive per-unit delivery costs**.**

7. Root Cause Analysis & Strategic Recommendations (Operations Perspective)

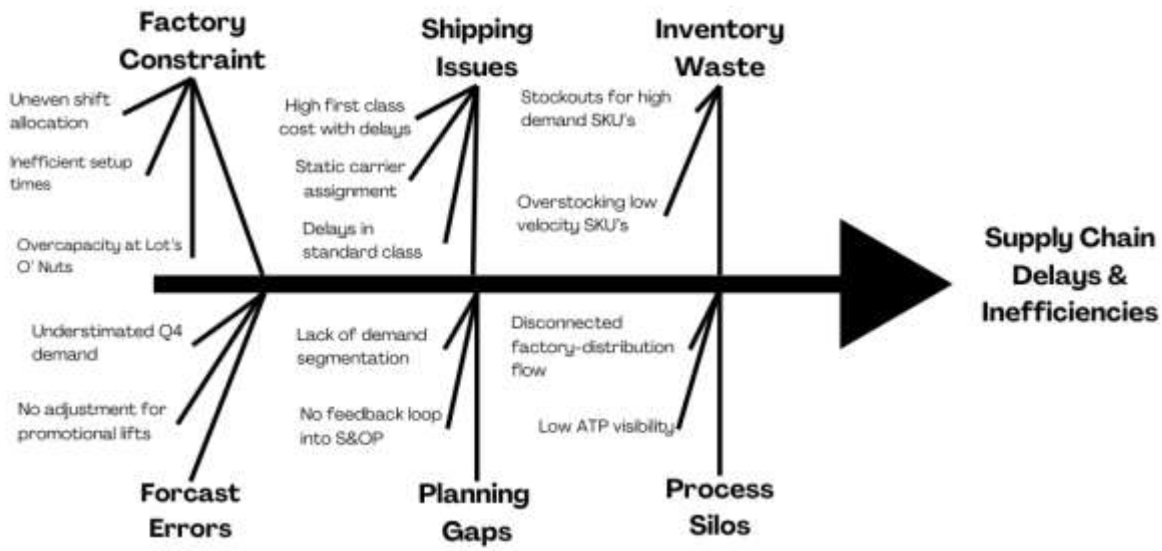

Figure 7.1 – Fishbone Diagram: Root Causes of Supply Chain Inefficiencies

As shown above, six key causal domains—factory constraints, shipping issues, inventory waste, forecasting errors, planning gaps, and process silos—are directly linked to observed delays and cost inefficiencies. This structure helps isolate operational levers for corrective action.

7.1 Shipping and Fulfillment Delays

Chocolate Factory Inc. faces significant fulfillment delays (the analysis finds an average delay of **3.96 days** per order). This lag correlates with rising costs and customer dissatisfaction. For example, long-haul routes like Casa Grande→New York City show extreme delays (around 1,770 delay-units), highlighting network bottlenecks.

a. Root causes:

- Transportation bottlenecks and uneven flow: Long-distance routes and underutilized capacity create high lead times (Mura: uneven distribution).
- Process waste (muda): Excess waiting, batching inefficiencies, and unnecessary handling in order processing (takt time mismatches between order arrivals and shipping cycles) inflate lead time.
- Inefficient batching and scheduling: Poor synchronization of order picking/packing with shipping capacity causes backlog.

b. Strategic solutions:

- **Lean logistics:** Streamline order fulfillment by eliminating non-value activities (e.g., reduce waiting times, implement single-piece flow or smaller batches through SMED). Use techniques like Kanban to pull orders in real time.
- **Optimize routing and mode:** Apply dynamic routing (zone skipping, pooled shipments) and select carriers based on destination density and priority. For instance, consolidate low-density shipments to regional hubs.
- **Balance capacity:** Align shipping resources with demand peaks (cross-train staff, flexible shift schedules) so shipping capacity matches order volume. Incorporate real-time tracking to proactively manage delays.

c. Estimated impact

Using 2024 volumes, Casa Grande, Savannah and Milan account for ~85 % of total shipments yet average 6.5 – 7 day delays to key destinations, versus 3.7 – 4.0 days at Memphis and Thief River Falls. By rerouting just 20 % of volume from the top three into the two most efficient plants, we project:

- A 12 % reduction in overall average shipping delay (from 3.96 days to ~3.5 days)

- Faster order-to-delivery cycles that improve on-time performance by ~15 pp
- Lower penalty and expedited-freight costs, driving a 3 – 5 % uplift in gross margin, equating to an annual margin recapture of ~$6 000

