ADIDAS SALES DATA ANALYSIS

A PROJECT REPORT

Submitted by

NISHANT REDDY (1RVU23CSE120)

DHANUSH S GOWDA (1RVU23CSE145)

ANKUR T S (1RVU23CSE061)

DARSHAN B (1RVU23CSE135)

in partial fulfillment for the award of the degree of

B.Tech (Hons.) – Computer Science & Engineering

in **Data Analysis with Python**
to Dr. Sahana Prasad

School of Computer Science and Engineering
RV University
RV Vidyaniketan, 8th Mile, Mysuru Road, Bengaluru, Karnataka, India – 562112.

December 2023

ACKNOWLEDGEMENT

It is a great pleasure for us to acknowledge the assistance and support of a large number of individuals who have been responsible for the successful completion of this project work.

First, we take this opportunity to express our sincere gratitude to the **School of Computer Science and Engineering, RV UNIVERSITY**, **Bengaluru** for providing us with a great opportunity to pursue our Bachelor's Degree in this institution.

In particular, we would like to thank our professor, **Dr. Sahana Prasad, Data Analysis in Python Professor, RV UNIVERSITY**, **Bengaluru** for sparing her valuable time to extend help in every step of our project work, which paved the way for smooth progress and fruitful culmination of the project.

We would also like to thank **Dr. Sanjay R. Chitnis, Dean, School of Computer Science and Engineering, RV UNIVERSITY**, **Bengaluru** for his constant encouragement and expert advice.

It is a matter of immense pleasure to express our sincere thanks to **Dr. Mydhili K Nair, Head of Department, School of Computer Science & Engineering, RV UNIVERSITY**, **Bengaluru** for providing right academic guidance that made our task possible.

We are also grateful to our family and friends who provided us with every requirement throughout the course. We would like to thank one and all who directly or indirectly helped us in completing the Project work successfully.

Name: **NISHANT REDDY** Signature
USN: 1RVU23CSE120

Name: **DHANUSH S GOWDA** Signature
USN: 1RVU23CSE145

Name: **ANKUR T S** Signature
USN: 1RVU23CSE061

Name: **DARSHAN B** Signature
USN: 1RVU23CSE135

TABLE OF CONTENTS

1. INTRODUCTION

This project delves into the intricacies of Adidas Sales in the USA during 2020 & 2021, driven by the recognition of Adidas as a pivotal player in the clothing industry. The rationale behind selecting this dataset lies in its potential to offer valuable insights into consumer behaviour, market trends, and operational strategies within the clothing and sportswear sector.

Our objective encompasses deciphering patterns, identifying influential factors impacting sales, and furnishing actionable insights beneficial for strategic decision-making.
The significance of our analysis extends beyond the confines of a semester project. By comprehensively understanding Adidas sales dynamics, we aim to provide a foundation for future endeavours in market research and data analytics. The insights derived can inform Adidas's marketing strategies, inventory management, and overall business decisions. Furthermore, this analysis serves as a benchmark for other industry stakeholders seeking to enhance their understanding of market dynamics and consumer preferences.

The project required meticulous efforts from our team, spanning data preprocessing, exploratory data analysis, and the application of statistical methods. Challenges, such as handling missing data and outliers, were addressed systematically to ensure the integrity and reliability of our findings. The collaborative nature of our team, encompassing diverse skill sets in Python Programming and Statistical Analysis, has been instrumental in the success of this project.
The project's relevance lies not only in its contribution to academic learning but also in its practical applications for industry players. The effort invested in understanding and interpreting the data contributes to a holistic understanding of the complexities inherent in data analysis.

In conclusion, this project serves as a stepping stone for future investigations into market dynamics, consumer behavior, and data-driven decision-making. The insights gleaned from our analysis present a valuable resource for academia, industry professionals, and Adidas alike, positioning this endeavor as a meaningful contribution to the field of data analysis and business intelligence.

2. METHODOLOGY

Table 2.1: Detailed methodological steps involved

Sl no.	Process	Description
1.	Data Collection	The data collection process involves gathering, measuring, and recording information or data points, typically through systematic methods such as surveys, interviews, or sensor technology. We collected Adidas Sales data from the years 2020 & 2021 in USA from various open data base sources and Kaggle.
2.	Data Cleaning (Data Preprocessing)	The data cleaning process involves identifying and rectifying errors, inaccuracies, and inconsistencies in a dataset to ensure its accuracy, reliability, and suitability for analysis.
3.	Data Analysis	The data analysis process involves deriving insights from data through organizing, exploring, interpreting, and modeling, to make informed decisions and solve problems effectively. We used various data analysis tools in Python, including Pandas and Numpy to analyze the data and perform feature engineering.
4.	Data Visualization	The data visualization process involves representing and presenting data in visual form using charts, graphs, and other visual elements to facilitate understanding and analysis. We used Matplotlib and Seaborn to create visualizations that insights into key trends and patterns in the sales data.

3. INFERENCE

1. Importing necessary libraries and dataset.

We begin by importing essential libraries for data manipulation and visualization. Then, we load the Adidas sales dataset into a Pandas data frame named 'df' and display the first 10 rows to get an initial look at the data.

2. Data Inspection.

This section provides a quick overview of the dataset. We check its dimensions, column names, and information about data types and null values to understand its structure. There were 9648 rows and 13 columns out of which 12 columns were of type ‘object’ and one ‘int64’.

3. Data Cleaning (Data Preprocessing).

This part involves cleaning and preprocessing the data. We removed ‘$’ and ‘%’ symbols from all columns and converted them to numeric type, calculate new columns for total sales and operating profit, ensure text consistency by converting to lowercase, and handle the 'Invoice Date' column as datetime. There were no duplicate rows and columns involved in the dataset.

4. Exploratory Data Analysis (EDA).

We explore the dataset by obtaining descriptive statistics and calculating the range of numerical columns to understand the spread and characteristics of the data.

```
range of Price_per_Unit : 103.0
range of Units_Sold : 1275.0
range of Total_Sales : 82500.0
range of Operating_Profit : 39000.0
range of Operating_Margin : 70.0
```

Under EDA, we performed the following type of analysis to better understand the data set in all perspectives.

a. *Sales Analysis*

In this section, we could identify that the most number of times listed product was ‘men’s athletic footwear’. But Sales wise the ‘men’s street footwear’ had the highest total sales.
The graph below illustrates the timeline of total sales and operating profits over the course of several months.

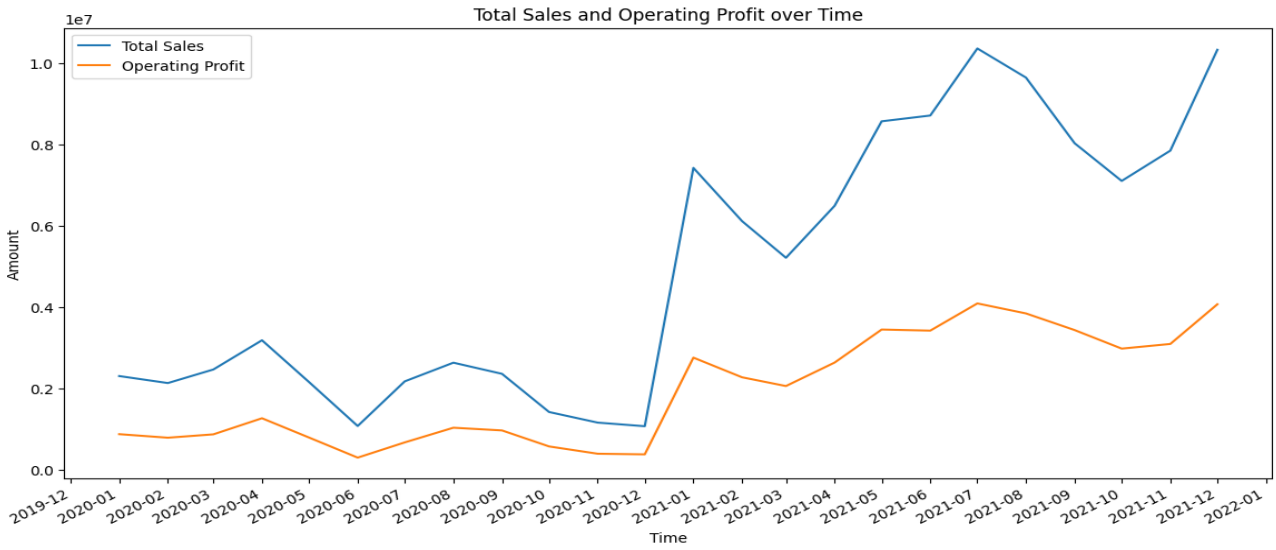


Figure 1: Total Sales and Operating Profit for each month from Jan 2020 - Dec 2021

By analyzing the timeline graph (Figure 1) from December 2020 to January 2021, a noticeable surge in total sales is apparent. However, the corresponding increase in operating profit is not proportional. This disparity could be attributed to significant discounts offered during the Christmas and New Year period or expenditure on advertisements. A similar trend is observed in December 2021. It's important to highlight that, post-December 2020, the company emphasized sales over profits.

Sales Method	Total Sales
in-store	35664375.0
online	44965657.0
outlet	39536618.0

Sales Method	Operating Profit
in-store	12759128.75
online	19552537.72
outlet	14913301.23

Sales Method	Variation of units sold
in-store	203.410458
online	176.269773
outlet	232.193750

Among various sales methods (Online, In-Store, Outlet), ‘Online’ exhibits the highest total sales and profit, while ‘In-Store’ records the lowest. ‘Outlet’ shows the highest standard deviation, indicating significant daily unit sales variability, whereas ‘Online’ has the lowest variation. This observation suggests a consistent preference among consumers for online purchases.

The below information provided outlines the average number of units sold per product per day, essentially reflecting the average daily transaction.

