Final Project for DEPI

Supply Chain

Retail Stores Inventory and Demand
Technical Report

The Final Project for Top Analyst Team

Fatmaelzahraa Ali Khamiss Ahmed	**Leader**
Heba Mohamed El sherif Mohamed	**Member**
Bola Ghandour Ramzi Hikam	**Member**
Gerges Qadis Moniur Escarous	**Member**
Mariam Khaled Khalil Khalila	**Member**

Introduction

Real-world data is rarely clean and ready for analysis. This project's core objective was to establish data quality and accuracy for the **“Retail Store Inventory and Demand Forecasting”** dataset, making it suitable for subsequent analytical and modeling tasks. This was achieved by executing a robust Data Preparation process followed by comprehensive Data Cleaning utilizing powerful Python libraries such as Pandas and Numpy.

The project was executed in two primary phases, ensuring a logical flow from raw data to a clean and structured analytical model:

1. **Data Preparation (Dimensional Modeling):** Transforming the flat CSV file into a flexible **Star Schema** with designated Dimension and Fact tables.
2. **Data Cleaning:** Applying advanced validation, imputation and correction techniques to the resulting “**Fact_Demand”** table.

Project Index

1. Data Gathering

We started looking into trusted websites about interesting "**Supply Chain"** datasets and we landed on this project from **Kaggle** that analyzes **"Retail Stores Inventory and Demand"** from 2022 to 2024.

https://www.kaggle.com/datasets/chamodperera87/weekly-retail-prices-in-colombo-sri-lanka

2. Data Preparation and Star Schema Generation

2.1. Initial Data Structure and Objective

The project began with a single unnormalized CSV file containing **16 columns**. This structure was inadequate for dimensional analysis. The primary goal of this phase was to generate a dimensional model to explicitly display the relationships between entities and separate descriptive attributes (Dimensions) from quantitative metrics (Facts/Measures).

2.2. Dimension Table Extraction and Key Generation

Five distinct dimension tables were created by extracting unique combinations of categorical attributes from the source data. A unique **Surrogate Key** starting from 1 was generated for each row in the dimension tables to serve as the primary key.

- **Dim_Store:** Extracted unique combinations of the **Store_ID** and **Region** attributes. Its primary key is **Store_Key**.
- **Dim_Product:** Extracted unique combinations of **Product_ID** and **Category**. Its primary key is **Product_Key**.
- **Dim_Date:** Generated from the **Date** column, incorporating **Seasonality**. Its key is **Date_Key** (formatted as YYYYMMDD).
- **Dim_Weather:** Extracted unique **Weather_Condition** values. Its primary key is **Weather_Key**.
- **Dim_Promotion_Event:** Created from the combination of **Discount**, **Promotion** and **Epidemic** flags. Its primary key is **Promo_Event_Key**.

2.3. Fact Table Construction (Fact_Demand)

The central **Fact_Demand** table was constructed by merging the original data with the newly generated dimension tables based on the natural keys. This process resulted in a table that links the five Foreign Keys to the operational metrics (Measures).

- **Final Fact Columns:** The table includes the five Foreign Keys (**Store_Key, Product_Key, Date_Key, Weather_Key, Promo_Event_Key**) and the following six Measures: **Inventory_Level, Units_Sold, Units_Ordered, Price, Competitor_Pricing, and Demand**.
- The generated fact table was initially exported as **Fact_Demand.csv.**

3. Fact_Demand Data Cleaning

The cleaning procedures were applied directly to the structured **Fact_Demand** table, which consists of **243,200 records**.

3.1. Data Type Optimization

The foreign key columns (**Store_Key, Product_Key, Weather_Key, Promo_Event_Key**) were initially loaded as **int64**. To achieve better memory management and potentially faster analytical processing within Pandas, they were explicitly converted to the **Categorical** data type.

3.2. Handling Missing Values (Imputation)

A detailed missing value check revealed that only one key measure was significantly affected:

- **Affected Column: Units_Ordered**.
- **Extent of Missingness: 149,532** records were missing, equating to **61.485%** of the column's data.
- **Action Taken:** Due to the high percentage of missing data, imputation was necessary. The missing values in **Units_Ordered** were filled using the **Median** value of the column. The median was chosen over the mean because it is less sensitive to extreme outliers, ensuring a more robust estimate for a count-based variable.