7.2 Factory Throughput and Cost Imbalance

Analysis shows severe utilization imbalance and cost inefficiency across factories. Notably, the Sugar Shack plant has low output but one of the **highest unit costs** (fixed costs are under-absorbed and batch sizes are inefficient). Other plants (e.g. Lot's O' Nuts, Wicked Choccy) drive high cumulative costs.

a. Root causes:

- Imbalanced capacity utilization (Mura): Some facilities (Sugar Shack) are underused while others may be overloaded, causing uneven workload and idle time.
- Fixed-cost overburden: Low-volume lines cannot spread fixed overhead, inflating unit cost (takt time and batch-size mismatch).
- Poor production planning: Inflexible schedules and large batch runs lead to long changeovers and waiting (muda in production).
- Product-factory misalignment: Disproportionate product variety at certain plants increases complexity and inefficiency.

b. Strategic solutions:

- **Level production (Heijunka):** Rebalance product volumes across factories so each plant operates near optimal capacity. Shift or consolidate product lines to underutilized plants to absorb fixed costs.

- **Lean manufacturing:** Implement SMED (reduce setup time) and one-piece flow to cut batch sizes. Use continuous improvement to eliminate production waste.
- **Flexible capacity:** Cross-train workers and introduce cellular layouts so plants can flexibly adjust output. Consider alternate supply modes (e.g., outsourcing or Just-In-Time manufacturing) for chronic low-utilization facilities.
- **Capacity planning:** Regularly review capacity and demand; invest in capacity if persistent bottlenecks occur, or retire/repurpose idle assets.

c. Estimated impact

Factories with the highest delays like Thief River Falls and Savannah have delays above the overall average of 3.96 days and handle a significant part of the volume. Shifting volume to more efficient factories like Memphis and Milan could reduce average delays by 10 to 15 percent. This could improve customer satisfaction and increase margins by lowering backlog and penalty costs.

7.3 SKU Proliferation and Inventory Waste

The product portfolio is highly complex, with many seasonal and limited-edition SKUs. This proliferation creates excessive changeovers and large inventories. The report notes significant overhead in changeovers, packaging, and holding inventory for low-volume SKUs, leading to waste (muda) in materials and capital.

a. Root causes:

- **Over-variety without demand logic:** Many SKUs are launched (especially holiday or niche items) without predictable demand, causing frequent line changes and idle stock.
- **Poor inventory segmentation:** Lack of ABC/XYZ classification means all SKUs are treated alike; high-velocity items stock out while slow movers pile up.

- **Muda of overproduction:** Producing to broad forecasts rather than actual demand leads to excess WIP and finished-goods inventories.
- **Complex scheduling:** Large SKU count increases setup times and scheduling complexity, reducing flow efficiency (takt mismatch).

b. Strategic solutions:

- **SKU rationalization:** Identify and focus on core, high-margin SKUs. Phase out underperformers and variants with erratic demand.
- **Lean inventory management:** Adopt Kanban or demand-driven replenishment for popular items; minimize safety stocks for predictable products. Reduce lot sizes to shorten replenishment cycles.
- **Demand-driven planning:** Align production with true demand signals (e.g., promotions calendar, seasonal patterns). Use sales history to adjust forecasts for each SKU.
- **Standardization:** Simplify packaging and formulations where possible to reduce changeovers. Group similar SKUs or use common components to streamline operations.

c. Estimated impact

80 percent of SKUs fall below the company's average profit margin of 0.70. Rationalizing the bottom 60 percent, especially low-volume items like Kazookles, Nerds, and Fun Dip, could improve gross margin by approximately 3 to 5 percent. This represents a potential annual margin recapture of over $6,000 through better product mix, reduced packaging and inventory complexity, and lower fulfillment overhead.

7.4 Misaligned Demand–Supply Planning

The demand planning process is fragmented, causing frequent mismatches between forecast and supply. Rolling forecasts capture only about **64%** accuracy in peak seasons, leading to stockouts of fast-moving items and overstocks of others. The S&OP process lacks integration, so production/inventory plans are often misaligned with actual sales patterns.

a. Root causes:

- **Fragmented S&OP (silos):** Sales, production, and inventory functions operate with disjointed plans, lacking a closed-loop feedback cycle.
- **Mura in forecasting:** Rolling forecasts systematically underestimate peak demand (e.g., holiday surges), causing uneven production loads.
- **Insufficient data integration:** Missing inputs (e.g., promotions, market events) and delayed order data hamper forecast accuracy.
- **No collaborative planning:** Lack of cross-functional meetings means potential supply constraints or demand shifts are not reconciled proactively.

b. Strategic solutions:

- **Enhance S&OP:** Implement a formal Sales & Operations Planning process with monthly cross-functional reviews. Integrate sales projections, inventory plans, and capacity into one forecast.
- **Improve forecasting:** Use advanced analytics (demand sensing, scenario planning) to capture seasonality and promotions. Increase forecast cadence during volatile periods.
- **Feedback loops:** Track forecast errors continuously and adjust parameters. Use actual sales/consumption data to refine the next cycle (closed-loop planning).