Product	Average units sold
men's apparel	190.960772
men's athletic footwear	270.513043
men's street footwear	368.521739
women's apparel	269.792910
women's athletic footwear	197.531756
women's street footwear	243.948383

Notably, 'Men's Street Footwear' exhibits the highest sold product per day, while 'Men's Apparel' indicates the least.

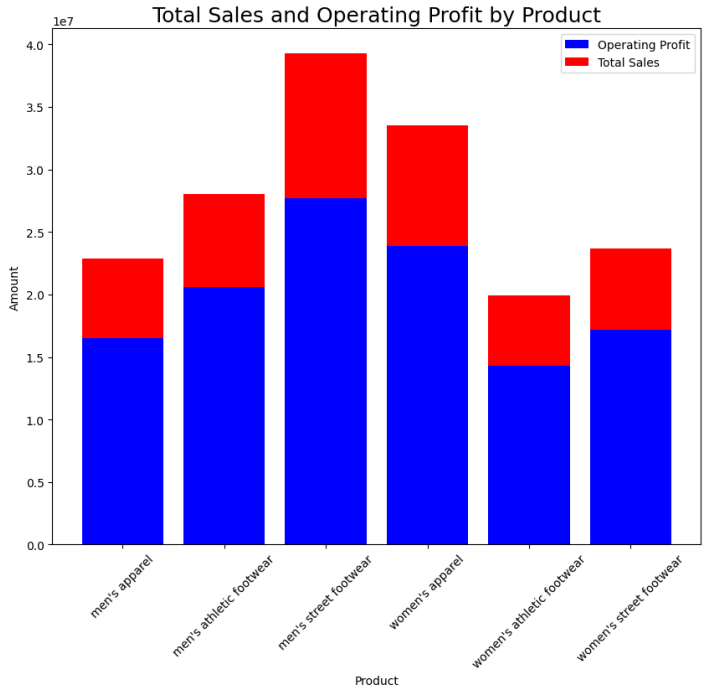


Figure 2: Total Sales and Operating profit of each product

From Figure 2, it is clear that men's street footwear has the highest total sales and operating profit.

b. Profitability Analysis.

This segment focuses on profitability analysis. We determine the median operating profit and median operating margin by product.

Product	median operating profit
men's apparel	2679.415
men's athletic footwear	3293.760

men's street footwear	5201.500
women's apparel	4004.200
women's athletic footwear	2357.100
women's street footwear	2703.000

Product	Operating margin median
men's apparel	40.0
men's athletic footwear	40.0
men's street footwear	45.0
women's apparel	44.0
women's athletic footwear	41.0
women's street footwear	40.0

Interestingly, 'men's street footwear' emerges with the highest median operating profit and the highest median operating margin. Furthermore, a time series plot (Figure 3) is generated to visually depict the fluctuation of the median operating margin over time.

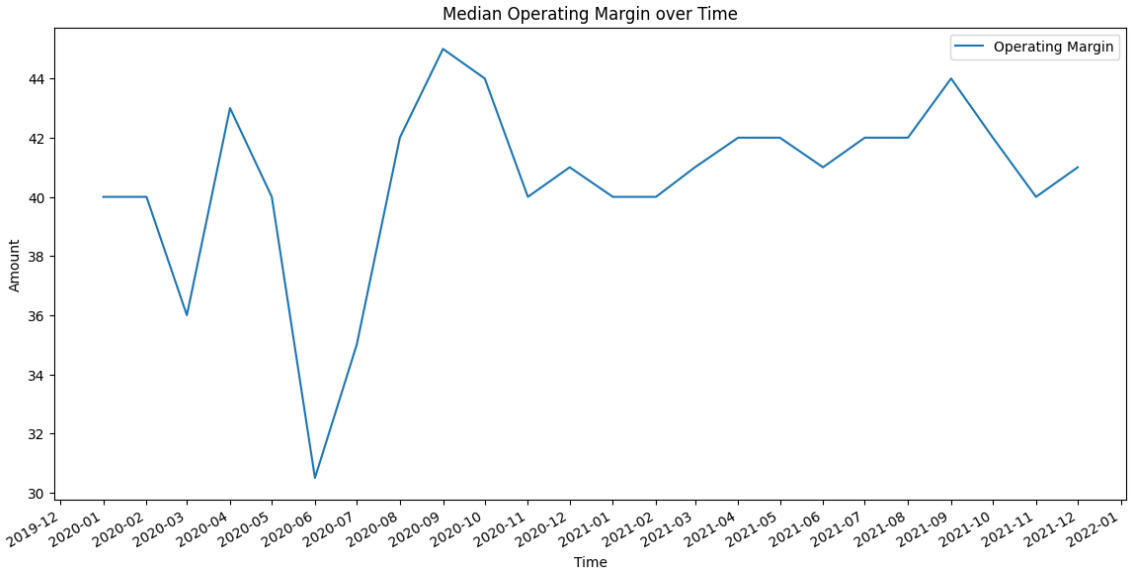


Figure 3: Operating margin for each month from Jan 2020 - Dec 2021

Significantly, during June 2020, the company experienced its lowest operating margin in a two-year span. This decline may be attributed to increased production costs or various economic factors, including the influence of the Covid-19 pandemic. Despite the company's sustained sales, the operating margin was adversely affected during this specific period.

Upon analyzing the operating profits data over the two-year period, the calculated kurtosis (7.181) and skewness (2.335) reveal significant insights. The positive kurtosis signals a distribution with heavier tails and a sharper peak compared to a normal distribution. Moreover, the positive skewness indicates a right-skewed distribution, suggesting a notable stretch towards the right side of the data distribution.

The operating margin distribution for each sales method exhibits positive skewness.

Sales Method	skewness of operating margin
in-store	0.363159

online	0.251876
outlet	0.148255

Notably, 'In-Store' records the highest skewness, while 'Outlet' demonstrates the lowest. The positive skewness in operating margins for each sales method suggests that there are occasional high values that extend the right tail of the distribution.

c. Regional Analysis.

In this section, we conduct a regional analysis. Generally, the western region of USA exhibits the highest sales for the company, with New York emerging as the state and city with the highest sales.

Region wise: Region	Total sales
midwest	16674434.0
northeast	25078267.0
south	20603356.0
southeast	21374436.0
west	36436157.0
State wise: new york	
City wise: new york	

Interestingly, ‘Men's Apparel’ stands out as the most popular product both at the city and state levels. However, on a broader scale ‘Men's Athletic Footwear’ takes the lead in popularity within the western region. It's crucial to note that the definition of popularity here is based on the frequency of product listings rather than the quantity of products sold. When considering the actual number of products sold, 'Men's Street Footwear' emerges as the more preferred choice among consumers overall.

The below information shows the total profits made in two years for each sales method in a particular region.

Region	Sales Method	Total operating profit
midwest	in-store	2316565.00
	online	3133263.98
	outlet	1410116.25
northeast	in-store	4254420.00
	online	2246831.65
	outlet	3231522.25
south	in-store	134800.00
	online	4149888.22
	outlet	4936917.10
southeast	in-store	2558256.25
	online	5080401.63
	outlet	754401.32
west	in-store	3495087.50
	online	4942152.24
	outlet	4580344.31

While the 'West' region boasts the overall highest sales, it is noteworthy that in the 'Southeast' region, the 'Online' sales method achieves the highest total operating profit.

d. Retailer Analysis.

This section focuses on retailer analysis. We calculate total sales and operating profit by retailer, visualize this information with pie charts, and explore average operating profit by retailer and sales method.

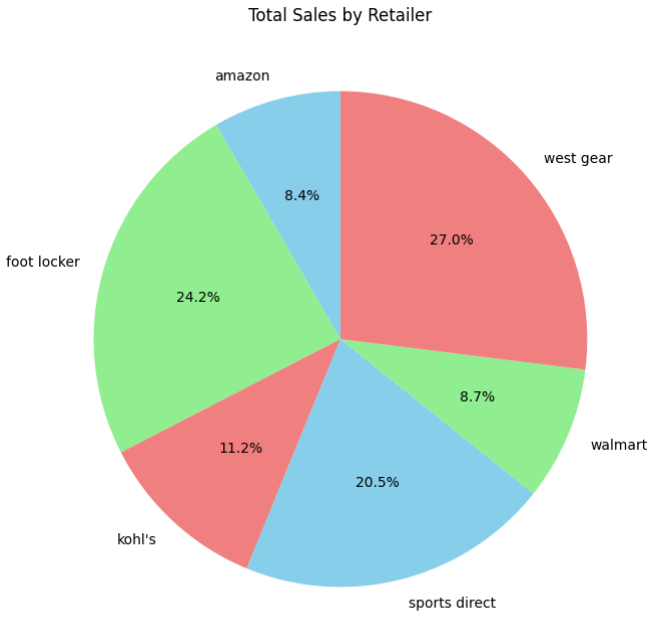


Figure 4: Total Sales contribution by each retailer

This pie chart (Figure 4) shows the total sales by each retailer. West Gear has the most total sales, followed by Foot Locker. Even though online sales are highest overall, it's interesting that Amazon has the lowest total sales among the retailers.

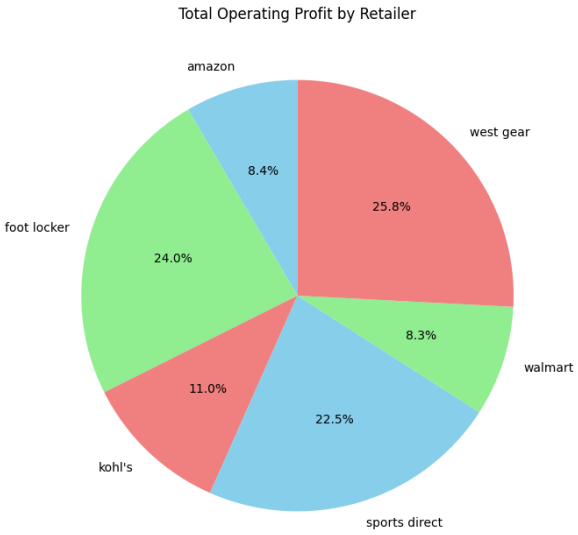


Figure 5: Total Operating profit contribution by each retailer

From Figure 5, it is clear that the retailer with the highest operating profit is West Gear, closely followed by Foot Locker. Despite online mode achieving the highest operating profit overall, it is notable that Amazon reports the least operating profit among the retailers.

