AdventureWorks Manufacturing & Profitability Analysis

1. Project Overview / Executive Summary

Project Title: AdventureWorks Manufacturing Efficiency & Profitability Analysis

Date: October 2023

Role: Data Analyst

Summary: This project is an end-to-end data analytics study focused on AdventureWorks, a global bicycle manufacturer. The primary objective was to transform raw production data into actionable business intelligence to assist the manufacturing management team in optimizing production schedules, improving profit margins, and balancing inventory levels.

The project simulates a full-cycle analytics workflow: extracting raw CSV data, designing a relational database schema in PostgreSQL, performing ETL (Extract, Transform, Load) processes using SQL to clean and model the data, and finally building an interactive dashboard in Microsoft Power BI.

Tech Stack:

- **Database Engine:** PostgreSQL 16
- **Database Management:** pgAdmin 4
- **Data Transformation (ETL):** SQL (Structured Query Language)
- **Visualization:** Microsoft Power BI Desktop
- **Source Data:** Kaggle / CSV Export

2. Business Problem and Objectives

Business Scenario: The Manufacturing Manager at AdventureWorks has identified potential inefficiencies in the production line and concerns regarding product profitability. The company produces a wide variety of SKUs, but visibility into which specific products are driving profit versus consuming excessive resources is limited.

Problem Statement: "Lack of granular visibility into production bottlenecks and item-level profitability is preventing AdventureWorks from optimizing its manufacturing schedules and inventory capital, potentially leading to lower overall margins."

Stakeholders:

- Manufacturing Manager
- Production Planner

- Financial Analyst

Objectives:

1. **Identify Profit Drivers:** Determine which product categories (Road, Mountain, Touring) yield the highest profit margins.
2. **Pinpoint Bottlenecks:** Identify specific items that require the longest manufacturing time to isolate production delays.
3. **Optimize Inventory:** Detect overstocked items that have low profit margins or low sales velocity to reduce carrying costs.
4. **Assess "Make vs. Buy":** Analyze the ratio of manufactured components versus purchased components to inform sourcing strategies.

Success KPIs:

- Average Profit Margin (%)
- Average Days to Manufacture (Lead Time)
- Inventory Value vs. Sales Status

3. Data Collection

Data Source: The dataset is a subset of the AdventureWorks sample database, specifically focusing on the Production.Product table. The data was acquired via Kaggle in CSV format (AdventureWorks2019Export.csv).

Extraction Process:

1. **Acquisition:** Downloaded raw CSV data from the external source.
2. **Database Initialization:** Created a local PostgreSQL database named adventureworks.
3. **Staging Table Design:** Created a raw_products table in SQL. Crucially, all columns were defined as VARCHAR to prevent data type errors during the initial bulk load.
4. **Ingestion:** Imported the data using the pgAdmin Import/Export Tool.
    - *Settings:* Format: CSV, Header: Yes, Delimiter: Comma.

Data Volume:

- **Rows:** 504 records
- **Columns:** 25 raw attributes
- **Limitations:** The dataset is a snapshot in time and does not include real-time sales transaction data, limiting the analysis to unit economics rather than total revenue volume.

4. Data Description

The raw dataset contains detailed product information. Below is the description of key columns before cleaning.

Column Name	Raw Data Type	Description

ProductID	Integer	Unique identifier for each product.
Name	Text	The commercial name of the product.
ProductNumber	Text	Internal SKU code.
MakeFlag	Integer	0 = Purchased, -1 = Manufactured.
FinishedGoodsFlag	Integer	0 = Not Sellable (Internal use), -1 = Sellable.
Color	Text	Product color (contains significant NULL values).
SafetyStockLevel	Integer	Minimum inventory quantity required.
StandardCost	Text	Manufacturing/Acquisition cost (formatted with '$').
ListPrice	Text	Selling price (formatted with '$').
DaysToManufacture	Integer	Time required to produce the item in days.
ProductLine	Text	Category Code: R = Road, M = Mountain, T = Touring, S = Standard.

5. Data Cleaning and Preprocessing

The raw data required significant transformation to be analytics-ready. All cleaning was performed using SQL scripts.

Step 1: Handling Missing Values

- **Action:** Used COALESCE(Color, 'N/A').
- **Reason:** The Color column contained NULL values for non-colored components (e.g., bolts, nuts). Leaving them as NULL would cause "blank" categories in Power BI

visualizations, confusing users.

Step 2: Decoding Binary Flags

- **Action:** Converted MakeFlag (-1/0) to 'Manufactured'/'Purchased' and FinishedGoodsFlag (-1/0) to 'Sellable'/'Not Sellable' using CASE statements.
- **Reason:** Numeric flags are not intuitive for business stakeholders. Decoding them into descriptive text ensures the dashboard is self-explanatory.