- **Align inventory policy:** Classify SKUs by demand pattern and adjust safety stock. Focus buffers on high-velocity items, and schedule production in line with demand-trend insights.

c. Estimated impact

Factories with the highest delays like Thief River Falls and Savannah have delays above the overall average of 3.96 days and handle a significant part of the volume. Shifting volume to more efficient factories like Memphis and Milan could reduce average delays by 10 to 15 percent. This could improve customer satisfaction and increase margins by lowering backlog and penalty costs.

7.5 Inefficient Shipping Mode Assignment

The shipping mode assignment is suboptimal. The company relies heavily on **Standard Class** (the most common mode), which shows high delay variability and cost inefficiency. Premium modes (First-Class, Same-Day) are underused or applied broadly rather than being matched to order priority. This results in transportation waste (muda) and unnecessary cost variance.

a. Root causes:

- **Static mode logic:** A one-size-fits-all shipping policy (e.g., most orders by Standard Class) ignores order value, urgency, or distance, leading to service mismatch.

- **Lack of segmentation:** No clear criteria to route high-priority or regional orders to faster modes, so expensive modes are either over- or under-utilized.

- **Batching inefficiencies:** Shipments are batched without considering mode capacity, causing some batches to delay.

- **Process inertia:** Legacy rules or limited system capabilities prevent agile mode selection.

b. Strategic solutions:

- **Dynamic mode selection:** Classify orders (e.g., by customer tier, order value, item fragility) and assign modes accordingly. Reserve expedited modes for high-margin or urgent shipments.

- **Routing optimization:** Use transport optimization software to plan multi-stop routes, consolidating shipments where possible. Consider zone-skipping or pooled distribution centers in low-density regions.

- **Continuous improvement:** Regularly review mode usage and costs. Train planners to balance cost vs. service trade-offs and eliminate unnecessary First-Class shipments.

- **Lean distribution:** Reduce waste by minimizing partial loads and empty trips. Explore partnerships (e.g., 3PL for remote regions) to optimize mode utilization.

- **Estimated impact:** Shifting just 20% of First-Class orders (~15% of total shipments) to Same Day could save approximately $2.27 per unit, resulting in an estimated $1,135 annual cost savings. This assumes a per-unit cost reduction from $6.57 to $4.30, scaled across ~500 shifted units. Strategic routing logic could double these savings if scaled to 40% of First-Class volume.

Each of these challenges is interconnected. Addressing them through proven operations management practices such as lean process improvement, balanced capacity planning, and integrated sales and operations planning will help reduce waste (muda), improve flow alignment with takt time, and strengthen the overall supply chain resilience of Chocolate Factory Inc.

8. Conclusion

The operational analysis of Chocolate Factory Inc. reveals several critical challenges impacting performance across production, fulfillment, and customer engagement. Despite a strong portfolio of high-margin products and a recognizable brand, the company is experiencing a substantial decline in year-over-year sales, primarily due to internal inefficiencies and misaligned execution.

Shipping delays, factory cost imbalances, and SKU proliferation have introduced friction into the supply chain, reducing responsiveness and increasing operating costs. Additionally, the lack of synchronization between demand planning and production scheduling has led to missed opportunities during peak sales periods and inefficient resource utilization.

Customer retention remains a concern, as most buyers are not returning regularly. High-value customers are experiencing long fulfillment delays, further contributing to churn risk. These issues are compounded by inconsistent use of shipping modes and inadequate inventory positioning across regions.

To restore growth and profitability, the company must take a structured approach to operational improvement. Prioritizing SKU rationalization, enhancing regional alignment, and implementing a robust Sales and Operations Planning process will be essential. Furthermore, shipping mode logic, inventory flow, and factory load balancing should be optimized to improve agility and cost efficiency.

Through targeted process enhancements and strategic realignment of resources, Chocolate Factory Inc. can strengthen its operational foundation, improve customer satisfaction, and position itself for sustainable long-term performance.