Retailer	Sales method	Average operating profit
amazon	in-store	7083.972458
	online	3773.055296
	outlet	3818.121821
foot locker	in-store	6256.570156
	online	3717.388738
	outlet	4189.243405
kohl's	in-store	7359.071181
	online	3831.638194
	outlet	6179.129774
sports direct	in-store	7035.712457
	online	4542.114372
	outlet	5457.985430
walmart	in-store	13325.337838
	online	4781.102743
	outlet	6753.334784
west gear	in-store	7868.115165
	online	3859.311032
	outlet	4260.573448

Despite ‘West Gear’ securing the highest total operating profit, Walmart outperformed them in terms of average operational profits. While online mode dominated in overall operating profit, a closer examination by retailer's sales method revealed that in-store transactions yielded the highest average operating profit.
The below lists the most popular (most frequently listed) products sold by each retailer. This gives us an idea about how each retailer is making more profits through their products.

Retailer	Product	Highest number of times listed
amazon	men's athletic footwear	159
	men's street footwear	159
	women's apparel	159
	women's street footwear	157
foot locker	men's street footwear	449
	women's apparel	433
kohl's	men's athletic footwear	172
	men's street footwear	172
	women's apparel	172
	women's street footwear	172
sports direct	women's street footwear	342
walmart	men's apparel	113
west gear	women's street footwear	400

e. Pricing Analysis.

Finally, we enter the domain of pricing analysis. We calculate the correlation matrix for relevant columns and visualize the correlations using a heatmap.

	Price_per_Unit	Units_Sold	Total_Sales	Operating_Profit	Operating_Margin
Price_per_Unit	1.000000	0.265869	0.539547	0.503683	-0.137486
Units_Sold	0.265869	1.000000	0.919339	0.871993	-0.305479
Total_Sales	0.539547	0.919339	1.000000	0.935372	-0.302295
Operating_Profit	0.503683	0.871993	0.935372	1.000000	-0.047491
Operating_Margin	-0.137486	-0.305479	-0.302295	-0.047491	1.000000

If Price per unit is more, the total sales & operating profit will be more since they are positively correlated. Conversely, price per unit shows a negative weak correlation with operating margin.

The correlation between price per unit and units sold is positive but weak, suggesting that an increase in price per unit doesn't guarantee a proportional increase in units sold. There is a strong positive correlation between units sold and operating profit, as well as between units sold and total sales, which aligns logically.

Interestingly, operating margin and units sold exhibit a negative moderate correlation. This implies that higher sales or units sold may be associated with increased advertising expenses, potentially leading to a scenario where sales are high, but the profit margin is comparatively lower.

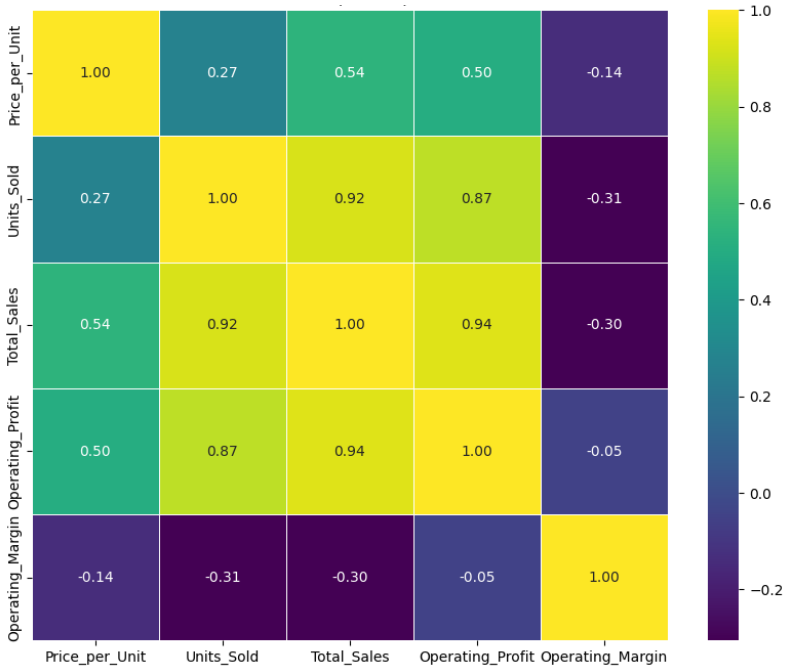


Figure 6: Heat map to show the correlation between different variables.

From figure 7, we can say that in general, the price per unit for each product follows a normal distribution. However, 'men's athletic footwear' exhibits a slight left skewness which suggests that a majority of units are priced lower, with some higher-priced units pulling the distribution towards the right, while 'Women's Athletic Footwear' shows a slight right skewness which suggests that a majority of units are priced higher, with some lower-priced units pulling the distribution towards the left.

The median price per unit is highest for ‘women's apparel’ and ‘men's apparel’, whereas ‘women's street footwear' and 'women's athletic footwear' have the lowest median price per unit. Moreover, the presence of numerous outliers toward the right side of the graph is notable for each product.

