Retail Business Performance & Profitability Analysis

Introduction

The goal of this project is to perform a complete end-to-end analysis of a retail sales dataset using PostgreSQL, Python (Pandas), and Tableau. The project begins with data cleaning and management in PostgreSQL to ensure accuracy and consistency. It then moves into exploratory data analysis in Python, focusing on understanding the relationship between inventory holding periods and profitability. Finally, an interactive Tableau dashboard is built to visualize sales trends, profitability, and inventory patterns, allowing dynamic filtering by region, product category, and season. This integrated approach aims to uncover key business insights and support data-driven decision-making.

1.PostgreSQL Data Import & Analysis

Overview

This part of the project focuses on the structured import, cleaning, and analysis of a retail sales dataset using **PostgreSQL**. The goal was to prepare the data for analysis, ensure data integrity, and derive meaningful insights such as profit margins and performance by product category.

Step 1: Table Creation in PostgreSQL

The first step was to design and create a relational table named retail_data in PostgreSQL to store the retail sales information. Each column was assigned an appropriate data type, including DATE, VARCHAR, INT, and FLOAT. A SERIAL primary key (Row_ID) was added to uniquely identify each record.

Query Query History

```
1  CREATE TABLE retail_data (
2      Row_ID SERIAL PRIMARY KEY,
3      Order_ID VARCHAR(50),
4      Order_Date TEXT,     -- Use TEXT first
5      Ship_Date TEXT,
6      Ship_Mode VARCHAR(50),
7      Customer_ID VARCHAR(50),
8      Customer_Name VARCHAR(255),
9      Segment VARCHAR(50),
10     Country VARCHAR(50),
11     City VARCHAR(100),
12     State VARCHAR(100),
13     Postal_Code VARCHAR(20),
14     Region VARCHAR(50),
15     Product_ID VARCHAR(50),
16     Category VARCHAR(50),
17     Sub_Category VARCHAR(50),
18     Product_Name VARCHAR(255),
19     Sales FLOAT,
20     Quantity INT,
21     Discount FLOAT,
22     Profit FLOAT
23 );
```

Data Output Messages Notifications

```
CREATE TABLE

Query returned successfully in 51 msec.
```

Step 2: Importing Data into PostgreSQL

The retail sales data was imported from a CSV file into the retail_data table using the COPY command. The CSV file was encoded using WIN1252 to prevent encoding issues that may occur when using other formats (e.g., UTF-8).

```
26 COPY retail_data (Row_ID, Order_ID, Order_Date, Ship_Date, Ship_Mode, Customer_ID, Customer_Name, Segment,
27                   Country, City, State, Postal_Code, Region, Product_ID, Category, Sub_Category, Product_Name,
28                   Sales, Quantity, Discount, Profit)
29 FROM 'C:/ProgramData/Microsoft/Windows/Start Menu/Programs/PostgreSQL 16/Sales.csv'
30 WITH (FORMAT csv, HEADER, DELIMITER ',', QUOTE '"', ENCODING 'WIN1252');
31
32
33
```

Data Output Messages Notifications

```
COPY 9994

Query returned successfully in 113 msec.
```

This command imports the CSV data into the PostgreSQL database while handling the CSV file's formatting and ensuring proper character encoding.

Step 3: Data Cleaning

To ensure the dataset was clean and valid for analysis, we removed any rows where key columns such as Order_ID, Product_ID, Sales, or Profit were NULL. This ensured that only complete records were retained for analysis.

```
31
32  DELETE FROM retail_data
33  WHERE
34      Order_ID IS NULL
35      OR Product_ID IS NULL
36      OR Sales IS NULL
37      OR Profit IS NULL
38      OR Category IS NULL
39      OR Sub_Category IS NULL;
40
```

Data Output | Messages | Notifications

```
DELETE 0

Query returned successfully in 78 msec.
```

Step 4: Profit Margin Analysis

a) Profit Margin by Category

We calculated the profit margin for each product category. Profit margin is calculated as the ratio of Profit to Sales, expressed as a percentage.

```sql
SELECT
    Category,
    ROUND((SUM(Profit)::numeric / NULLIF(SUM(Sales), 0)::numeric) * 100, 2) AS Profit_Margin_Percentage
FROM retail_data
GROUP BY Category
ORDER BY Profit_Margin_Percentage DESC;
```

Data Output | Messages | Notifications

	category character varying (50)	profit_margin_percentage numeric
1	Technology	17.40
2	Office Supplies	17.04
3	Furniture	2.49