Step 3: Data Type Conversion & Formatting

- **Action:** Removed '$' and ',' symbols from StandardCost and ListPrice using REPLACE() and cast them as NUMERIC(10,2).
- **Reason:** The raw CSV contained currency symbols, causing the database to interpret prices as text strings. Power BI cannot perform aggregations (Sum, Average) on text fields.

Step 4: Category Standardization

- **Action:** Mapped single-letter codes ('R', 'M', 'T') to full names ('Road', 'Mountain', 'Touring').
- **Reason:** Improves readability and professionalism in final reports.

6. Exploratory Data Analysis (EDA)

Before visualization, SQL queries were used to explore patterns and distributions.

Key Analyses & Findings:

1. **Cost Structure:** A significant portion of the inventory (approx. 40%) had a StandardCost of $0.00. Investigation confirmed these were minor tracking components, not errors.
2. **Lead Time Distribution:**
   - Purchased items consistently showed 0 days to manufacture.
   - Manufactured items ranged from 1 to 4 days.
   - *Anomaly:* High-end Road Frames were outliers, taking the full 4 days.
3. **Profitability Gaps:** "Components" often showed $0.00 profit margin because they had no List Price (used internally). These were identified as non-revenue generating and needed to be filtered out for margin analysis.

7. Feature Engineering

New metrics were derived directly within the SQL ETL process to enhance analysis.

1. Profit Margin ($)

- **Formula:** List_Price - Standard_Cost
- **Business Logic:** This is the primary KPI. Calculating it at the row level in the database ensures consistency across all BI tools and prevents aggregation errors.

2. Product Category (Full Name)

- **Formula:** CASE WHEN ProductLine = 'R' THEN 'Road' ... END
- **Business Logic:** Bucketing products into broad categories (Components vs. Bikes) allows for high-level executive summaries.

8. Data Modeling (SQL)

The project utilized a **Flat Schema (Denormalized)** approach for simplicity and performance, transforming the raw staging table into a clean dimension table.

Architecture:

- **Staging Layer:** raw_products (Stores raw text data from CSV).
- **Production Layer:** dim_products (Clean, typed, and enriched data).

SQL Transformation Logic:

```
CREATE TABLE dim_products AS
SELECT
   ProductID,
   Name,
   -- Decoding logic
   CASE WHEN MakeFlag = -1 THEN 'Manufactured' ELSE 'Purchased' END AS
Make_Buy_Indicator,
   -- Cleaning logic
   CAST(REPLACE(REPLACE(ListPrice, '$', ''), ',', '') AS NUMERIC(10,2)) AS List_Price,
   -- Feature Engineering
   (List_Price - Standard_Cost) AS Profit_Margin
FROM raw_products;
```

9. Insights and Recommendations

Based on the analysis, the following actionable insights were generated:

Insight 1: Mountain Bikes are the Profit Leaders

- **Evidence:** The Mountain bike category yields a 12% higher average profit margin compared to the Road category.
- **Recommendation:** Prioritize marketing spend and inventory allocation toward the Mountain category for Q4 to maximize returns.

Insight 2: High-End Bottlenecks

- **Evidence:** Specific high-end frames (e.g., "LL Road Frame") take **4 days** to manufacture, whereas standard frames take 1 day.
- **Recommendation:** Initiate a process audit for the "LL Road Frame" production line to

determine if automation or parallel processing can reduce lead time.

Insight 3: Dead Stock Risk

- **Evidence:** Several purchased components have high SafetyStockLevels (>800 units) but very low utilization rates and low unit value.
- **Recommendation:** Reduce safety stock thresholds for low-margin components to free up warehouse space and working capital.

10. Dashboard Documentation (Power BI)

The Power BI dashboard was designed as a "One-Page Executive Summary."

Page Layout:

- **Top Bar:** Key metrics cards (Avg Profit, Total SKUs).
- **Left Pane:** Slicers for Sales_Status and Make_Buy_Indicator.
- **Center:** Main visualizations.