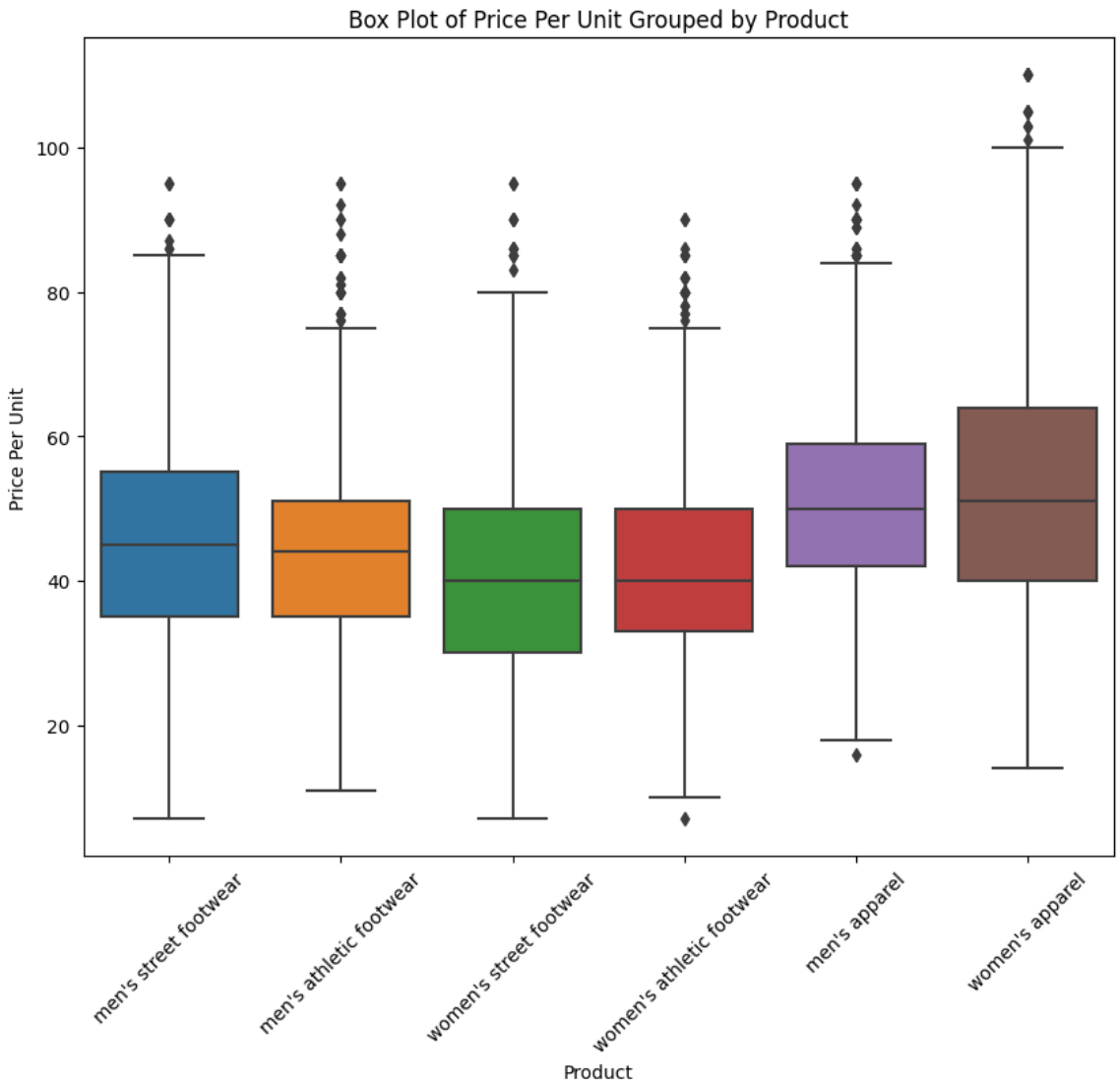


Figure 7: Box plot for various products with respect to price per unit

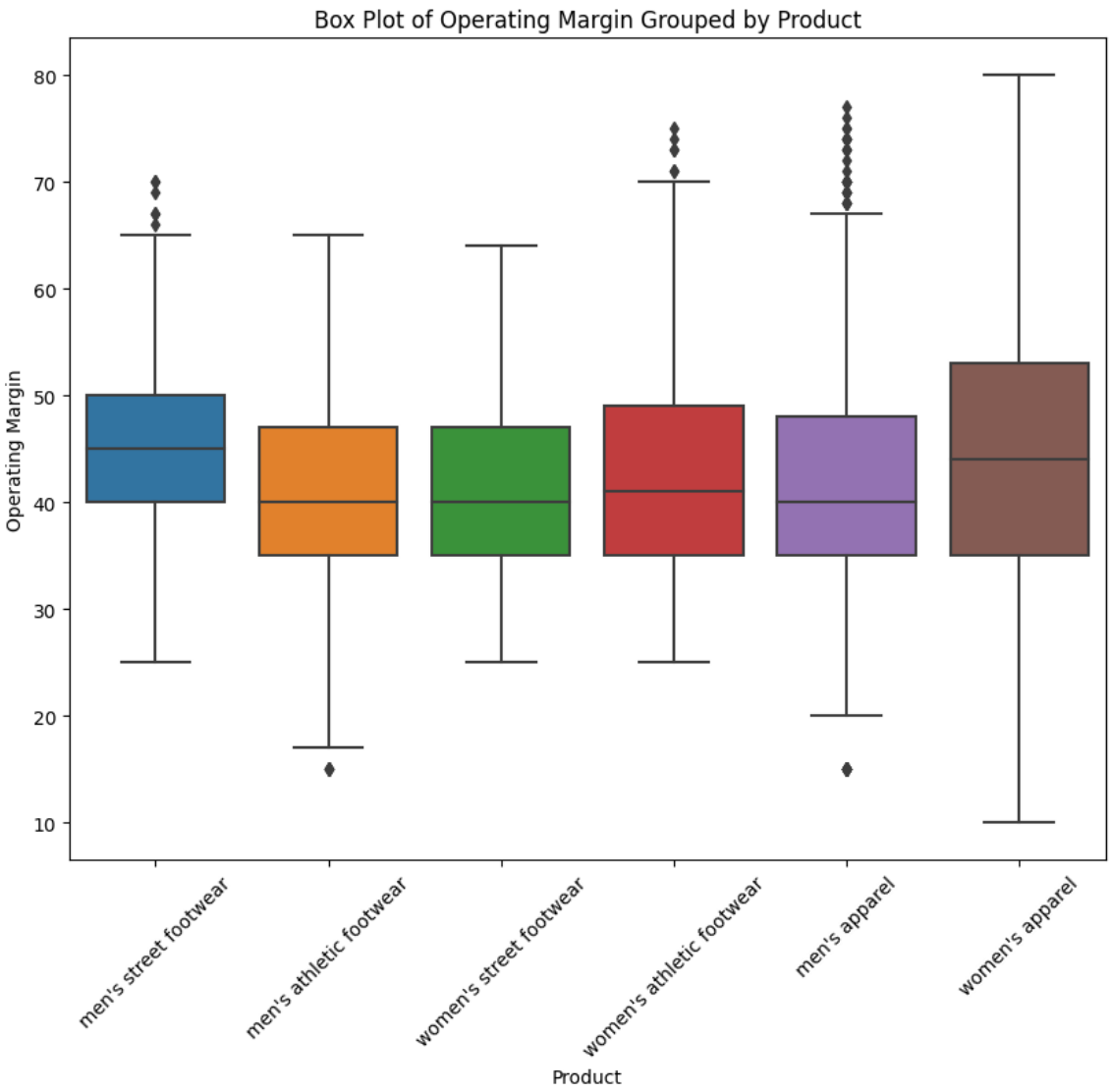


Figure 8: Box plot for various products with respect to operating margin

In general, the operating margin distribution for each product exhibits slight right skewness. However, interestingly, for 'Men's Street Footwear' and 'Women's Apparel,' the operating margin follows a normal distribution.

left skewness indicates that there are instances where the operating margin is lower than the average, and these lower values are influencing the overall distribution. This could be due to factors such as higher operating costs, lower profitability, or other issues affecting the financial performance of the product.

In practical terms, left skewness suggests that there are some periods or instances where the operating margin for the product is lower than usual, potentially impacting the profitability of that product.

4. RESULT AND DISCUSSION

1. Data Cleaning (Data Preprocessing):
    - The process of removing symbols and converting data types is well-executed, ensuring consistency.
    - The handling of datetime columns is appropriately carried out.
    - However, further insights into why these specific cleaning steps were chosen and their potential impact on the analysis would enhance transparency.
2. Exploratory Data Analysis (EDA):
    - The identification of the most listed and highest sales product provides valuable insights.
    - The timeline graph effectively captures patterns, but deeper exploration into the factors causing the surge in total sales post-December 2020 would enhance the analysis.
3. Sales Analysis:
    - The identification of the most frequently listed product ('Men's Athletic Footwear') and the highest total sales product ('Men's Street Footwear') is insightful.
    - The timeline graph effectively captures patterns, particularly the surge in total sales post-December 2020.
4. Profitability Analysis:
    - The focus on median operating profit and margin is commendable.
    - The time series plot offers visibility into fluctuations, but a more detailed exploration of the June 2020 decline would provide a richer understanding.
5. Regional Analysis:
    - The recognition of the western region, especially New York, as the leader in overall sales is well-established.
    - The popularity of 'Men's Athletic Footwear' in the western region is an interesting observation.
6. Retailer Analysis:
    - The calculation of total sales and operating profit by retailer is thorough.
    - While pie charts visually represent the data, additional insights into why certain retailers outperform others would strengthen the analysis.
7. Pricing Analysis:
    - The correlation matrix and heatmap effectively visualize relationships between variables.
    - The interpretation of positive correlations aligns logically, but delving deeper into potential causation would add depth.
    - The box plots offer valuable insights into the distribution characteristics of product prices and operating margins.

A new data frame is being created, organized by region, state, city, sales method, and product, featuring the median of operating profit. This structured dataset serves as a valuable resource for decision-making, enabling the company to make informed investments for optimal returns. By converting this data frame into a CSV file, the company gains the ability to closely monitor the median operating profit for all products across various regions, states, cities, and sales methods. This facilitates a strategic approach in assessing and enhancing profitability in diverse market segments.

Overall, the analysis is comprehensive and insightful, providing a solid foundation for understanding Adidas sales dynamics.

5. CONCLUSION

In conclusion, our Adidas Sales Project has provided valuable insights into the dynamics of the brand's performance in the USA market. Beginning with data cleaning and exploration, we ensured the foundation for our analysis was robust.

The sales analysis revealed key trends, identifying popular products and showcasing the evolution of total sales and operating profit over time. This phase set the stage for a comprehensive understanding of Adidas's market presence.

Profitability analysis deepened our insight into the financial performance of each product, offering a nuanced perspective on their contributions. The regional analysis brought geographical nuances to the forefront, emphasizing the impact of location on sales patterns.

Retailer analysis illuminated the distinctive contributions of each player, contributing to a more comprehensive understanding of Adidas's market partnerships. Finally, the pricing analysis delved into the strategic aspects, decoding the relationships between pricing metrics.

In essence, this project has been a systematic exploration, moving beyond the surface to uncover the intricacies of Adidas sales. Each phase added a layer of understanding, transforming raw data into meaningful insights. This comprehensive analysis not only informs our understanding of past performance but also lays a solid foundation for future strategic considerations in the dynamic landscape of the sportswear market.

6. Appendix

Complete code in detail

1. Importing necessary libraries and dataset

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
df = pd.read_csv('Adidas US Sales Datasets.csv')
df.head(10)
```

	Retailer	Retailer_ID	Invoice_Date	Region	State	City	Product	Price_per_Unit	Units_Sold	Total_Sales	Operating_Profit	Operating_Margin	Sales_Method
0	Foot Locker	1185732	01-01-2020	Northeast	New York	New York	Men's Street Footwear	$50.00	1,200	$6,00,000	$3,00,000	50%	In-store
1	Foot Locker	1185732	02-01-2020	Northeast	New York	New York	Men's Athletic Footwear	$50.00	1,000	$5,00,000	$1,50,000	30%	In-store
2	Foot Locker	1185732	03-01-2020	Northeast	New York	New York	Women's Street Footwear	$40.00	1,000	$4,00,000	$1,40,000	35%	In-store
3	Foot Locker	1185732	04-01-2020	Northeast	New York	New York	Women's Athletic Footwear	$45.00	850	$3,82,500	$1,33,875	35%	In-store
4	Foot Locker	1185732	05-01-2020	Northeast	New York	New York	Men's Apparel	$60.00	900	$5,40,000	$1,62,000	30%	In-store
5	Foot Locker	1185732	06-01-2020	Northeast	New York	New York	Women's Apparel	$50.00	1,000	$5,00,000	$1,25,000	25%	In-store
6	Foot Locker	1185732	07-01-2020	Northeast	New York	New York	Men's Street Footwear	$50.00	1,250	$6,25,000	$3,12,500	50%	In-store
7	Foot Locker	1185732	08-01-2020	Northeast	New York	New York	Men's Athletic Footwear	$50.00	900	$4,50,000	$1,35,000	30%	Outlet
8	Foot Locker	1185732	21-01-2020	Northeast	New York	New York	Women's Street Footwear	$40.00	950	$3,80,000	$1,33,000	35%	Outlet
9	Foot Locker	1185732	22-01-2020	Northeast	New York	New York	Women's Athletic Footwear	$45.00	825	$3,71,250	$1,29,938	35%	Outlet

2. Data inspection

```
df.shape
#There are 9648 rows and 13 columns
```

```
(9648, 13)
```

```
df.columns
#This gives us the list of columns in the  dataset
```

```
Index(['Retailer', 'Retailer_ID', 'Invoice_Date', 'Region', 'State', 'City',
       'Product', 'Price_per_Unit', 'Units_Sold', 'Total_Sales',
       'Operating_Profit', 'Operating_Margin', 'Sales_Method'],
      dtype='object')
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9648 entries, 0 to 9647
Data columns (total 13 columns):
 #   Column            Non-Null Count  Dtype 
---  ------            --------------  ----- 
 0   Retailer          9648 non-null   object
 1   Retailer_ID       9648 non-null   int64 
 2   Invoice_Date      9648 non-null   object
 3   Region            9648 non-null   object
 4   State             9648 non-null   object
 5   City              9648 non-null   object
 6   Product           9648 non-null   object
 7   Price_per_Unit    9648 non-null   object
 8   Units_Sold        9648 non-null   object
 9   Total_Sales       9648 non-null   object
 10  Operating_Profit  9648 non-null   object
 11  Operating_Margin  9648 non-null   object
 12  Sales_Method      9648 non-null   object
dtypes: int64(1), object(12)
memory usage: 980.0+ KB
```

3. Data cleaning (Data preprocessing)

