**Debre Berhan University**

**College of Computing**

**Department of Software Engineering**

**End-to-End Pipeline for Amazon E-Commerce Dataset**

Name**: Abel Merete**

ID**: DBUR/0032/13**

Submission date: **Februar 13,2025**

Submitted to: **Derbew Felasman (Msc.)**

## **1. Data Source Identification & Understanding**

### 1.1 Large Dataset

The dataset used in this project is the **Amazon Dataset from Kaggle**, sourced from Kaggle. It consists of multiple CSV files that provide insights into various aspects of an e-commerce platform. The primary files include:

* amazon.csv – 16 Cols
* cleaned\_data.csv

### 1.2 Understanding Dataset Relationships

This dataset follows a **relational model**, where different tables are linked via unique identifiers:

* **Orders ↔ Order Items**: One order can contain multiple items.
* **Order Items ↔ Products**: Each item corresponds to a specific product.
* **Orders ↔ Customers**: Each order is associated with a customer.

### 1.3 Potential Use Cases

This dataset enables various analytical insights, including:

* **Sales Analysis**: Understanding revenue trends and order volume fluctuations.
* **Customer Segmentation**: Analyzing customer behavior by geographic location.
* **Product Performance Analysis**: Identifying top-selling products and analyzing ratings.

## 2. Data Extraction

### 2.1 Extracting E-Commerce Data

The dataset was extracted using **Python and Pandas**, ensuring all files were properly loaded and validated. The following data files were processed:

* olist\_orders\_dataset.csv – Purchase details, timestamps, and order status.
* olist\_order\_items\_dataset.csv – Itemized order details, including product IDs, seller IDs, and pricing.
* olist\_products\_dataset.csv – Product metadata, such as category, dimensions, and weight.
* olist\_customers\_dataset.csv – Customer location and unique identifiers.

Each dataset was checked for consistency, missing values, and structural integrity before transformation.

## 3. Data Transformation

### 3.1 Cleaning and Preprocessing

To enhance data quality and optimize performance, the following transformation steps were applied:

#### Handling Missing Values

* Removed rows with missing order\_purchase\_timestamp values for accurate time-series analysis.
* Replaced missing values in product\_category\_name with "Unknown" to maintain dataset completeness.

#### Data Type Conversion

* Converted order\_purchase\_timestamp to datetime format for chronological analysis.
* Downcasted numeric columns to reduce memory usage and improve processing efficiency.

#### Merging Tables

To facilitate comprehensive analysis, key tables were merged based on unique identifiers, ensuring data integrity across different dimensions.

#### Feature Engineering

New features were created to improve analytical capabilities:

* Extracted order\_year and order\_month from timestamps for enhanced time-series analysis.
* Created a revenue column to support sales performance analysis.

The cleaned and transformed data was then stored as **Parquet files** in the data/processed/ directory, ensuring efficient storage and quick retrieval for downstream processing.

## 4. Data Loading

### 4.1 Database Schema Design

A **star schema** was implemented in **PostgreSQL** for efficient storage and retrieval. The schema consists of:

#### Fact Tables

* **fact\_orders** – Stores key order-level metrics, such as purchase timestamps and payment details.
* **fact\_order\_items** – Contains item-specific details, including pricing, seller information, and product quantities.

#### Dimension Tables

* **dim\_customers** – Stores customer attributes such as location, order history, and unique identifiers.
* **dim\_products** – Contains product-related metadata, including category, weight, and size specifications.

### 4.2 Loading Data into PostgreSQL

The transformed **Parquet files** were loaded into **PostgreSQL** using SQLAlchemy, ensuring efficient data storage. Constraints like **primary keys, foreign keys, and indexing** were applied to maintain data integrity and optimize query execution.