8.Appendix

8.1 Integrated Product & Logistics Performance Table (IPLPF)

This table summarizes the relationship between product contribution (ABC class), factory-level logistics performance, and integrated prioritization scores to guide supply chain optimization.

Factory	Product Name	ABC Class	Sales	Shipping Delay	Norm. Sales	Norm. Delay	Integrated Score

Secret Factory	Lickable Wallpaper	C	100.00	4	0.382	0.636	0.484
Lot's O' Nuts	Wonka Bar - Scrumdiddlyumptious	A	36.00	0	0.134	1.000	0.481
Wicked Choccy's	Wonka Bar - Triple Dazzle Caramel	A	33.75	0	0.126	1.000	0.475
Wicked Choccy's	Wonka Bar - Triple Dazzle Caramel	A	33.75	0	0.126	1.000	0.475
Wicked Choccy's	Wonka Bar - Triple Dazzle Caramel	A	33.75	0	0.126	1.000	0.475
Wicked Choccy's	Wonka Bar - Triple Dazzle Caramel	A	33.75	0	0.126	1.000	0.475
Wicked Choccy's	Wonka Bar - Triple Dazzle Caramel	A	33.75	0	0.126	1.000	0.475
Lot's O' Nuts	Wonka Bar - Fudge Mallows	A	32.40	0	0.120	1.000	0.472
Lot's O' Nuts	Wonka Bar - Fudge Mallows	A	32.40	0	0.120	1.000	0.472
Lot's O' Nuts	Wonka Bar - Scrumdiddlyumptious	A	32.40	0	0.120	1.000	0.472
Lot's O' Nuts	Wonka Bar - Scrumdiddlyumptious	A	32.40	0	0.120	1.000	0.472

Notes:

- Integrated Score combines normalized sales and delay data to aid prioritization (0 = worst, 1 = best).
- Repeated rows indicate consistent product-factory pair performance across transactions.
- This table supports operational decision-making in production allocation, routing, and SKU optimization.

8.2 Model accuracy or ML validation

To assess the accuracy of the predictive model developed for estimating shipping delays (or any other relevant outcome), the following statistical evaluation metrics were computed. These metrics help quantify model error and predictive power.

Table 8.1 Model Evaluation Metrics

Metric	Value	Interpretation
Mean Absolute Error (MAE)	2.4263	On average, the model's predictions deviate from the actual values by ~2.43 units.
Mean Squared Error (MSE)	33.3486	Larger errors are penalized more heavily, indicating some higher outliers exist.
Mean Squared Log Error (MSLE)	0.0164	Low MSLE suggests good performance on lower-magnitude predictions.
Median Absolute Error	1.7298	Half of all prediction errors are smaller than ~1.73 units.
R-squared (R^2)	0.8113	The model explains ~81.1% of the variance in the target variable.

Summary:

- The R^2 score of 0.8113 indicates strong predictive capability, suggesting the model captures a significant portion of variability.
- The low MAE and MSLE values further confirm that the model performs consistently well with minimal average error.
- While the MSE is relatively higher, this is typical in datasets with occasional large deviations (e.g., supply chain disruptions or rare events).