```
# Ensure that the columns are of type string before trying to use string
methods on them
df['Price_per_Unit'] = df['Price_per_Unit'].astype(str).str.replace('$',
'').str.replace(',', '').astype(float)
df['Total_Sales'] = df['Total_Sales'].astype(str).str.replace('$',
'').str.replace(',', '').astype(float)
df['Operating_Profit'] = df['Operating_Profit'].astype(str).str.replace('$',
'').str.replace(',', '').astype(float)
df['Units_Sold'] = df['Units_Sold'].astype(str).str.replace(',',
'').astype(float)
df['Operating_Margin'] = df['Operating_Margin'].astype(str).str.replace('%',
'').astype(float)
df.head()
```

	Retailer	Retailer_ID	Invoice_Date	Region	State	City	Product	Price_per_Unit	Units_Sold	Total_Sales	Operating_Profit	Operating_Margin	Sales_Method
0	Foot Locker	1185732	01-01-2020	Northeast	New York	New York	Men's Street Footwear	50.0	1200.0	600000.0	300000.0	50.0	In-store
1	Foot Locker	1185732	02-01-2020	Northeast	New York	New York	Men's Athletic Footwear	50.0	1000.0	500000.0	150000.0	30.0	In-store
2	Foot Locker	1185732	03-01-2020	Northeast	New York	New York	Women's Street Footwear	40.0	1000.0	400000.0	140000.0	35.0	In-store
3	Foot Locker	1185732	04-01-2020	Northeast	New York	New York	Women's Athletic Footwear	45.0	850.0	382500.0	133875.0	35.0	In-store
4	Foot Locker	1185732	05-01-2020	Northeast	New York	New York	Men's Apparel	60.0	900.0	540000.0	162000.0	30.0	In-store

```
df_clean = df
df_clean['Total_Sales'] = df_clean['Price_per_Unit'] * df_clean['Units_Sold']
df_clean['Operating_Profit'] = df_clean['Total_Sales'] *
df_clean['Operating_Margin']/100
df_clean.head(3)
#The columns are now corrected and stored in a new dataframe called
'df_clean'
```

	Retailer	Retailer_ID	Invoice_Date	Region	State	City	Product	Price_per_Unit	Units_Sold	Total_Sales	Operating_Profit	Operating_Margin	Sales_Method
0	Foot Locker	1185732	01-01-2020	Northeast	New York	New York	Men's Street Footwear	50.0	1200.0	60000.0	30000.0	50.0	In-store
1	Foot Locker	1185732	02-01-2020	Northeast	New York	New York	Men's Athletic Footwear	50.0	1000.0	50000.0	15000.0	30.0	In-store
2	Foot Locker	1185732	03-01-2020	Northeast	New York	New York	Women's Street Footwear	40.0	1000.0	40000.0	14000.0	35.0	In-store

```
for column in df_clean.columns:
    if df_clean[column].dtype == 'O':
        df_clean[column] = df_clean[column].str.lower()
df_clean.head(3)
#Converting all strings to lower case
```

	Retailer	Retailer_ID	Invoice_Date	Region	State	City	Product	Price_per_Unit	Units_Sold	Total_Sales	Operating_Profit	Operating_Margin	Sales_Method
0	foot locker	1185732	01-01-2020	northeast	new york	new york	men's street footwear	50.0	1200.0	60000.0	30000.0	50.0	in-store
1	foot locker	1185732	02-01-2020	northeast	new york	new york	men's athletic footwear	50.0	1000.0	50000.0	15000.0	30.0	in-store
2	foot locker	1185732	03-01-2020	northeast	new york	new york	women's street footwear	40.0	1000.0	40000.0	14000.0	35.0	in-store

```
df_clean['Invoice_Date'] = pd.to_datetime(df_clean['Invoice_Date'],
format="%d-%m-%Y")
#Converting Invoice Date to datetime format
```

```
df_clean.info()
#The new dataframe has all the columns of proper data types
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9648 entries, 0 to 9647
Data columns (total 13 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   Retailer          9648 non-null   object
 1   Retailer_ID       9648 non-null   int64
 2   Invoice_Date      9648 non-null   datetime64[ns]
 3   Region            9648 non-null   object
 4   State             9648 non-null   object
 5   City              9648 non-null   object
 6   Product           9648 non-null   object
 7   Price_per_Unit    9648 non-null   float64
 8   Units_Sold        9648 non-null   float64
 9   Total_Sales       9648 non-null   float64
 10  Operating_Profit  9648 non-null   float64
 11  Operating_Margin  9648 non-null   float64
 12  Sales_Method      9648 non-null   object
dtypes: datetime64[ns](1), float64(5), int64(1), object(6)
memory usage: 980.0+ KB
```

```
df_clean.drop_duplicates(inplace = True)
df_clean.shape
#There are No Duplicate Rows in the Dataset
```

```
(9648, 13)
```

```
df_clean.columns.value_counts()
#There are No Duplicate Columns in the Dataset
```

```
City                1
State               1
Retailer            1
Price_per_Unit      1
Region              1
Retailer_ID         1
Operating_Profit    1
Total_Sales         1
Invoice_Date        1
Product             1
Units_Sold          1
Operating_Margin    1
Sales_Method        1
dtype: int64
```

```
df_clean.isnull().sum()
#There are no null values in the data set
```

```
Retailer            0
Retailer_ID         0
Invoice_Date        0
Region              0
State               0
City                0
Product             0
Price_per_Unit      0
Units_Sold          0
Total_Sales         0
Operating_Profit    0
Operating_Margin    0
Sales_Method        0
dtype: int64
```

4. Exploratory Data Analysis

```
df_clean.describe()
```

	Retailer_ID	Price_per_Unit	Units_Sold	Total_Sales	Operating_Profit	Operating_Margin
count	9.648000e+03	9648.000000	9648.000000	9648.000000	9648.000000	9648.000000
mean	1.173850e+06	45.216625	256.930037	12455.083955	4894.793501	42.299129
std	2.636038e+04	14.705397	214.252030	12716.392111	4866.464372	9.719742
min	1.128299e+06	7.000000	0.000000	0.000000	0.000000	10.000000
25%	1.185732e+06	35.000000	106.000000	4065.250000	1753.440000	35.000000
50%	1.185732e+06	45.000000	176.000000	7803.500000	3262.980000	41.000000
75%	1.185732e+06	55.000000	350.000000	15864.500000	6192.360000	49.000000
max	1.197831e+06	110.000000	1275.000000	82500.000000	39000.000000	80.000000

```
range = ['Price_per_Unit','Units_Sold', 'Total_Sales', 'Operating_Profit',
'Operating_Margin']
for r in range:
    print("range of ", r , " : ", df_clean[r].max() - df_clean[r].min())
#The range of numerical columns are as follows
```

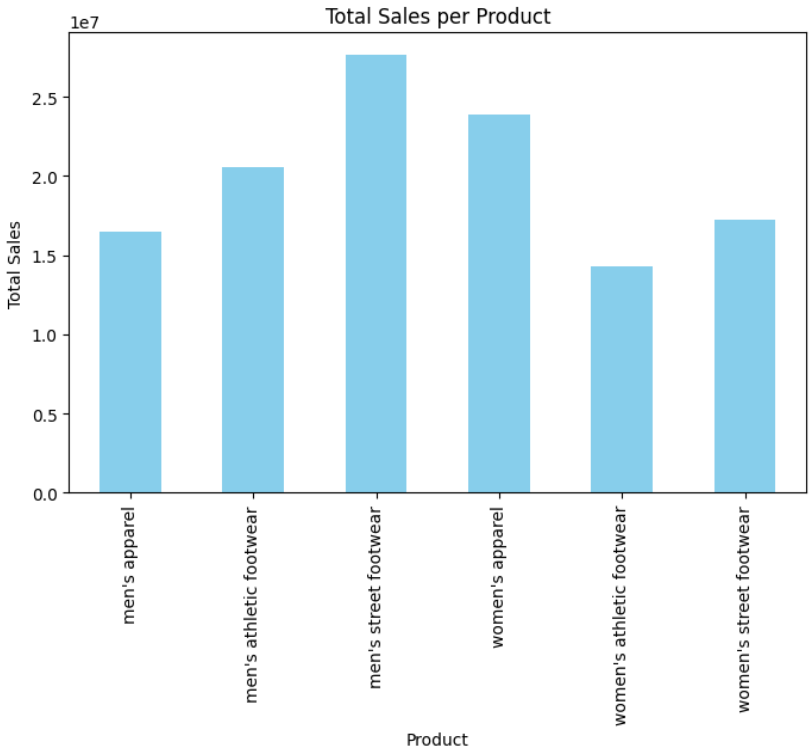
```
range of  Price_per_Unit  :  103.0
range of  Units_Sold  :  1275.0
range of  Total_Sales  :  82500.0
range of  Operating_Profit  :  39000.0
range of  Operating_Margin  :  70.0
```

Sales Analysis