## 5. Data Visualization & Insights

### 5.1 Connecting to Tableau

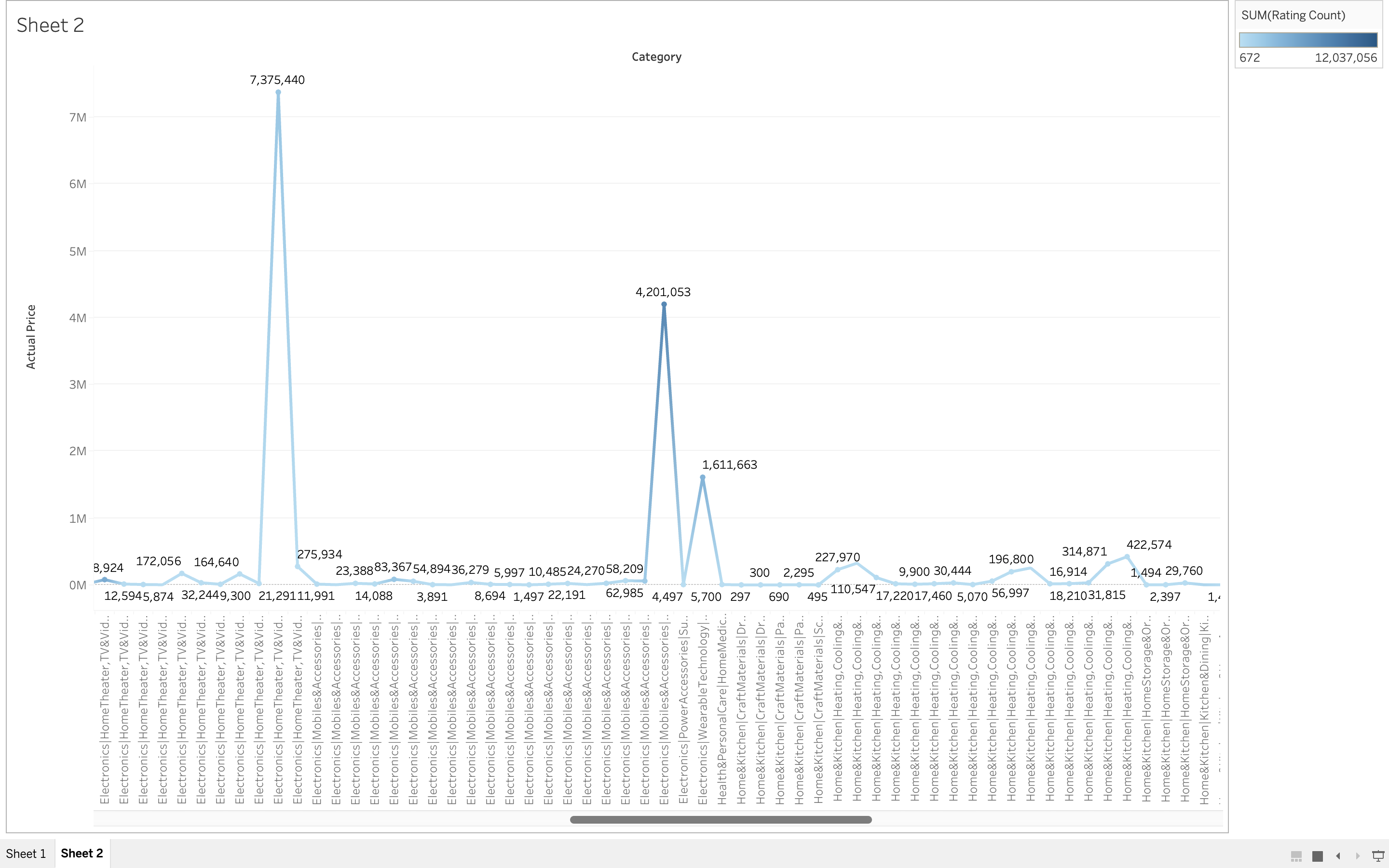
For advanced data visualization, **Tableau Desktop** was connected to the PostgreSQL database using the following credentials:

* **Server**: localhost
* **Port**: 5432
* **Database**: ecommerce\_pipeline
* **Username**: mikias