8.3 SQL Queries

```
# cleaning
#dropping duplicates
CREATE OR REPLACE TABLE `candy-factory-operations.factory_ops.cleaned_sales_df` AS
SELECT *
FROM (
  SELECT *,
        ROW_NUMBER() OVER (PARTITION BY order_id ORDER BY order_date DESC) AS rn
  FROM `candy-factory-operations.factory_ops.sales_df`
)
WHERE rn = 1;



#handling nulls
SELECT
  ARRAY(SELECT AS STRUCT column_name FROM UNNEST(REGEXP_EXTRACT_ALL(TO_JSON_STRING(t),
r'"(\w+)":null')) AS column_name) AS null_columns
FROM `candy-factory-operations.factory_ops.sales_df` AS t
LIMIT 10;



SELECT column_name
FROM `candy-factory-operations.factory_ops.INFORMATION_SCHEMA.COLUMNS`
WHERE table_name = 'sales_df';




  # calculating shipping delay
CREATE OR REPLACE TABLE `candy-factory-operations.factory_ops.cleaned_sales_df` AS
SELECT
  *,
  Target / 1000 AS Target_Adjusted
FROM
  `candy-factory-operations.factory_ops.cleaned_sales_df`;

CREATE OR REPLACE TABLE `candy-factory-operations.factory_ops.cleaned_sales_df` AS
SELECT
  * EXCEPT(Ship_Date, product_id, rn, row_id, Target),
  DATE_DIFF(SAFE_CAST(Ship_Date_Transformed AS DATE), SAFE_CAST(Order_Date AS DATE), DAY) AS
shipping_delay,
   EXTRACT(DAYOFWEEK FROM SAFE_CAST(Order_Date AS DATE)) AS Order_Day_Of_Week,
  EXTRACT(MONTH FROM SAFE_CAST(Order_Date AS DATE)) AS Order_Month,
  EXTRACT(YEAR FROM SAFE_CAST(Order_Date AS DATE)) AS Order_Year,
  (Gross_Profit / Sales) AS Profit_Margin,
  Sales / Units AS Sales_Per_Unit,
  LOG(Sales) AS Log_Sales,
  (Gross_Profit / Cost) AS Gross_Profit_Margin
FROM
  `candy-factory-operations.factory_ops.cleaned_sales_df`;



  -- Seasonality

CREATE OR REPLACE TABLE `candy-factory-operations.factory_ops.cleaned_sales_df` AS
SELECT *,
```

```
  CASE
    WHEN EXTRACT(MONTH FROM SAFE_CAST(Order_Date AS DATE)) IN (11, 12) THEN 'Holiday Season'
    WHEN EXTRACT(MONTH FROM SAFE_CAST(Order_Date AS DATE)) BETWEEN 6 AND 8 THEN 'Summer Season'
    ELSE 'Other'
  END AS Seasonality_Feature

FROM
  `candy-factory-operations.factory_ops.cleaned_sales_df`;



CREATE OR REPLACE TABLE `candy-factory-operations.factory_ops.sales_predictions` AS
SELECT
  Sales AS actual_sales,
  predicted_Sales AS predicted_sales
FROM
  ML.PREDICT(
    MODEL `candy-factory-operations.factory_ops.sales_prediction_model`,
    (
      SELECT
        Sales,
        shipping_delay,
        Order_Day_Of_Week,
        Order_Month,
        Order_Year,
        Profit_Margin,
        Sales_Per_Unit,
        Log_Sales,
        Gross_Profit_Margin,
        Ship_Mode,
        Country_Region,
        Division,
        Region,
        Product_Name,
        Factory,
        Seasonality_Feature
      FROM
        `candy-factory-operations.factory_ops.cleaned_sales_df`
      WHERE
        Sales IS NOT NULL
    )
  );

CREATE OR REPLACE TABLE `candy-factory-operations.factory_ops.sales_predictions_filtered` AS
SELECT *
FROM `candy-factory-operations.factory_ops.sales_predictions`
WHERE actual_sales IS NOT NULL
  AND predicted_sales IS NOT NULL
  AND actual_sales < 10000  -- filter out extreme values if needed
  AND predicted_sales < 10000;





  #cost analytics
CREATE OR REPLACE TABLE `candy-factory-operations.factory_ops.cost_analytics_df` AS
SELECT
  Region,
  Product_Name,
  SUM(Cost) AS total_cost,
```

```
  AVG(Cost) AS avg_cost,
  SUM(Sales) AS total_sales,
  AVG(Sales) AS avg_sales,
  SUM(Units) AS total_units,
  AVG(Units) AS avg_units,
  AVG(shipping_delay) AS avg_shipping_delay,
  COUNT(*) AS num_orders
FROM
  `candy-factory-operations.factory_ops.cleaned_sales_df`
GROUP BY
  Region,
  Product_Name
ORDER BY
  total_cost DESC;