This query helps determine the overall profitability of different product categories, helping to identify which categories yield the highest margins.

b) Profit Margin by Sub-Category

For a more granular view, we also calculated the profit margin for each **sub-category** within every category.

```sql
SELECT
    Category,
    Sub_Category,
    ROUND((SUM(Profit)::numeric / NULLIF(SUM(Sales), 0)::numeric) * 100, 2) AS Profit_Margin_Percentage
FROM retail_data
GROUP BY Category, Sub_Category
ORDER BY Category, Profit_Margin_Percentage DESC;
```

Data Output | Messages | Notifications

	category character varying (50)	sub_category character varying (50)	profit_margin_percentage numeric
1	Furniture	Furnishings	14.24
2	Furniture	Chairs	8.10
3	Furniture	Bookcases	-3.02
4	Furniture	Tables	-8.56
5	Office Supplies	Labels	44.42
6	Office Supplies	Paper	43.39
7	Office Supplies	Envelopes	42.27
8	Office Supplies	Fasteners	31.40

Total rows: 17 of 17 Query complete 00:00:00.108

This analysis helps to identify which specific sub-categories are performing better or worse in terms of profit.

Step 5: Additional Analytical Queries

a) Top 10 Products by Total Profit

To understand which products are driving the most profit, we calculated the total profit for each product and displayed the top 10 most profitable products.

```sql
SELECT
    Product_Name,
    SUM(Sales) AS Total_Sales,
    SUM(Profit) AS Total_Profit
FROM retail_data
GROUP BY Product_Name
ORDER BY Total_Profit DESC
LIMIT 10;
```

Data Output | Messages | Notifications

	product_name character varying (255)	total_sales double precision	total_profit double precision
1	Canon imageCLASS 2200 Advanced Copier	61599.824	25199.928000000004
2	Fellowes PB500 Electric Punch Plastic Comb Binding Machine with Manual Bi…	27453.384	7753.039
3	Hewlett Packard LaserJet 3310 Copier	18839.686	6983.8836
4	Canon PC1060 Personal Laser Copier	11619.833999999999	4570.9347
5	HP Designjet T520 Inkjet Large Format Printer - 24" Color	18374.895	4094.9766
6	Ativa V4110MDD Micro-Cut Shredder	7699.89	3772.9461
7	3D Systems Cube Printer, 2nd Generation, Magenta	14299.89	3717.9714000000004
8	Plantronics Savi W720 Multi-Device Wireless Headset System	9367.289999999999	3696.2819999999997
9	Ibico EPK-21 Electric Binding System	15875.916000000001	3345.2823
10	Zebra ZM400 Thermal Label Printer	6965.700000000001	3343.536

This query identifies the products with the highest total profit, which are crucial for sales and marketing strategies

b) Bottom 10 Sub-Categories by Profit Margin

Lastly, we identified the bottom 10 sub-categories that have the lowest profit margins. These sub-categories are potential areas for cost control, pricing adjustments, or promotional efforts.

```sql
SELECT
    Category,
    Sub_Category,
    ROUND((SUM(Profit)::numeric / NULLIF(SUM(Sales), 0)::numeric) * 100, 2) AS Profit_Margin_Percentage,
    SUM(Profit) AS Total_Profit
FROM retail_data
GROUP BY Category, Sub_Category
ORDER BY Profit_Margin_Percentage ASC
LIMIT 10;
```

Data Output | Messages | Notifications

	category character varying (50)	sub_category character varying (50)	profit_margin_percentage numeric	total_profit double precision
1	Furniture	Tables	-8.56	-17725.481100000008
2	Furniture	Bookcases	-3.02	-3472.5559999999978
3	Office Supplies	Supplies	-2.55	-1189.0994999999984
4	Technology	Machines	1.79	3384.7569
5	Furniture	Chairs	8.10	26590.166300000026
6	Office Supplies	Storage	9.51	21278.826399999998
7	Technology	Phones	13.49	44515.7306
8	Furniture	Furnishings	14.24	13059.143599999985
9	Office Supplies	Binders	14.86	30221.763299999995
10	Office Supplies	Appliances	16.87	18138.005399999995

You are currently ru

This helps pinpoint areas where improvements can be made to increase profitability.

Python Analysis of Inventory Days and Profitability

Overview:

In this section of the project, we utilized Python and Pandas to analyze the relationship between **Inventory Days** and **Profitability**. Our goal was to explore whether there exists any correlation between these two variables and to gain insights into the overall distribution of **Inventory Days** and **Profitability** across the dataset.

