Cleaning and Analysis of a PostgreSQL Super Store Database

Executive Summary

This project demonstrates advanced data cleaning and analytical skills using SQL with a complex Super Store database. The analysis focuses on identifying the top product categories and products based on sales and profit while addressing key data quality challenges in preparation for business reporting. The approach highlights skills in joining multiple tables, aggregating results, and ranking products—all essential for business intelligence and reporting tasks.

Project Objectives

- Clean and analyze sales and product data stored in PostgreSQL.
- Identify **top-performing products** within each main category by total sales and profit.
- Rank products for targeted business decision-making.
- Demonstrate robust SQL skills including window functions, CTEs, type handling, and multi-table joins.

Technologies & Tools Used

- **Database:** PostgreSQL
- **Analysis Environment:** Jupyter Notebook (`notebook.ipynb`)
- **Key SQL Techniques:**
    - JOINs across orders, products, returns, and people tables
    - Data type casting (e.g., text to date/number)
    - Aggregations, window functions (`RANK()`), and CTEs
    - Data cleaning steps (handling nulls, incorrect types)

Dataset Description

This Super Store database consists of several tables:

- **orders:** Sales transactions (with text dates and numeric fields as double precision)
- **returned_orders:** Records of returned orders
- **people:** Salesperson details, connected by region
- **products:** Product catalog with categories and subcategories

Note: Data required careful cleaning due to patterns like text-based dates and inconsistent numeric types.

Methodology

1. **Data Exploration & Cleaning:**
    - Inspected table schemas and clarified relationships from the data dictionary.
    - Addressed incorrect data types (e.g., converted `order_date` from text to date).
    - Handled numeric field normalization for accurate aggregation.
2. **Product Ranking Logic:**
    - Joined `orders` with `products` to aggregate sales and profit by product.
    - Utilized a CTE and window function (`RANK()`) to rank products within each category by total sales.
    - Selected the **top 5 products by sales for each category**, including their total profits.
3. **Result Formatting:**
    - Organized output into clear, recruiter-ready tables for business review.

Key SQL Query

```
WITH product_sales AS (
  SELECT
    p.category,
    p.product_name,
    ROUND(SUM(o.sales)::numeric, 2) AS product_total_sales,
    ROUND(SUM(o.profit)::numeric, 2) AS product_total_profit
  FROM orders o
  JOIN products p ON o.product_id = p.product_id
  GROUP BY p.category, p.product_name
),
ranked_products AS (
  SELECT *,
    RANK() OVER (PARTITION BY category ORDER BY product_total_sales DESC) AS
product_rank
  FROM product_sales
)
SELECT
  category,
  product_name,
  product_total_sales,
  product_total_profit,
  product_rank
FROM ranked_products
WHERE product_rank <= 5
ORDER BY category ASC, product_rank ASC;
```

Results & Insights

Top 5 Products by Sales in Each Category:

Category	Product Name	Total Sales	Total Profit	Rank
Furniture	Hon Executive Leather Armchair, Adjustable	58,193.48	5,997.25	1
Furniture	Office Star Executive Leather Armchair, Adjustable	51,449.80	4,925.80	2
Furniture	Harbour Creations Executive Leather Armchair, Adjustable	50,121.52	10,427.33	3

...	...	...	...	...
Office Supplies	Eldon File Cart, Single Width	39,873.23	5,571.26	1
...	...	...	...	...
Technology	Apple Smart Phone, Full Size	86,935.78	5,921.58	1
Technology	Cisco Smart Phone, Full Size	76,441.53	17,238.52	2
...	...	...	...	...

- **Apple Smart Phone** led technology sales, while top Leather Armchairs dominated the furniture category.
- Products with top sales didn't always yield the highest profit—demonstrating the importance of evaluating both metrics for strategy.

Challenges & Solutions

Challenge	Solution
Inconsistent data types	SQL casting of text dates/numbers to correct types
Data normalization	Applied rounding and type conversion for currency
Large, multi-table structure	Used JOINS and CTEs for modular, readable queries

Conclusion

This project demonstrates the **critical role of data cleaning and structured SQL queries** for business analytics—translating raw transactional records into actionable rankings for sales, marketing, and category management. The approach is generalizable to any sales-driven enterprise and highlights expertise in multi-table data organization, cleaning, and advanced SQL analytics.

Next Steps

- Expand to sub-category and person-based performance analysis.
- Integrate returned order data for a holistic profitability assessment.
- Automate scorecard/dashboards for rolling performance reviews.

See the Jupyter notebook (notebook.ipynb) for SQL context, output tables, and reproducibility.