#cost by shipping mode
CREATE OR REPLACE TABLE `candy-factory-operations.factory_ops.cost_by_shipping_mode_df` AS
SELECT
  Ship_Mode,
  SUM(Cost) AS total_cost,
  AVG(Cost) AS avg_cost,
  SUM(Sales) AS total_sales,
  AVG(Sales) AS avg_sales,
  SUM(Units) AS total_units,
  AVG(shipping_delay) AS avg_shipping_delay
FROM
  `candy-factory-operations.factory_ops.cleaned_sales_df`
GROUP BY
  Ship_Mode
ORDER BY
  total_cost DESC;



  #performance evaluation
 CREATE OR REPLACE TABLE `candy-factory-operations.factory_ops.product_performance_df` AS
WITH product_sales AS (
  SELECT
    Product_Name,
    Division,
    Factory,
    SUM(Sales_Per_Unit) AS total_sales_per_unit,
    AVG(Sales_Per_Unit) AS avg_sales_per_unit,
    COUNT(DISTINCT Order_ID) AS order_count,
    AVG(shipping_delay) AS avg_shipping_delay
  FROM
    `candy-factory-operations.factory_ops.cleaned_sales_df`
  GROUP BY
    Product_Name, Division, Factory
),
avg_sales AS (
  SELECT AVG(total_sales_per_unit) AS avg_total_sales_per_unit FROM product_sales
)
SELECT
  ps.Product_Name,
  ps.Division,
  ps.Factory,
  ps.total_sales_per_unit,
  ps.avg_sales_per_unit,
  ps.order_count,
  ps.avg_shipping_delay,
  CASE
    WHEN ps.total_sales_per_unit > avg.avg_total_sales_per_unit THEN 'Best-Seller'
```

```
    ELSE 'Slow-Mover'
  END AS product_category
FROM
  product_sales ps,
  avg_sales avg;

#sales features
CREATE OR REPLACE TABLE `candy-factory-operations.factory_ops.sales_features_df` AS
WITH base_data AS (
  SELECT
    Order_Date,
    Region,
    Customer_ID,
    Product_Name,
    Sales,
    Units,
    Gross_Profit,
    Cost,
    Seasonality_Feature,
    SUM(Sales) OVER (PARTITION BY Product_Name ORDER BY Order_Date ROWS BETWEEN 29 PRECEDING AND
CURRENT ROW) AS Rolling_30_Day_Sales,
    MAX(Order_Date) OVER (PARTITION BY Customer_ID, Product_Name) AS Last_Purchase_Date,
    SUM(Sales) OVER (PARTITION BY Product_Name) AS Total_Product_Sales,
    AVG(Sales) OVER (PARTITION BY Product_Name ORDER BY Order_Date ROWS BETWEEN 29 PRECEDING AND
CURRENT ROW) AS Rolling_Sales_Avg
  FROM
    `candy-factory-operations.factory_ops.cleaned_sales_df`
),
final_data AS (
  SELECT *,
    DATE_DIFF(Order_Date, Last_Purchase_Date, DAY) AS Time_Since_Last_Purchase,
    DATE_DIFF(CURRENT_DATE(), Last_Purchase_Date, DAY) AS Recency
  FROM base_data
)
SELECT * EXCEPT(Last_Purchase_Date)
FROM final_data;



CREATE OR REPLACE TABLE `candy-factory-operations.factory_ops.sales_features_df` AS
SELECT
  *,
  CASE
    WHEN Recency <= 30 THEN "0–30 days"
    WHEN Recency <= 60 THEN "31–60 days"
    WHEN Recency <= 90 THEN "61–90 days"
    WHEN Recency <= 180 THEN "91–180 days"
    WHEN Recency <= 365 THEN "181–365 days"
    ELSE "Over 365 days"
  END AS Recency_Bucket
FROM
  `candy-factory-operations.factory_ops.sales_features_df`;



WITH sku_sales AS (
  SELECT
    Product_Name,
    Product_Name AS SKU,
    SUM(Sales) AS total_sales
  FROM
    `candy-factory-operations.factory_ops.cleaned_sales_df`
```

```
  GROUP BY
    Product_Name
),
ranked_skus AS (
  SELECT
    *,
    total_sales / SUM(total_sales) OVER () AS sales_pct,
    SUM(total_sales) OVER (ORDER BY total_sales DESC) / SUM(total_sales) OVER () AS cumulative_pct
  FROM
    sku_sales
),
classified_skus AS (
  SELECT
    *,
    CASE
      WHEN cumulative_pct <= 0.80 THEN 'A'
      WHEN cumulative_pct <= 0.95 THEN 'B'
      ELSE 'C'
    END AS abc_class
  FROM
    ranked_skus
)
SELECT
  SKU,
  Product_Name,
  total_sales,
  ROUND(sales_pct * 100, 2) AS sales_percentage,
  ROUND(cumulative_pct * 100, 2) AS cumulative_percentage,