3.3. Outlier Correction (Negative Prices)

A review of the descriptive statistics identified logical inconsistencies (outliers) in the **Price** column:

- **Inconsistency Found: 3** records contained negative values for price.
- **Action Taken:** Since prices cannot logically be negative in a retail context, these values were corrected by applying the **.abs() (absolute value)** function to the entire **Price** column. This successfully converted the three negative prices into their positive, non-negative equivalents, ensuring compliance with business logic.

3.4. Consistency and Duplication Checks

Two crucial checks were performed to ensure data integrity:

- **Duplicate Row Check:** A check for completely identical rows across all columns was performed.
    - **Result: Zero** duplicate rows were found, confirming the structural uniqueness of the dataset.
- **Business Logic Check:** The data was validated to ensure that the number of units sold never exceeded the inventory level at that specific time and location (i.e., **Units_Sold** > **Inventory_Level**).
    - **Result: Zero** rows were found to be logically inconsistent, validating the sales and inventory metrics.

4. Conclusion and Export

Following the rigorous dimensional modeling and detailed cleaning process, the **Fact_Demand** dataset is now fully prepared for advanced analytical tasks. All data types have been optimized, missing values have been robustly imputed, outliers have been corrected based on business rules, and the final dataset is free of duplicates and logical inconsistencies.

The final, clean fact table was exported to the file **'Fact_Demand_Cleaned.csv'**.

5. Data Visualization

Applying advanced visualization using Power BI, DAX and Python.

5.1. Dashboard: Financial Performance & Growth

This analysis relies on data from the Transaction/Order table, specifically leveraging columns for Units Ordered, Units Sold and Revenue, using the date column for time series analysis. Stacked Bar Charts are employed for comparing annual revenue on a month by month basis (YOY Comparison). The Fulfillment Rate, calculated as (Units_Sold / Units_Ordered), is a critical technical metric. The large gap observed in this rate indicates a potential data pipeline issue or significant leakage in the order processing stage. The Revenue Forecasting, displayed via a Line Chart, is based on a time series regression model (such as ARIMA or Prophet) which requires model stationarity tests.

5.2. Dashboard: Inventory & Operations

This dashboard integrates data from both Inventory and Sales tables and requires geographical context for regional performance classification. Key metrics like Inventory Turnover Rate and Current Inventory Value are presented using KPI Cards. A Pie/Donut Chart is used for the geometric segmentation of total sales by region. Crucially, the Scatter Plot visualizes the relationship between Units Ordered and Actual Demand. This chart indicates a strong, positive linear correlation between the two variables. The regional variance, particularly the lower performance in the Southern region, suggests the need for a multivariate analysis to identify the specific factors influencing inventory turnover there.

5.3. Dashboard: Pricing & Promotions

This analysis is based on pricing and promotional data, including Discount Rate, Promo Flag, Base Sales and Competitor Price. The Promo Period Uplift is a key derived technical metric, calculated by comparing sales during the promotional period to a calculated baseline. This methodology is often employed in A/B testing or incremental sales analysis. The data is used to determine the Price Elasticity of Demand. The close alignment between demand and competitor pricing shown in the Line Chart suggests the presence of cross-price elasticity, necessitating an econometric modeling approach for developing an algorithmic pricing strategy.

5.4. Dashboard: External Factors & Advanced Analysis

This dashboard involves integrating internal sales data with external data such as Weather Condition (a categorical variable) and a binary temporal flag (e.g., Pandemic_Flag). The primary statistical tool used to quantify the relationship between these variables is the Pearson Correlation Coefficient. The result of 0.00 for the correlation between the Pandemic_Flag and Demand indicates a lack of a linear relationship. This finding suggests that while a significant sales drop occurred, the overall impact on demand was non-linear or related to confounding variables like distribution channel availability. The sales analysis by weather condition relies on data partitioning using the categorical weather variable.