```
df_clean['Product'].value_counts().idxmax()
# The most listed product is 'men's athletic footwear'
```

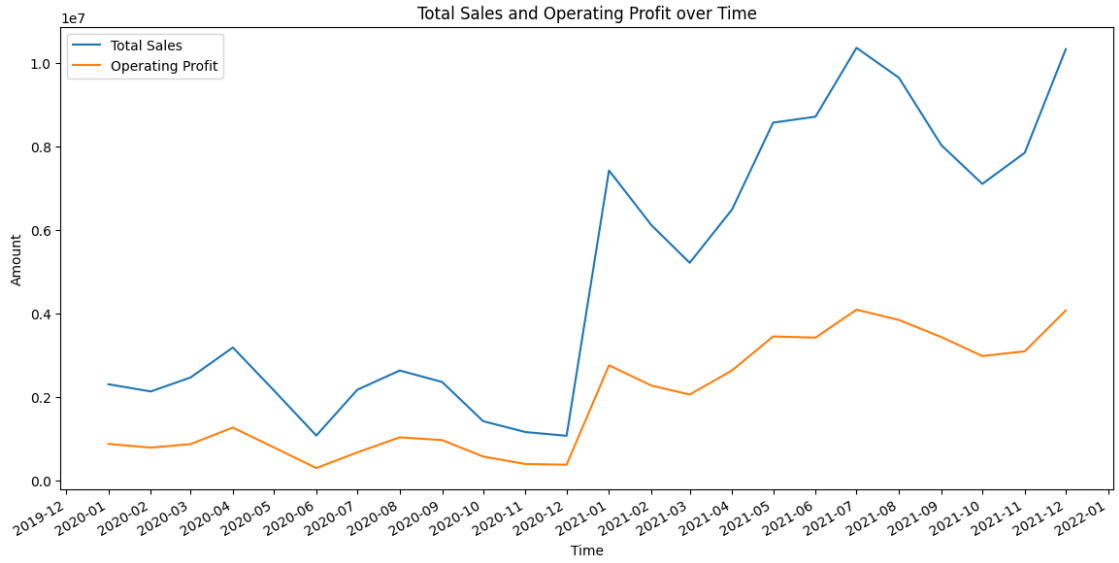
```
"men's street footwear"
```

```
product_sales = df_clean.groupby('Product')['Total_Sales'].sum()

plt.figure(figsize=(8, 5))
product_sales.plot(kind='bar', color='skyblue')
plt.xlabel('Product')
plt.ylabel('Total Sales')
plt.title('Total Sales per Product')
plt.show()
# Men's Street Footwear' has the highest total sales
```




```
import matplotlib.dates as mdates

df_clean['Year'] = df_clean['Invoice_Date'].dt.year
df_clean['Month'] = df_clean['Invoice_Date'].dt.month
monthly_data = df_clean.groupby(['Year', 'Month']).agg({
    'Total_Sales': 'sum',
    'Operating_Profit': 'sum'
}).reset_index()
monthly_data['Date'] = pd.to_datetime(monthly_data[['Year',
'Month']].assign(day=1))
```

```
plt.figure(figsize=(14, 7))

plt.plot(monthly_data['Date'], monthly_data['Total_Sales'], label='Total
Sales')
plt.plot(monthly_data['Date'], monthly_data['Operating_Profit'],
label='Operating Profit')

plt.gca().xaxis.set_major_formatter(mdates.DateFormatter('%Y-%m'))
plt.gca().xaxis.set_major_locator(mdates.MonthLocator())
plt.gcf().autofmt_xdate() # for the x date angle

plt.title("Total Sales and Operating Profit over Time")
plt.xlabel('Time')
plt.ylabel('Amount')
plt.legend()

plt.show()
#The below graph shows trend of total sales and operating profit over time
#There is a gradual increase in total sales and operating profit over time.
#During dec 2020 to jan 2021, there is a sudden increase in total sales and
operating profit but the point to note is that the extent of increase of
profit is lesser than that of sales.
```




```
print("Total sales: ", df_clean.groupby(['Sales_Method']).Total_Sales.sum())
print("Total profit: ",
df_clean.groupby(['Sales_Method']).Operating_Profit.sum())
# Out of different sales methods, 'online' has the highest total sales and
profit whereas, 'in-store' has the lowest total sales and profit.
```

```
Total sales:  Sales_Method
in-store    35664375.0
```

```
online      44965657.0
outlet      39536618.0
Name: Total_Sales, dtype: float64
Total profit:  Sales_Method
in-store    12759128.75
online      19552537.72
outlet      14913301.23
Name: Operating_Profit, dtype: float64
```

```
df_clean.groupby(['Product'])['Units_Sold'].mean()
#The below lists the average number of units sold per product per day. That
is, the average transaction per day.
# men's street footwear has the highest transaction per day whereas, men's
apparel has the least.
```

```
Product
men's apparel                190.960772
men's athletic footwear      270.513043
men's street footwear        368.521739
women's apparel              269.792910
women's athletic footwear    197.531756
women's street footwear      243.948383
Name: Units_Sold, dtype: float64
```

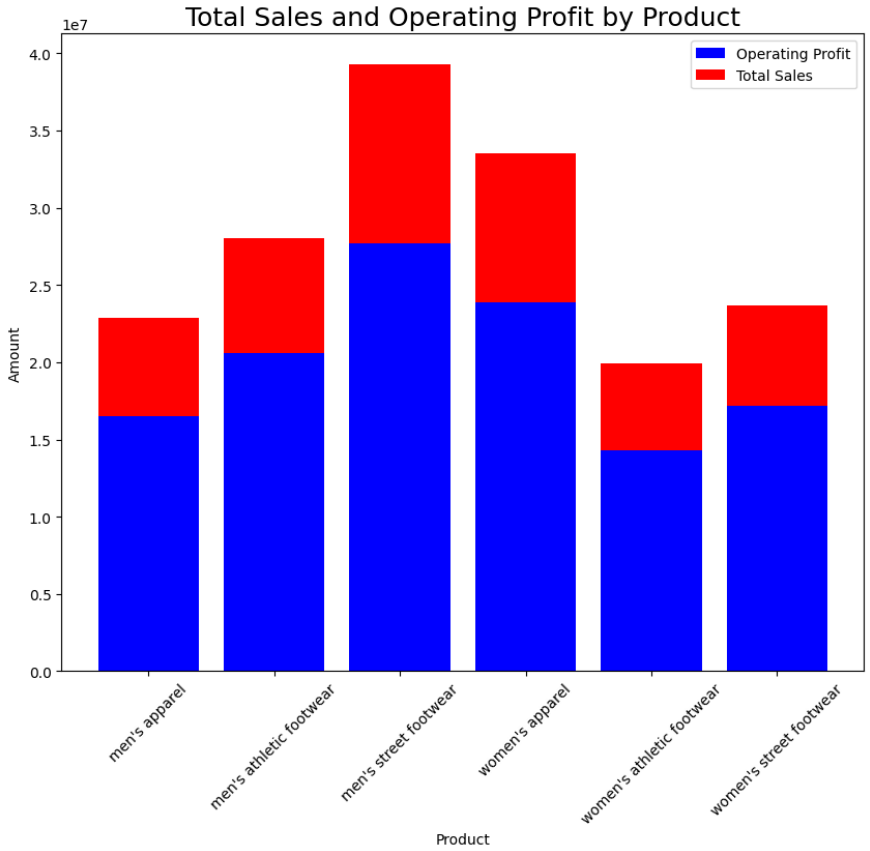
```
df_clean.groupby(['Sales_Method'])['Units_Sold'].std()
#The belows lists the amount of units sold variation for each sales method
#It is clear that 'outlet' has the highest variation which means that the
number of units sold every day differs a lot, while 'online' has the lowest
variation
```

```
Sales_Method
in-store    203.410458
online      176.269773
outlet      232.193750
Name: Units_Sold, dtype: float64
```

```
df_clean.groupby(['Product'])['Total_Sales'].std()
#These values represent the spread or variability of 'Total_Sales' within
each product category.
# A higher standard deviation indicates greater variability in sales within
that product category.
```

```
Product
men's apparel                10783.845485
men's athletic footwear      12707.987489
men's street footwear        14978.931461
women's apparel              14199.454793
women's athletic footwear     9630.473858
women's street footwear      11196.643353
Name: Total_Sales, dtype: float64
```

```
product_sales_profit = df_clean.groupby('Product')[['Total_Sales',
'Operating_Profit']].sum()

plt.figure(figsize=[10,8])
plt.bar(product_sales_profit.index, product_sales_profit['Total_Sales'],
color='blue')
plt.bar(product_sales_profit.index, product_sales_profit['Operating_Profit'],
bottom=product_sales_profit['Total_Sales'], color='red')

plt.title("Total Sales and Operating Profit by Product", fontsize=18)
plt.xlabel("Product")
plt.ylabel("Amount")
plt.legend(["Operating Profit", "Total Sales"])
plt.xticks(rotation=45)
plt.show()
#This visualization shows operating profit and total salesby product.
#It is clear that men's street footwear has the highest total sales and
operating profit.
```



Profitability Analysis

```
df_clean.groupby(['Product'])['Operating_Profit'].median()
#Considering median, there is more profit in men's street footwear and least
in women's athletic footwear.
```

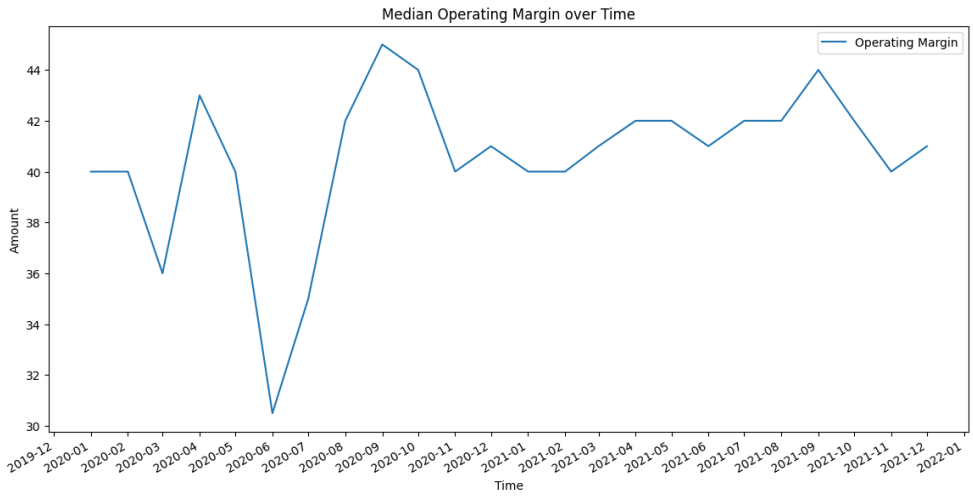
```
Product
men's apparel                2679.415
men's athletic footwear      3293.760
men's street footwear        5201.500
women's apparel              4004.200
women's athletic footwear    2357.100
women's street footwear      2703.000
Name: Operating_Profit, dtype: float64
```