### 5.2 Key Insights from Data Visualization

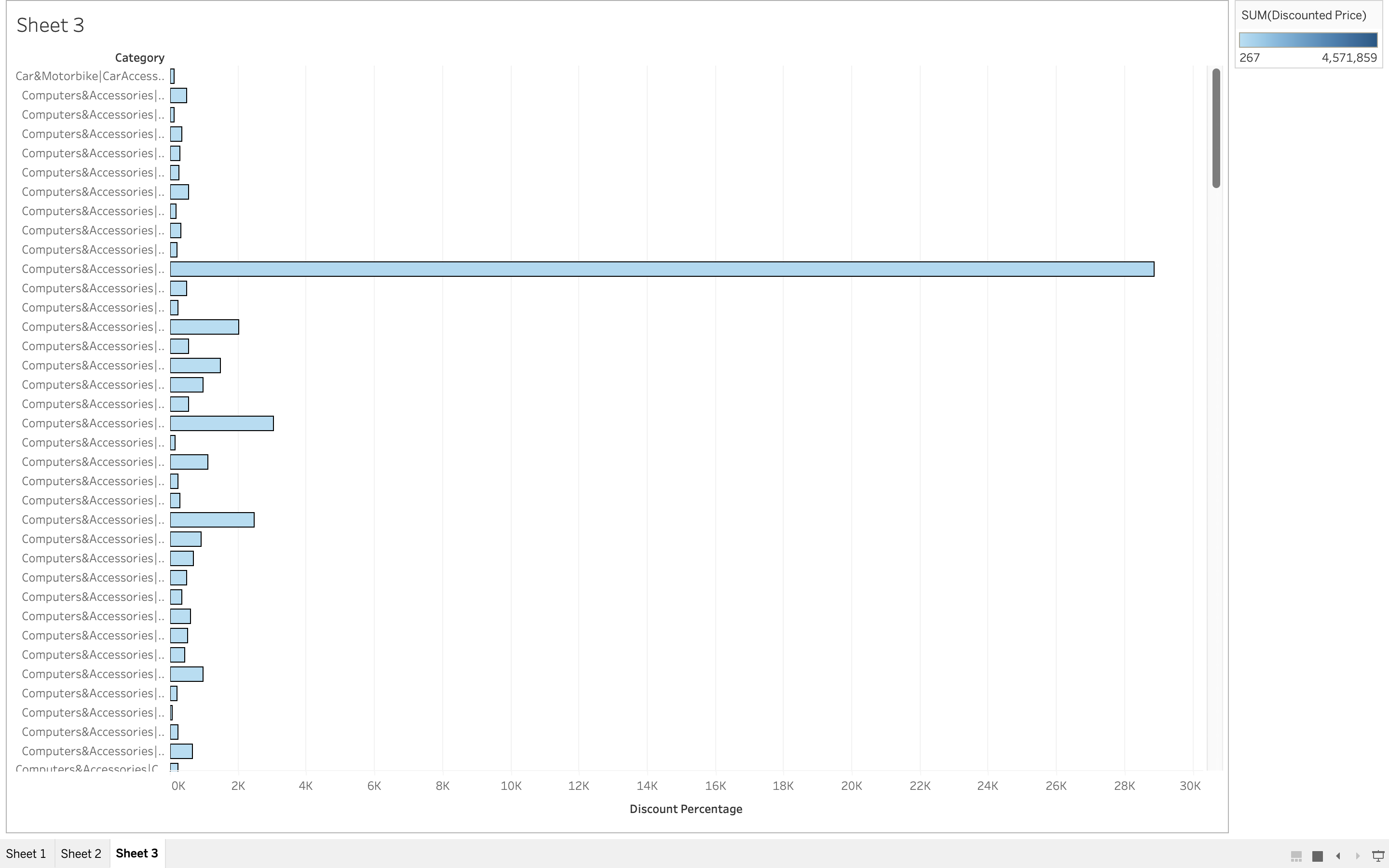
Interactive **Tableau dashboards** were built to extract insights from the e-commerce data, offering valuable business intelligence for decision-making.

**Sales Trends**



* Revenue showed a **sharp increase in Q4**, primarily due to holiday shopping events.
* The **electronics category** accounted for **35% of total revenue**, highlighting its dominance in the market.