Data Processing and Calculations:

Data Preprocessing- The dataset was first loaded into a Pandas DataFrame and necessary date columns, namely Order_Date and Ship_Date, were converted into datetime format to facilitate date-based calculations.

```
# Convert columns to datetime
data['Order Date'] = pd.to_datetime(data['Order Date'], format='%m/%d/%Y')
data['Ship Date'] = pd.to_datetime(data['Ship Date'], format='%m/%d/%Y')

# Display the first few rows to check date conversion
data[['Order Date', 'Ship Date']].head()
```

	Order Date	Ship Date
0	2016-11-08	2016-11-11
1	2016-11-08	2016-11-11
2	2016-06-12	2016-06-16
3	2015-10-11	2015-10-18
4	2015-10-11	2015-10-18

Inventory Days was computed as the difference between the Ship_Date and Order_Date. This metric represents the time taken for each product to be shipped after the order was placed.

```
# Calculate Inventory Days (difference between Ship Date and Order Date)
data['Inventory_Days'] = (data['Ship Date'] - data['Order Date']).dt.days

# Display the first few rows with the new Inventory_Days column
data[['Order ID', 'Order Date', 'Ship Date', 'Inventory_Days']].head()
```

	Order ID	Order Date	Ship Date	Inventory_Days
0	CA-2016-152156	2016-11-08	2016-11-11	3
1	CA-2016-152156	2016-11-08	2016-11-11	3
2	CA-2016-138688	2016-06-12	2016-06-16	4
3	US-2015-108966	2015-10-11	2015-10-18	7
4	US-2015-108966	2015-10-11	2015-10-18	7

Profitability was calculated as the ratio of **Profit** to **Sales** for each record, indicating how profitable each sale was.

```
# Calculate Profitability as Profit / Sales
data['Profitability'] = data['Profit'] / data['Sales']

# Display the first few rows with the Profitability column
data[['Order ID', 'Profit', 'Sales', 'Profitability']].head()
```

	Order ID	Profit	Sales	Profitability
0	CA-2016-152156	41.9136	261.9600	0.1600
1	CA-2016-152156	219.5820	731.9400	0.3000
2	CA-2016-138688	6.8714	14.6200	0.4700
3	US-2015-108966	-383.0310	957.5775	-0.4000
4	US-2015-108966	2.5164	22.3680	0.1125

Correlation Analysis Between Inventory Days and Profitability:

We calculated the correlation between **Inventory Days** and **Profitability** using the Pandas .corr() function. This function measures the linear relationship between the two variables.

```
# Calculate the correlation between Inventory Days and Profitability
correlation = data['Inventory_Days'].corr(data['Profitability'])

# Output the correlation result
print(f"Correlation between Inventory Days and Profitability: {correlation:.2f}")
```

