



DATA ANALYST ASSIGNMENT

CUSTOMER & OPERATIONS ANALYSIS

[PROJECT DRIVE LINK](#)

PROBLEM STATEMENT

1. Excel Assignment: Sales & Inventory Dashboard

- **Objective:** Analyze sales, inventory levels, and out-of-stock patterns.

2. SQL Assignment: Customer Purchase & Delivery Analysis

- **Objective:** Understand customer retention, purchasing trends, and delivery performance.

3. Python Assignment: Customer Segmentation & Demand Patterns

- **Objective:** Segment customers based on purchasing behavior and detect demand trends.

4. Business Case Study: Discount Impact Analysis

- **Objective:** Analyze the impact of discount strategies on profitability and retention.

1. Excel Assignment: Sales & Inventory Dashboard - ([Project link](#))

Objective

- Analyze sales, inventory levels, and out-of-stock patterns.

Tasks

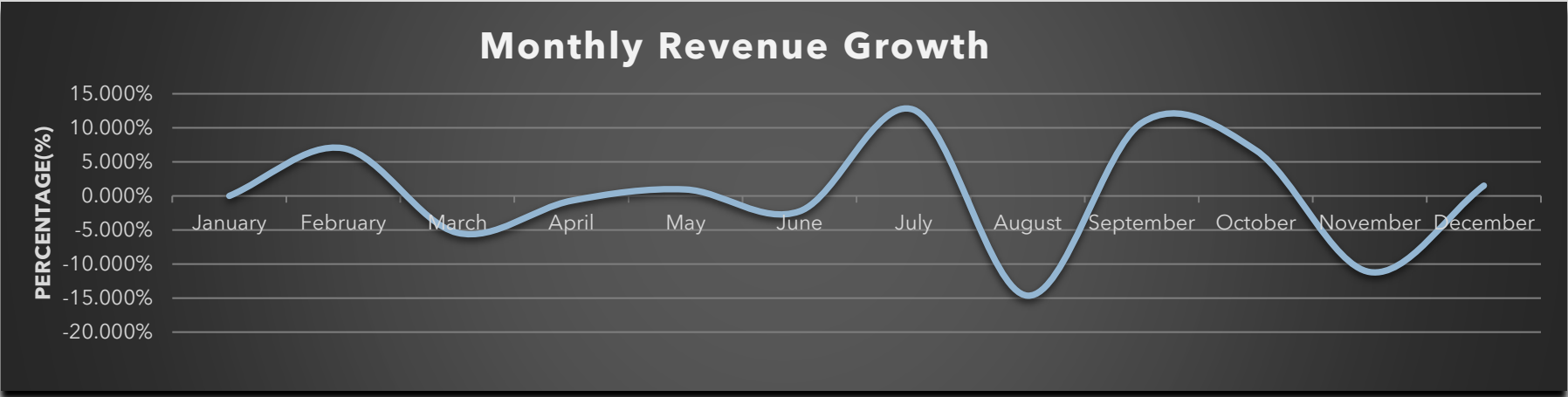
- A. Calculate **monthly revenue growth**, **average order value**, and **customer retention rate**.
- B. Identify **top-selling categories** and **most returned products**.
- C. Analyze **inventory levels** to find **products frequently out of stock**.
- D. Create a **dashboard** displaying revenue trends, product demand, and stock levels.

Dataset: sales_data.csv, inventory.csv

[1(A)] Calculate **monthly revenue growth**, **average order value**, and **customer retention rate**.

[a(i)] **Monthly Revenue Growth**

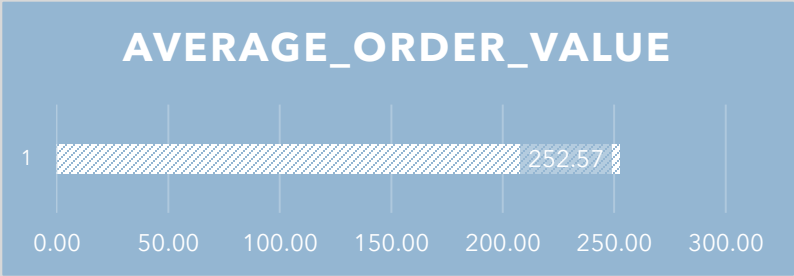
Month	Total_Revanue_Per_Month	Revanue_Growth
January	102152.91	0.000%
February	109304.17	7.001%
March	103387.82	-5.413%
April	102659.68	-0.704%
May	103633.13	0.948%
June	101293.64	-2.257%
July	114113.43	12.66%
August	97395.46	-14.65%
September	107841.21	10.73%
October	115082.64	6.71%
November	102224.75	-11.17%
December	103771.55	1.513%