```
df_clean.groupby(['Product'])['Operating_Margin'].median()
#men's street footwear has the highest operating margin . And the general
operating margin is 40%.
```

```
Product
men's apparel                40.0
men's athletic footwear      40.0
men's street footwear        45.0
women's apparel              44.0
women's athletic footwear    41.0
women's street footwear      40.0
Name: Operating_Margin, dtype: float64
```

```
import matplotlib.dates as mdates

df_clean['Year'] = df_clean['Invoice_Date'].dt.year
df_clean['Month'] = df_clean['Invoice_Date'].dt.month
monthly_data = df_clean.groupby(['Year', 'Month']).agg({
    'Operating_Margin': 'median'
}).reset_index()
monthly_data['Date'] = pd.to_datetime(monthly_data[['Year',
'Month']].assign(day=1))

plt.figure(figsize=(14, 7))
plt.plot(monthly_data['Date'], monthly_data['Operating_Margin'],
label='Operating Margin')

plt.gca().xaxis.set_major_formatter(mdates.DateFormatter('%Y-%m'))
plt.gca().xaxis.set_major_locator(mdates.MonthLocator())
plt.gcf().autofmt_xdate() # for the x date angle

plt.title("Median Operating Margin over Time")
plt.xlabel('Time')
plt.ylabel('Amount')
plt.legend()
```

```
plt.show()
#The below graph shows trend of median operating margin over time.
#It is clear that the median operating margin is constantly changing over
time and was the least in june 2020.
#The highest median operating margin was in sept 2020
```




```
# Filter the dataframe for June 2020 and sales method as online
filtered_data = df_clean[(df_clean['Year'] == 2020) & (df_clean['Month'] ==
6) & (df_clean['Sales_Method'] == 'online')]

# Calculate total sales
total_sales = filtered_data['Total_Sales'].sum()
total_sales
#The total sales that happened via online in June 2020 is 223,569 units.
```

```
223569.0
```

```
filtered_data = df_clean[(df_clean['Year'] == 2020) & (df_clean['Month'] ==
9) & (df_clean['Sales_Method'] == 'online')]

# Calculate total sales
total_sales = filtered_data['Total_Sales'].sum()
total_sales
```

```
456214.0
```

```
# Calculate total sales for June 2020
total_sales_June2020 = df_clean[(df_clean['Year'] == 2020) &
(df_clean['Month'] == 6)]['Total_Sales'].sum()

# Calculate percentage of online sales for June 2020
percentage_online_sales_June2020 = (total_sales / total_sales_June2020) * 100
percentage_online_sales_June2020
#The percentage of sales via online in June 2020 is approximately 20.62%.
```

```
42.078631683997514
```

```
total_sales_June2020 = df_clean[(df_clean['Year'] == 2020) &
(df_clean['Month'] == 9)]['Total_Sales'].sum()

# Calculate percentage of online sales for Sept 2020
percentage_online_sales_June2020 = (total_sales / total_sales_June2020) * 100
percentage_online_sales_June2020
```

```
19.268348928025084
```

```
print("Kurtosis : ", df_clean['Operating_Profit'].kurt())
print("Skewness : ", df_clean['Operating_Profit'].skew())
#The positive kurtosis suggests that the distribution has heavier tails and a
sharper peak compared to a normal distribution.
#The positive skewness suggests that the distribution is right-skewed,
meaning it is stretched more to the right.
```

```
Kurtosis :  7.181164174449948
Skewness :  2.334590765903441
```

```
df_clean.groupby(['Sales_Method'])['Operating_Margin'].skew()
#The operating margin for each sales method is right skewed or positively
skewed. And in-store has the highest skewness whereas, outlet has the lowest
skewness.
#This means that the right tail is longer and fatter than the left tail.
```

```
Sales_Method
in-store    0.363159
online      0.251876
outlet      0.148255
Name: Operating_Margin, dtype: float64
```

Regional Analysis

```
print("Region wise: ",df_clean.groupby(['Region'])['Total_Sales'].sum())
print("State wise: ",
df_clean.groupby(['State'])['Total_Sales'].sum().idxmax())
print("City wise: ",
df_clean.groupby(['City'])['Total_Sales'].sum().idxmax())
#In general the western region has highest sales. The State with highest
sales is New York.
```

```
Region wise:  Region
midwest      16674434.0
northeast    25078267.0
south        20603356.0
southeast    21374436.0
west         36436157.0
Name: Total_Sales, dtype: float64
```

```
State wise:  new york
City wise:  new york
```

```
print(df_clean.groupby(['Region'])['Product'].value_counts().idxmax())
print(df_clean.groupby(['State'])['Product'].value_counts().idxmax())
print(df_clean.groupby(['City'])['Product'].value_counts().idxmax())
#The following gives the overall most popular Product in each of region,
state and city division.
#men's apparel is the most popular , city and state wise but on a bigger
picture, men's athletic footwear is the most popular product in western
region.
#Note: Here popularity is taken as the number of times a particular product
is listed and not number of products sold.
```

```
('west', "men's athletic footwear")
('california', "men's apparel")
('portland', "men's apparel")
```

```
print("Region wise: ", df_clean.groupby(['Region',
'Product'])['Units_Sold'].sum().idxmax())
print("State wise: ", df_clean.groupby(['State',
'Product'])['Units_Sold'].sum().idxmax())
print("City wise: ", df_clean.groupby(['City',
'Product'])['Units_Sold'].sum().idxmax())
#specific regional, state, city preferences for certain products are listed
below. In each of them, men's street footwear is more preferred.
```

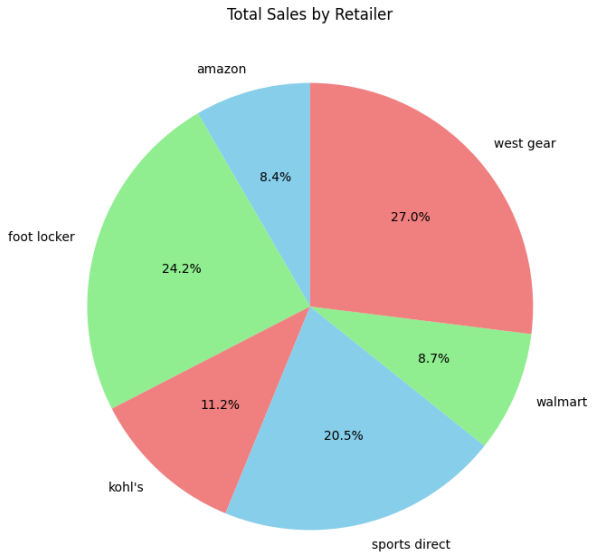
```
Region wise:  ('west', "men's street footwear")
State wise:  ('new york', "men's street footwear")
City wise:  ('charleston', "men's street footwear")
```

```
df_clean.groupby(['Region', 'Sales_Method'])['Operating_Profit'].sum()
#This tells us the best sales method for each region for max profit.
```

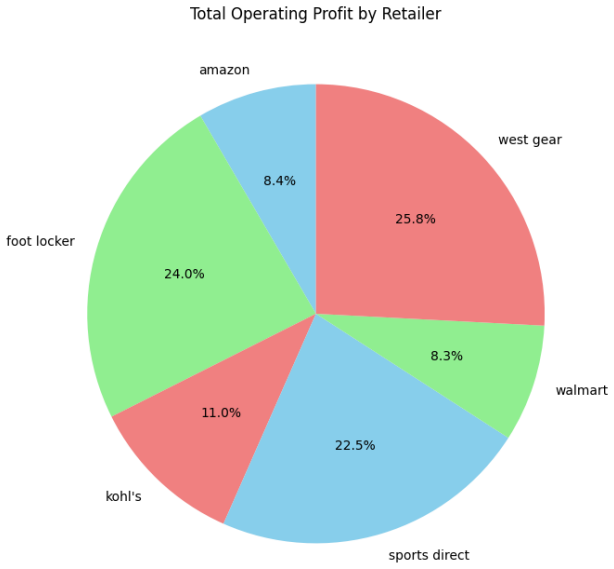
```
Region     Sales_Method
midwest    in-store        2316565.00
           online          3133263.98
           outlet          1410116.25
northeast  in-store        4254420.00
           online          2246831.65
           outlet          3231522.25
south      in-store         134800.00
           online          4149888.22
           outlet          4936917.10
southeast  in-store        2558256.25
           online          5080401.63
           outlet           754401.32
west       in-store        3495087.50
           online          4942152.24
           outlet          4580344.31
Name: Operating_Profit, dtype: float64
```

Retailer Analysis

```
sales_by_location = df_clean.groupby('Retailer')['Total_Sales'].sum()

# Plotting a pie chart for total sales by retailer
plt.figure(figsize=(8, 8))
sales_by_location.plot.pie(autopct='%1.1f%%', startangle=90,
colors=['skyblue', 'lightgreen', 'lightcoral'])
plt.title('Total Sales by Retailer')
plt.ylabel('')
plt.show()
#The total sales is highest by the retailer - west gear followed by foot
locker.
#Even though online mode has the highest sales, amazon has the least total
sales.
```




```
sales_by_location = df_clean.groupby('Retailer')['Operating_Profit'].sum()