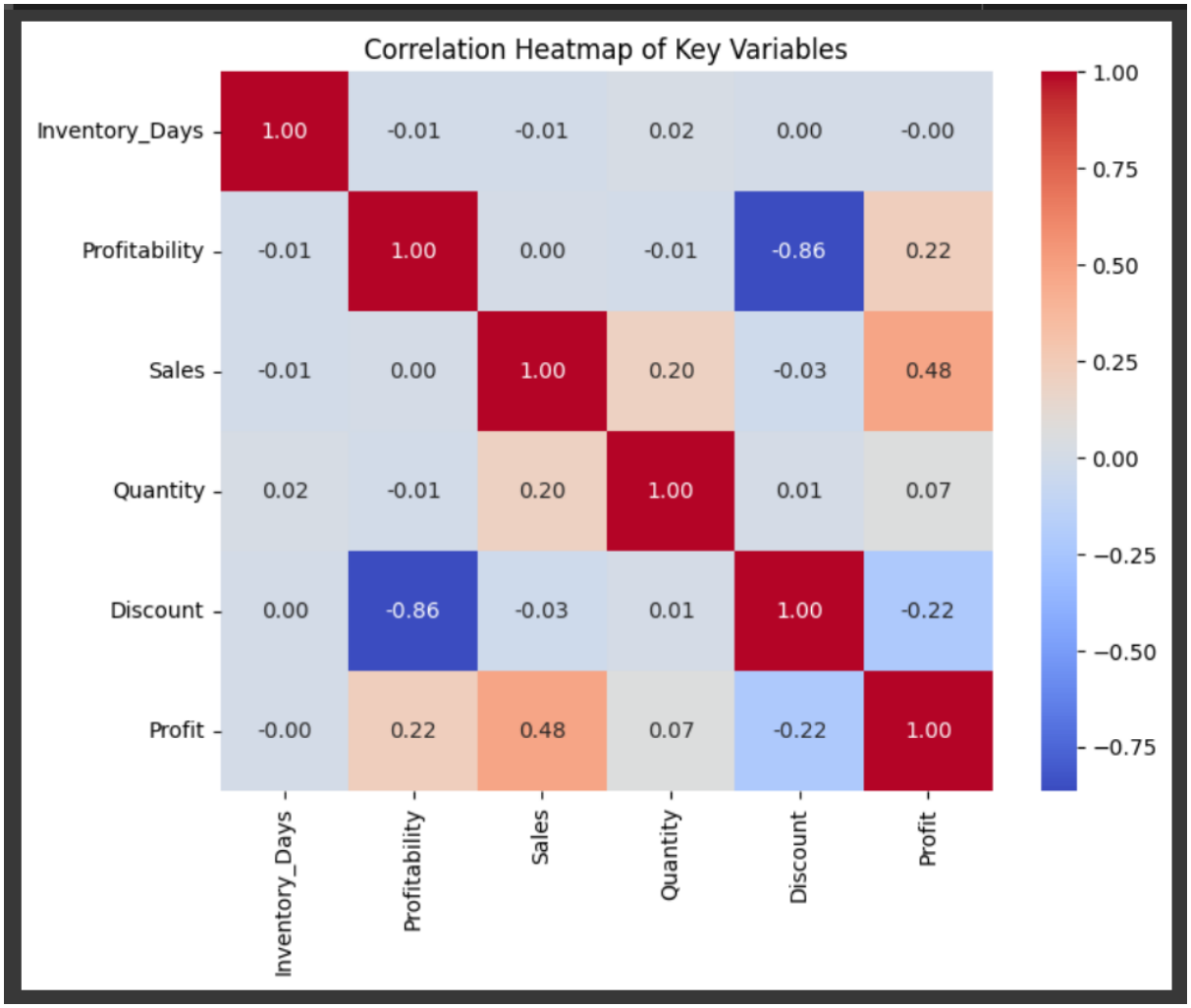
```
Correlation between Inventory Days and Profitability: -0.01
```

The result was stored in the correlation variable. I then printed the correlation value rounded to two decimal places for clarity. This helped us understand if Inventory Days had any significant effect on Profitability.

Visualizations:

To better understand the relationship between **Inventory Days** and **Profitability**, the following visualizations were created:

1. **Correlation Heatmap**: The **Correlation Heatmap** visualized the correlation matrix of key variables, with a focus on **Inventory Days**, **Profitability**, and other relevant factors such as **Sales** and **Profit**. This heatmap provides a clear, color-coded overview of how strongly different variables in the dataset are related to each other.



Key Findings:

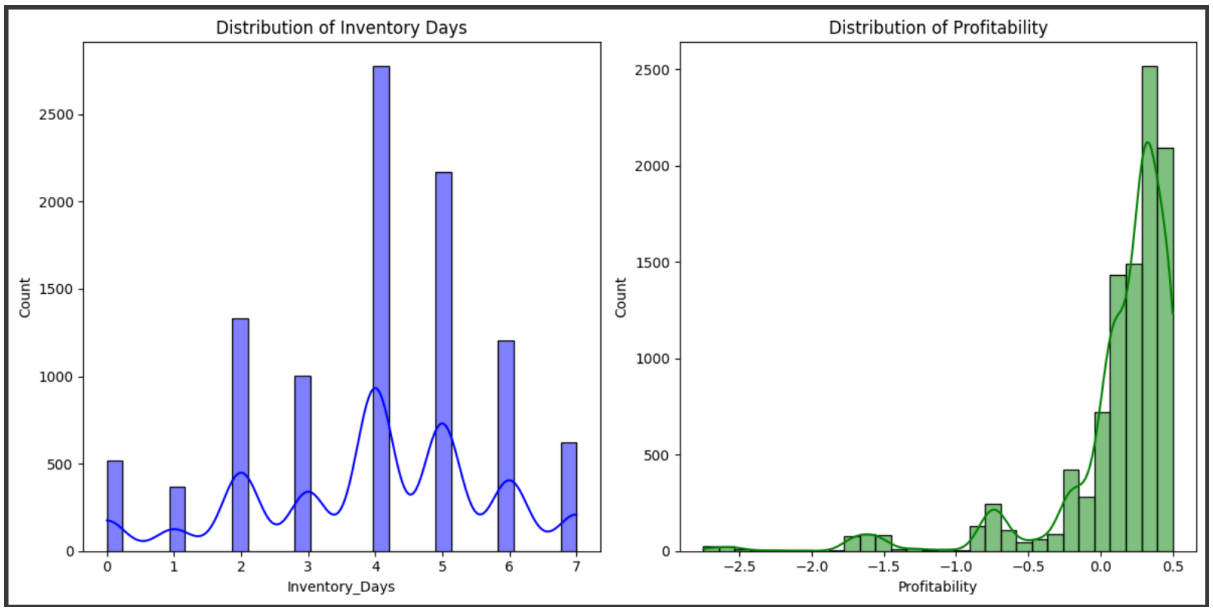
The result showed a very weak negative correlation of -0.01, indicating almost no direct relationship. This was visualized using a Correlation Heatmap to compare Inventory Days, Profitability, Sales, Quantity, Discount, and Profit.
The heatmap confirmed that Inventory Days had minimal impact on Profitability across the dataset.