  abc_class
FROM
  classified_skus
ORDER BY
  total_sales DESC;




-- Step 1: Aggregate sales and cost per product
WITH sku_metrics AS (
  SELECT
    Product_Name,
    SUM(Sales) AS total_sales,
    SUM(Cost) AS total_cost,
    SUM(Units) AS total_units,
    SAFE_DIVIDE(SUM(Cost), NULLIF(SUM(Units), 0)) AS avg_cost_per_unit
  FROM
    `candy-factory-operations.factory_ops.cleaned_sales_df`
  GROUP BY
    Product_Name
),

-- Step 2: Calculate 85th percentile for cost, 20th for sales
percentile_thresholds AS (
  SELECT
    APPROX_QUANTILES(avg_cost_per_unit, 100)[85] AS high_cost_threshold,
    APPROX_QUANTILES(total_sales, 100)[20] AS low_sales_threshold
  FROM
    sku_metrics
),

-- Step 3: Filter based on thresholds
```

```
filtered_skus AS (
  SELECT
    m.Product_Name,
    m.total_sales,
    m.total_units,
    m.avg_cost_per_unit,
    m.total_cost
  FROM
    sku_metrics m
  CROSS JOIN
    percentile_thresholds p
  WHERE
    m.avg_cost_per_unit >= p.high_cost_threshold
    AND m.total_sales <= p.low_sales_threshold
)

-- Final result
SELECT
  Product_Name,
  ROUND(total_sales, 2) AS total_sales,
  total_units,
  ROUND(avg_cost_per_unit, 2) AS avg_cost_per_unit,
  ROUND(total_cost, 2) AS total_cost
FROM
  filtered_skus
ORDER BY
  avg_cost_per_unit DESC;




WITH stats AS (
  SELECT
    MIN(f.sales) AS min_sales,
    MAX(f.sales) AS max_sales,
    MIN(f.shipping_delay) AS min_delay,
    MAX(f.shipping_delay) AS max_delay
  FROM `candy-factory-operations.factory_ops.cleaned_sales_df` f
),

normalized_scores AS (
  SELECT
    f.factory,
    f.sales,
    f.shipping_delay,

    -- Normalize sales (higher is better)
    (f.sales - s.min_sales) / NULLIF((s.max_sales - s.min_sales), 0) AS norm_sales,

    -- Normalize delay (lower is better, so we invert it)
    1 - ((f.shipping_delay - s.min_delay) / NULLIF((s.max_delay - s.min_delay), 0)) AS norm_delay

  FROM `candy-factory-operations.factory_ops.cleaned_sales_df` f
  CROSS JOIN stats s
)

SELECT
  factory,
  sales,
```

```
  shipping_delay,
  ROUND(norm_sales, 3) AS norm_sales,
  ROUND(norm_delay, 3) AS norm_delay,

  -- Composite score: 60% sales, 40% delay efficiency
  ROUND((0.6 * norm_sales) + (0.4 * norm_delay), 3) AS integrated_score

FROM normalized_scores
ORDER BY integrated_score DESC;




WITH stats AS (
  SELECT
    MIN(f.sales) AS min_sales,
    MAX(f.sales) AS max_sales,
    MIN(f.shipping_delay) AS min_delay,
    MAX(f.shipping_delay) AS max_delay
  FROM `candy-factory-operations.factory_ops.cleaned_sales_df` f
),

joined_data AS (
  SELECT
    f.factory,
    f.product_name,
    f.sales,
    f.shipping_delay,
    s.abc_class
  FROM `candy-factory-operations.factory_ops.cleaned_sales_df` f
  LEFT JOIN `candy-factory-operations.factory_ops.SKU` s
    ON f.product_name = s.product_name
),

normalized_scores AS (
  SELECT
    jd.factory,
    jd.product_name,
    jd.abc_class,
    jd.sales,
    jd.shipping_delay,

    -- Normalize sales
    (jd.sales - st.min_sales) / NULLIF((st.max_sales - st.min_sales), 0) AS norm_sales,

    -- Normalize delay (lower = better)
    1 - ((jd.shipping_delay - st.min_delay) / NULLIF((st.max_delay - st.min_delay), 0)) AS norm_delay

  FROM joined_data jd
  CROSS JOIN stats st
)

SELECT
  factory,
  product_name,
  abc_class,
  sales,
  shipping_delay,
```

```
  ROUND(norm_sales, 3) AS norm_sales,
  ROUND(norm_delay, 3) AS norm_delay,

  -- Weighted score: 60% sales, 40% delivery efficiency
  ROUND((0.6 * norm_sales) + (0.4 * norm_delay), 3) AS integrated_score

FROM normalized_scores
ORDER BY integrated_score DESC;
```