# Plotting a pie chart for operating profit by retailer
plt.figure(figsize=(8, 8))
sales_by_location.plot.pie(autopct='%1.1f%%', startangle=90,
colors=['skyblue', 'lightgreen', 'lightcoral'])
plt.title('Total Operating Profit by Retailer')
plt.ylabel('')
plt.show()
#The operating profit is highest by the retailer - west gear followed by foot
locker.
#Even though online mode has the highest operating profit, amazon has the
least operating profit.
```




```
df_clean.groupby(['Retailer', 'Sales_Method'])['Operating_Profit'].mean()
# The below lists the average operating profit by each retailer.
#Even though west gear had the highest total operating profit, walmart
outperformed them in terms of average operational profits.
# Even though online mode had the overall highest operating profit,
considering each retailer's sales method, in-store had the highest operating
profit.
```

```
Retailer       Sales_Method
amazon         in-store         7083.972458
               online           3773.055296
               outlet           3818.121821
foot locker    in-store         6256.570156
               online           3717.388738
               outlet           4189.243405
kohl's         in-store         7359.071181
               online           3831.638194
               outlet           6179.129774
sports direct  in-store         7035.712457
               online           4542.114372
               outlet           5457.985430
walmart        in-store        13325.337838
               online           4781.102743
               outlet           6753.334784
west gear      in-store         7868.115165
               online           3859.311032
               outlet           4260.573448
Name: Operating_Profit, dtype: float64
```

```
df_clean.groupby('Retailer')['Product'].value_counts()
#The below lists the most popular products sold by each retailer. This gives
```

```
us an idea about how each retailer is making more profits through their
products.
```

```
Retailer       Product
amazon         men's athletic footwear      159
               men's street footwear        159
               women's apparel              159
               women's athletic footwear    158
               men's apparel                157
               women's street footwear      157
foot locker    men's street footwear        449
               men's athletic footwear      442
               women's athletic footwear    442
               women's street footwear      438
               men's apparel                433
               women's apparel              433
kohl's         men's athletic footwear      172
               men's street footwear        172
               women's apparel              172
               women's street footwear      172
               men's apparel                171
               women's athletic footwear    171
sports direct  women's street footwear      342
               women's apparel              341
               men's apparel                339
               women's athletic footwear    338
               men's athletic footwear      337
               men's street footwear        335
walmart        men's apparel                113
               women's apparel              107
               men's athletic footwear      104
               women's athletic footwear    102
               men's street footwear        101
               women's street footwear       99
west gear      women's street footwear      400
               men's athletic footwear      396
               women's apparel              396
               women's athletic footwear    395
               men's street footwear        394
               men's apparel                393
Name: Product, dtype: int64
```

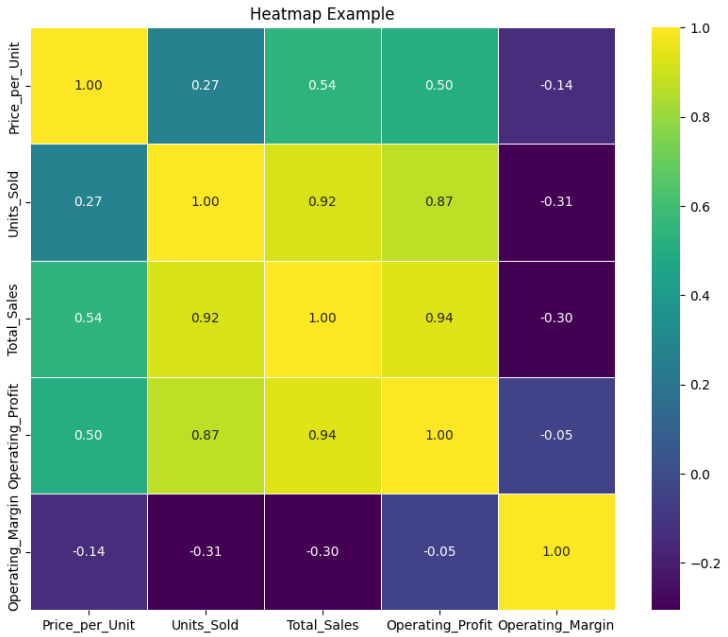
Pricing Analysis

```
df_clean[['Price_per_Unit','Units_Sold', 'Total_Sales', 'Operating_Profit',
'Operating_Margin']].corr()
# If Price per unit is more, the total sales & operating profit will be more
since they are positively correlated.
# Whereas, price per unit and operating margin has negative weak correlation.
# Price per unit and units sold are positively weakly correlated. This
```

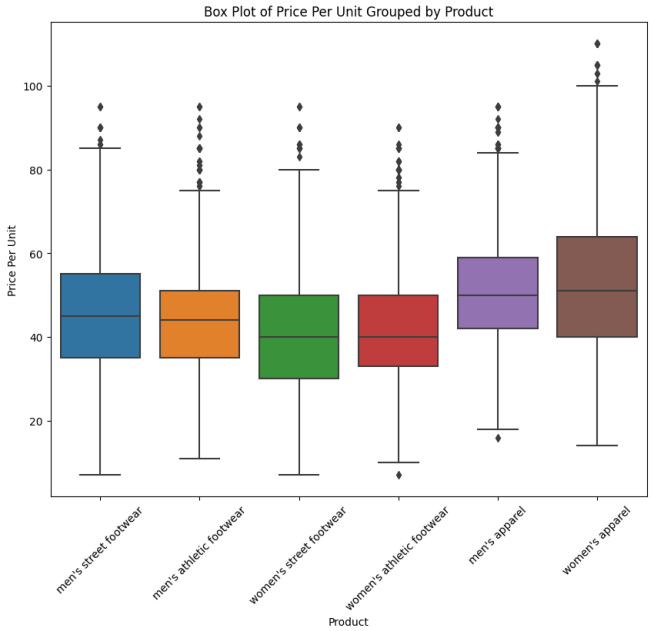
```
wouldn't guarantee that more price per unit would mean more units sold to
some extent.
# units sold and operating profit have positive strong correlation which is
by logic correct and same with units sold and total sales.
# Interestingly, operating margin and units sold are negatively moderately
correlated.
# As operating margin and unit sold is negatively moderately correlated, so
we can say that the most sales/more units would be sold due to excess money
spent on ads and hence we could justify that maybe the sales are high but the
profit margin is comparatively less.
```

	Price_per_Unit	Units_Sold	Total_Sales	Operating_Profit	Operating_Margin
Price_per_Unit	1.000000	0.265869	0.539547	0.503683	-0.137486
Units_Sold	0.265869	1.000000	0.919339	0.871993	-0.305479
Total_Sales	0.539547	0.919339	1.000000	0.935372	-0.302295
Operating_Profit	0.503683	0.871993	0.935372	1.000000	-0.047491
Operating_Margin	-0.137486	-0.305479	-0.302295	-0.047491	1.000000

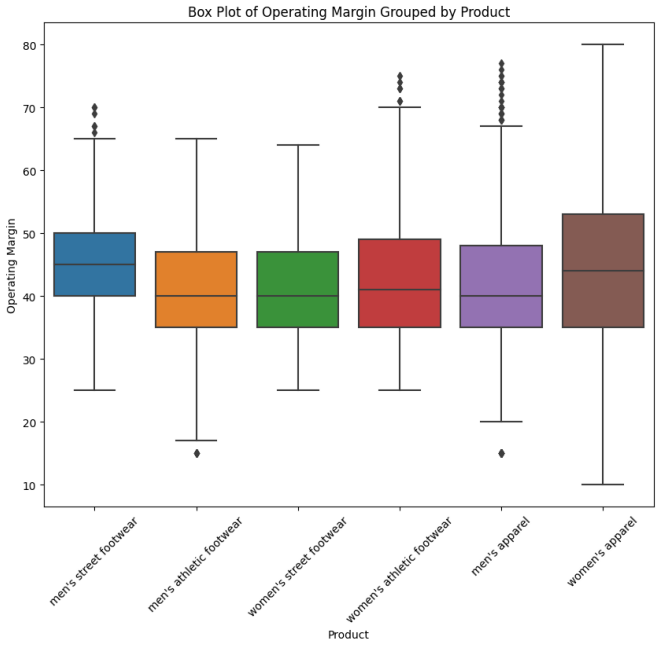
```
heatmap = df_clean[['Price_per_Unit','Units_Sold', 'Total_Sales',
'Operating_Profit', 'Operating_Margin']]
plt.figure(figsize=(10, 8))
sns.heatmap(heatmap.corr(), cmap='viridis', annot=True, fmt=".2f",
linewidths=.5)
plt.title('Heatmap')
plt.show()
```



```
plt.figure(figsize=(10, 8))
sns.boxplot(x='Product', y='Price_per_Unit', data=df_clean)
plt.title('Box Plot of Price Per Unit Grouped by Product')
plt.xlabel('Product')
plt.ylabel('Price Per Unit')
plt.xticks(rotation=45)
plt.show()
#Generally, price per unit for each product has normal distribution but for "men's athletic footwear" it is slightly left skewed
# and for "women's athletic footwear" it is slightly right skewed.

#The median price per unit for "men's apparel" and "women's apparel" is the highest.
#"women's street footwear" and "women's athletic footwear" has the lowest median price per unit.
#There are many outliers towards the right side of the graph for each product.
```




```
plt.figure(figsize=(10, 8))
sns.boxplot(x='Product', y='Operating_Margin', data=df_clean)
plt.title('Box Plot of Operating Margin Grouped by Product')
plt.xlabel('Product')
plt.ylabel('Operating Margin')
plt.xticks(rotation=45)
plt.show()
#In general, the operating margin for each product has slight right skewness,
# but for "men's street footwear" and "women's apparel" it is normal distributed.
```




```
final_table = df_clean.groupby(['Region', 'State',
'City','Sales_Method','Product']).Operating_Profit.median()
final_table = final_table.sort_values(ascending=False)
final_table.to_csv('Final_Profit_dataset.csv')

#This creates a new csv file which groups region, state, city, sales method
and product based on median operating profits which helps the company make
better decisions while investing to gain maximum profits.
```