2. **Distribution Analysis of Inventory Days and Profitability:**

We analyzed the distribution patterns for **Inventory Days** and **Profitability** to understand their behavior across the dataset.

- The distribution of **Inventory Days** is slightly uneven, showing peaks at around 2, 4, and 5 days, suggesting that most products are sold within a few days of inventory holding.

- On the other hand, the distribution of **Profitability** is skewed towards positive values, with a large concentration between 0 and 0.5, indicating that the majority of transactions are modestly profitable.



Key Insights:

- **Inventory Days** are most commonly clustered between 2 to 5 days, indicating fast-moving inventory in most cases.
- **Profitability** is positively skewed, with most transactions yielding small but positive profits.
- There are very few negative profitability records, which implies limited unprofitable sales.
- No strong outliers were visible, and both variables show relatively predictable and stable behaviour.
- This supports the earlier correlation finding that **Inventory Days have little to no significant impact on Profitability**.

Findings and Interpretation

- **Weak Correlation:**
  The correlation analysis between **Inventory Days** and **Profitability** showed a very weak negative correlation of **-0.01**, indicating almost no linear relationship. In practice, this means that the time a product spends in inventory does not significantly impact its profitability.
- **Distribution Insights:**
  - **Inventory Days:** The distribution revealed that most products are sold within 2–5 days, suggesting generally fast-moving inventory with minor inefficiencies in certain cases.
  - **Profitability:** The profitability distribution is skewed positively, showing that most sales are modestly profitable, with very few losses observed.

Overall, both distributions demonstrate stable patterns without major outliers, supporting the conclusion that inventory holding time has minimal effect on profitability in this dataset.

Part 3: Tableau Dashboard Development

Objective

The objective of this part was to create an interactive Tableau dashboard that provides meaningful insights into sales, profitability, inventory days, and regional performance. To enhance user interaction, filters were added for **Region**, **Product Type (Category)**, and **Season**.