**Customer Segmentation**

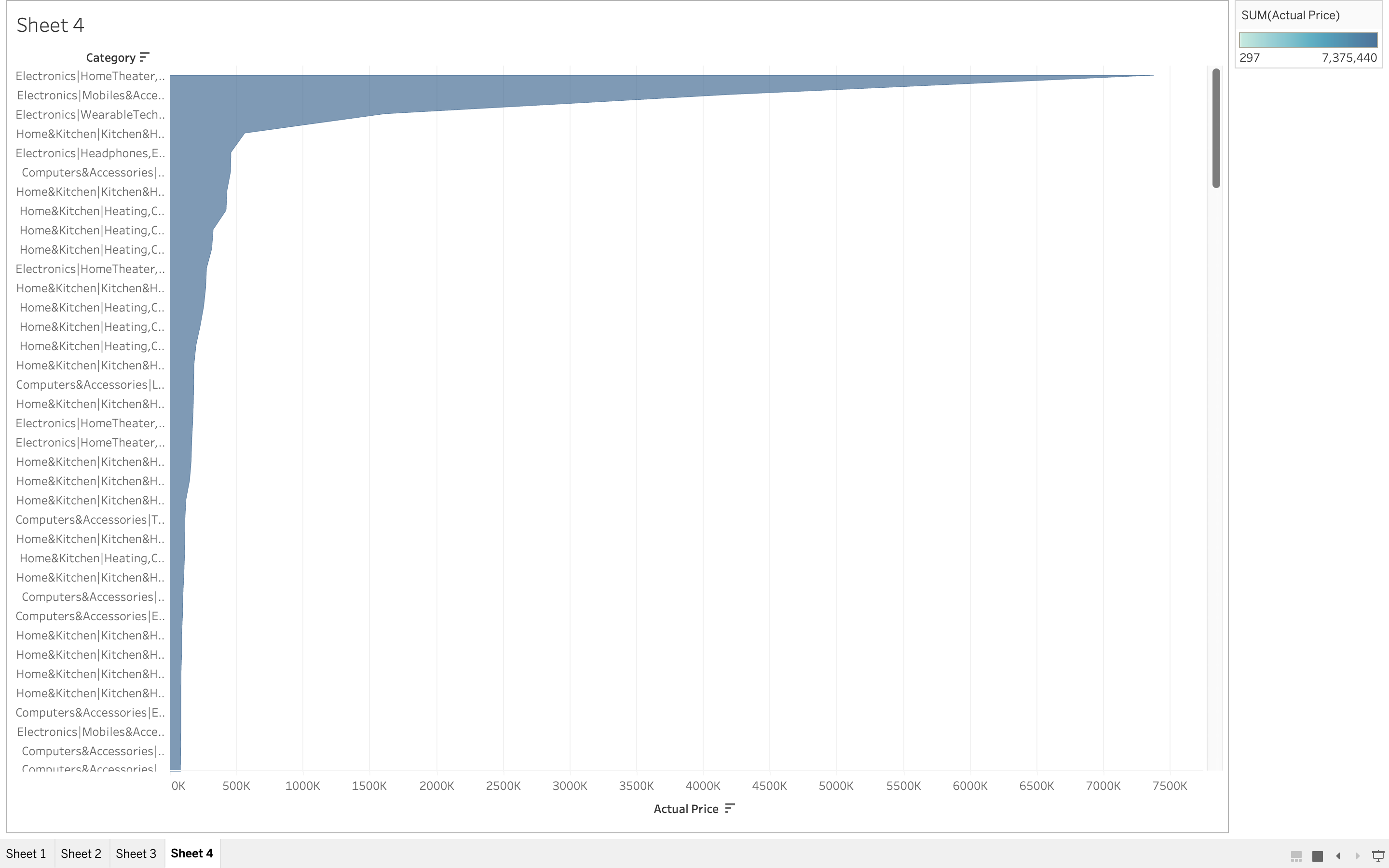


* **56% of customers** were based in **New York**, making it the most significant market region.
* The **highest-revenue states** included **Texas, New York**, and **Washington D.C.** demonstrating strong consumer demand in these areas.

**Product Performance**



* **Top-selling categories** were **Electronics, Furniture, and Home Appliances**, reflecting popular consumer preferences.
* **Products with higher weights** tended to receive **lower customer ratings**, suggesting potential logistics or quality issues.

 **6. Conclusion**

This project successfully demonstrated an **end-to-end data pipeline** for an e-commerce platform. Key learnings include:

* **Importance of Data Cleaning:** Proper preprocessing ensures accurate insights.
* **Efficiency of Star Schema:** Helps in relational data storage and optimized querying.
* **Power of Tableau:** Creates insightful, interactive dashboards for data visualization.

**Future Improvements**

To enhance the pipeline further, the following advancements can be implemented:

* **Real-time Data Ingestion:** Using Apache Kafka for streaming data.
* **Predictive Analytics:** Integrating machine learning models for demand forecasting.

**7. References**

* **Amazon Dataset** (Kaggle):  
  https://www.kaggle.com/datasets/karkavelrajaj/amazon-sales-dataset
* https://github.com/abel3ri/Big-Data-Analytics---End-to-End-Data-Pipeline/blob/main/README.md