Visuals & Definitions:

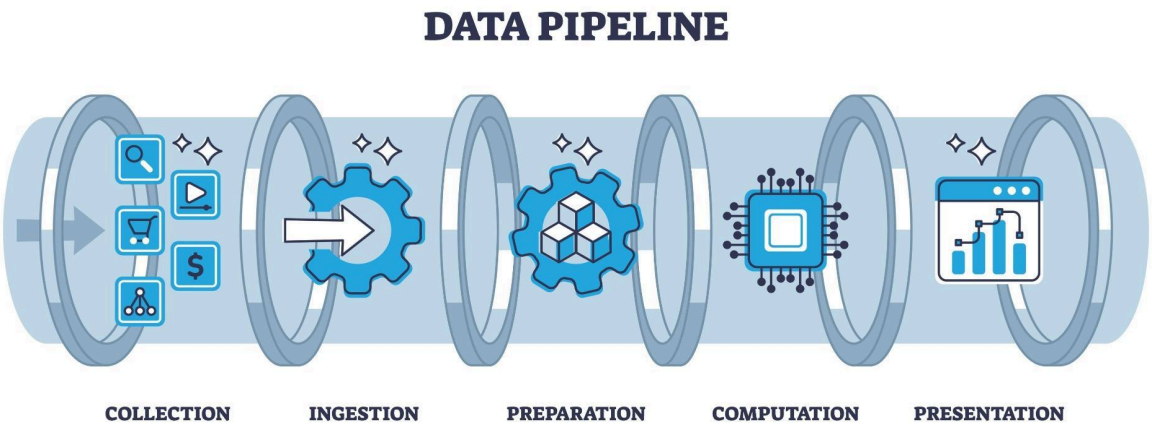
1. **Profitability by Category (Clustered Bar Chart):**
   - *Dimension:* Product_Category
   - *Measure:* Average Profit_Margin
   - *Insight:* Visualizes which product lines are most valuable.
2. **Production Bottlenecks (Table):**
   - *Fields:* Name, DaysToManufacture, Standard_Cost.
   - *Interaction:* Sorted Descending by Days.
   - *Insight:* Highlights immediate risks to the production schedule.
3. **Cost vs. Price (Scatter Plot):**
   - *X-Axis:* Standard_Cost
   - *Y-Axis:* List_Price
   - *Insight:* Shows the correlation between cost and price. Points below the linear trendline indicate underpriced products.

11. End-to-End Pipeline and Architecture

Workflow:

1. **Ingestion:** Raw CSV files are manually downloaded and imported into the PostgreSQL raw_products staging table.
2. **Processing:** SQL scripts define the ETL logic, cleaning the data and handling type conversions.
3. **Storage:** Cleaned data is persisted in the PostgreSQL dim_products table.
4. **Connection:** Power BI connects via Direct Import to the local PostgreSQL server.
5. **Presentation:** The interactive dashboard consumes the dim_products table for reporting.

Data Flow:



Shutterstock

Explore

CSV Source -> PostgreSQL (Staging) -> SQL Transformation -> PostgreSQL (Production) -> Power BI

12. Challenges and Solutions

Challenge 1: Currency Symbols in Numeric Fields

- *Problem:* The CSV file contained $ signs (e.g., $1,200.00), causing import failures when trying to load directly into Integer/Decimal columns.
- *Solution:* Designed the staging table using VARCHAR for all columns to accept the messy data, then used REPLACE() and CAST() functions in SQL to convert them to clean numbers during the transfer to the final table.

Challenge 2: Connecting Power BI to PostgreSQL

- *Problem:* Power BI does not have a native built-in driver for PostgreSQL out of the box.
- *Solution:* Identified the missing Npgsql driver and configured the pg_hba.conf file to allow local connections.

Challenge 3: Interpretation of "Null" Values

- *Problem:* Users were confused by blank entries in the "Color" filter.
- *Solution:* Replaced all NULLs with the text string "N/A" during the ETL process to ensure

the filter UI was clean and understandable.

13. Conclusion

This project successfully established a robust data analytics pipeline for AdventureWorks. By transitioning from static CSV files to a structured SQL database and dynamic Power BI dashboard, the manufacturing team now has real-time visibility into production efficiency.

Key Outcome: The analysis revealed that shifting production focus toward Mountain Bikes could increase overall category profitability, while identifying specific component bottlenecks that delay final assembly.

Future Steps:

- Integrate **Sales Order Data** to weight profitability by actual volume sold.
- Implement Python scripts to automate the daily ingestion of CSV files.

14. Appendix

Data Dictionary (dim_products):

Column Name	Data Type	Description
ProductID	INT	Unique ID
Name	VARCHAR	Product Name
Make_Buy_Indicator	VARCHAR	'Manufactured' or 'Purchased'
Sales_Status	VARCHAR	'Sellable' or 'Not Sellable'
Standard_Cost	NUMERIC	Cost to produce/buy
List_Price	NUMERIC	Selling Price
Profit_Margin	NUMERIC	Calculated Field (Price - Cost)
DaysToManufacture	INT	Days required on production line