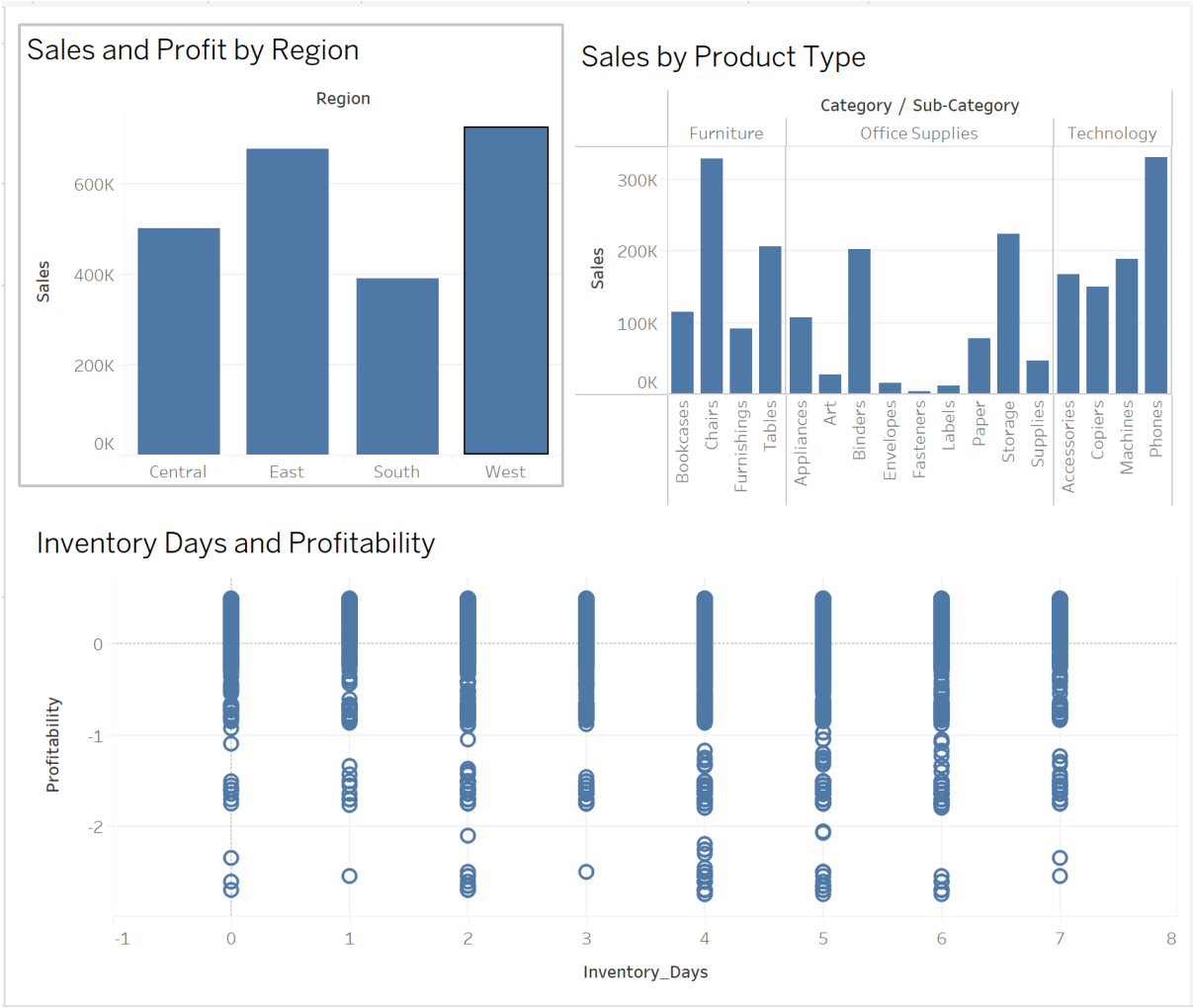
Process Overview

1. **Data Connection**
   The sales dataset was imported into Tableau directly from the cleaned file. Fields such as Order Date, Sales, Profit, Category, and Region were verified for correct data types.
2. **Data Preparation**
   A calculated field named **"Season"** was created using the Order Date to categorize sales into Winter, Spring, Summer, and Fall, ensuring seasonality analysis was possible.
3. **Visualizations Built**
   - **Sales and Profit by Region**: A bar chart was developed to show regional performance across key metrics.
   - **Sales by Product Type**: A chart displaying Sales and Profit segmented by Category and Sub-Category.
   - **Inventory Days vs. Profitability Scatter Plot**: To explore the relationship between inventory holding times and profitability.
   - **Summary Metrics and KPIs**: Quick overview cards showing total sales, total profit, and average inventory days.
4. **Filters Incorporated**
   - **Region**: Allows users to view data specific to different regions.
   - **Category (Product Type)**: Enables filtering by furniture, office supplies, technology, etc.
   - **Season**: Users can explore performance variations across Winter, Spring, Summer, and Fall.

5. **Dashboard Design**
   The dashboard was designed to be visually intuitive, with filters placed on the side for easy accessibility. Consistent color schemes were applied to distinguish between profit and loss and highlight key trends.

Key Features of the Dashboard

- Dynamic and interactive filtering across Region, Product Type, and Season.
- Quick visual comparisons of profitability and sales performance.
- Insight into how inventory duration relates to profitability across products and time periods.



Conclusion

This project involved importing and cleaning the sales data using PostgreSQL, analyzing relationships between inventory days and profitability using Python (Pandas), and creating an interactive dashboard in Tableau. The analysis found a very weak negative correlation between inventory days and profitability, suggesting minimal impact. The Tableau dashboard enabled dynamic insights across regions, product categories, and seasons, helping to better understand sales and profit trends. Overall, the project demonstrated a complete workflow from data management to business insight generation.