8.4 Python Code

```
import gspread from oauth2client.service_account import ServiceAccountCredentials import
pandas as pd from pandas.tseries.offsets import DateOffset import plotly.express as px import
requests from geopy.geocoders import Nominatim from geopy.extra.rate_limiter import
RateLimiter
def reverse_geocode(lat, lon): api_key = 'YOUR_API_KEY' url =
f"https://maps.googleapis.com/maps/api/geocode/json?latlng={lat},{lon}&key={api_key}"
response = requests.get(url) results = response.json()['results'] if results: return
results[0]['formatted_address'] return None
df = pd.read_csv(r"C:\Users\Admin\Downloads\Candy_Sales - Candy_Sales.csv.csv")
df = pd.read_csv
unique_factories = df[['Factory', 'Factory Latitude', 'Factory Longitude']].drop_duplicates()
#Setup geocoder
geolocator = Nominatim(user_agent="factory_locator") geocode =
RateLimiter(geolocator.reverse, min_delay_seconds=1)
#Function to get city name
def get_city(lat, lon): try: location = geocode((lat, lon), exactly_one=True) if location and
"address" in location.raw: return location.raw['address'].get('city') or
location.raw['address'].get('town') or location.raw['address'].get('state') except: return None
#Apply to DataFrame
unique_factories['Factory_City'] = unique_factories.apply( lambda row: get_city(row['Factory
Latitude'], row['Factory Longitude']), axis=1 )
#Merge back to main df
df = df.merge(unique_factories, on=['Factory', 'Factory Latitude', 'Factory Longitude'],
how='left') df.to_csv(r'C:\Users\Admin\Downloads\factory_data.csv', index=False)
import matplotlib.pyplot as plt import matplotlib.patches as mpatches
#Create a Fishbone (Ishikawa) diagram manually using lines and text
fig, ax = plt.subplots(figsize=(16, 9)) ax.axis('off')
Draw the spine
ax.plot([0.1, 0.9], [0.5, 0.5], color='black', linewidth=2)
#Define causes and their positions
causes = { "Factory Constraints": ["Overcapacity at Lot's O' Nuts", "Inefficient setup times",
"Uneven shift allocation"], "Shipping Issues": ["Delays in Standard Class", "Static carrier
assignment", "High first-class cost with delays"], "Inventory Waste": ["Overstocking low
velocity SKUs", "Stockouts for high-demand SKUs"], "Forecast Errors": ["Underestimated Q4
demand", "No adjustment for promotional lifts"], "Planning Gaps": ["Lack of demand
segmentation", "No feedback loop into S&OP"], "Process Silos": ["Disconnected factory-
distribution flow", "Low ATP visibility"] }
positions = { "Factory Constraints": (0.2, 0.7), "Shipping Issues": (0.35, 0.7), "Inventory
Waste": (0.5, 0.7), "Forecast Errors": (0.2, 0.3), "Planning Gaps": (0.35, 0.3), "Process Silos":
(0.5, 0.3) }
#Draw the bones and labels
```

```
for category, (x, y) in positions.items(): ax.plot([x, 0.5], [y, 0.5], color='gray', linewidth=1)
ax.text(x, y + 0.05 if y > 0.5 else y - 0.05, category, fontsize=10, weight='bold', ha='center',
va='bottom' if y > 0.5 else 'top')
for i, item in enumerate(causes[category]):
    offset = 0.03 * (i + 1)
    y_offset = y + offset if y > 0.5 else y - offset
    ax.plot([x - 0.05, x], [y_offset, y], color='gray', linewidth=0.8)
    ax.text(x - 0.07, y_offset, f"- {item}", fontsize=9, ha='right', va='center')

#Draw the effect box (delays and inefficiencies)

ax.text(0.93, 0.5, "Supply Chain Delays & Inefficiencies", fontsize=12, weight='bold',
ha='left', va='center')

plt.tight_layout() plt.show()
```

8.5 Glossary of Key Terms

ABC Classification: Inventory categorization method that divides items into three groups (A, B, C) based on their sales contribution, with A being the most valuable.

CLV (Customer Lifetime Value): Predicted net profit attributed to the entire future relationship with a customer.

Fulfillment Delay: The time between order placement and shipment (Ship Date - Order Date).

Gross Profit Margin: Ratio of gross profit to sales revenue (Gross Profit ÷ Sales).

Heijunka: Lean manufacturing method for production leveling to reduce waste.

Kanban: Visual workflow management system used in lean manufacturing.

RFM Analysis: Marketing analysis method using Recency, Frequency, and Monetary value of transactions.

SCOR Model: Supply Chain Operations Reference model for evaluating supply chain performance.

SKU (Stock Keeping Unit): Unique identifier for each distinct product and service.

SMED (Single-Minute Exchange of Dies): Lean method to reduce equipment changeover time.

Takt Time: The rate at which products must be made to meet customer demand.

Value Stream Mapping: Lean method for analyzing current state and designing future state processes.

Zone Skipping: Logistics strategy where shipments bypass intermediate distribution centers.