◦ [a(ii)] **Calculate Average order value**

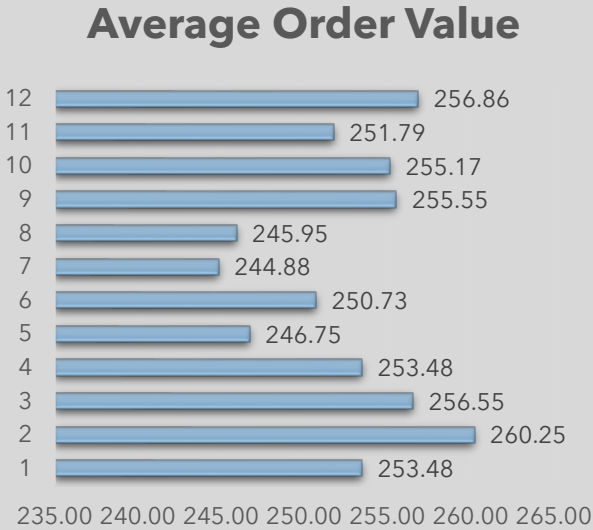
Yearly Average Order Value

Count_of_Total_Order	Total_Revenue	Average_Order_Value
5000	1262860.39	252.57



Monthly Average Order Value

Row Labels	Sum of revenue	Count of Month_Name	Average Order Value
January	102152.91	403	253.48
February	109304.17	420	260.25
March	103387.82	403	256.55
April	102659.68	405	253.48
May	103633.13	420	246.75
June	101293.64	404	250.73
July	114113.43	466	244.88
August	97395.46	396	245.95
September	107841.21	422	255.55
October	115082.64	451	255.17
November	102224.75	406	251.79
December	103771.55	404	256.86

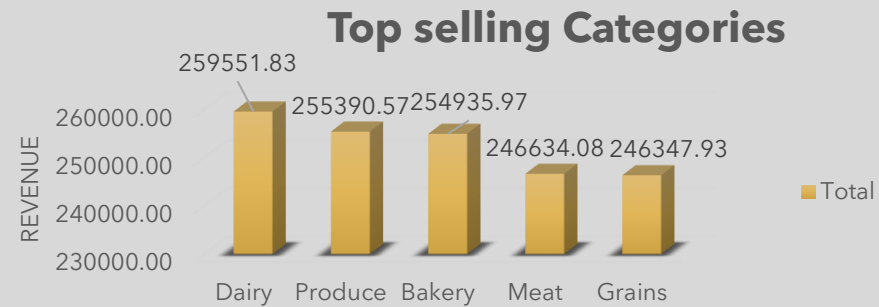


[1(B)] Identify **top-selling categories** and **most returned products**.

[b(i)] **Identify Top-selling categories**

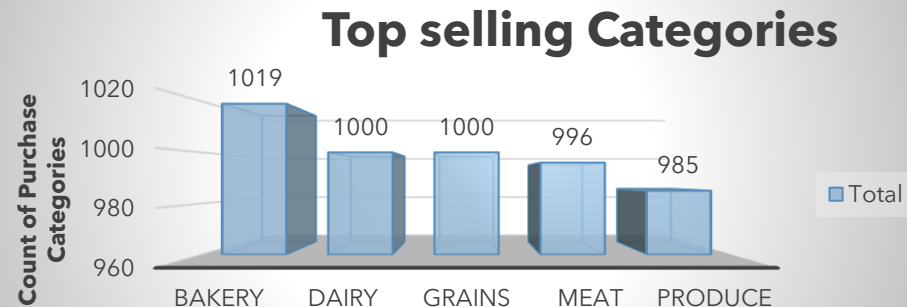
Revenue wise Top-selling Categories

Row Labels	Sum of revenue
Dairy	259551.83
Produce	255390.57
Bakery	254935.97
Meat	246634.08
Grains	246347.93



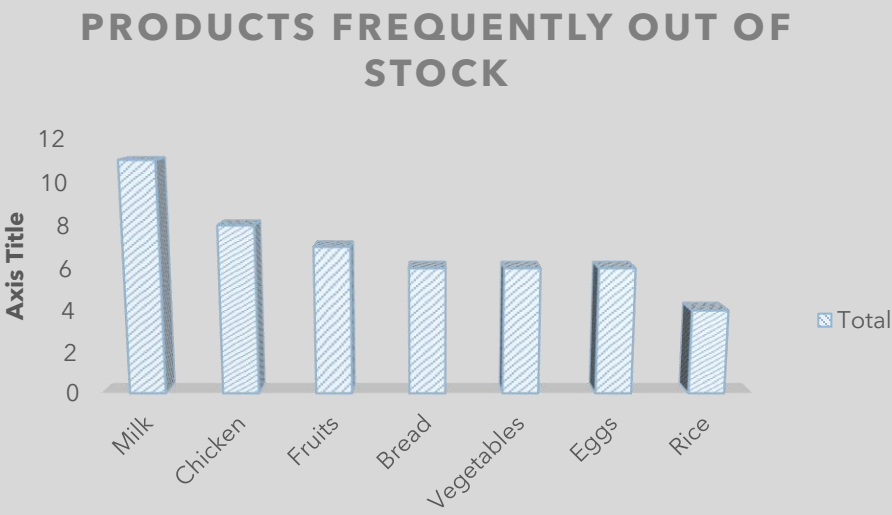
Frequently purchase wise Top-selling Categories

Row Labels	Count of category
Bakery	1019
Dairy	1000
Grains	1000
Meat	996
Produce	985



[1(C)] Analyze **inventory levels** to find **products frequently out of stock**.

out_of_stock	Yes
Row Labels	Count of out_of_stock
Milk	11
Chicken	8
Fruits	7
Bread	6
Vegetables	6
Eggs	6
Rice	4



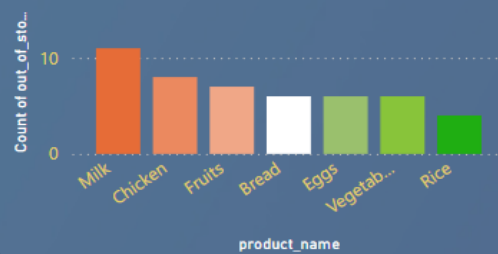
[1(D)] Create a **dashboard** displaying revenue trends, product demand, and stock levels. - [\(Project link\)](#)

Data Analyst Assignment: Customer & Operations Analysis

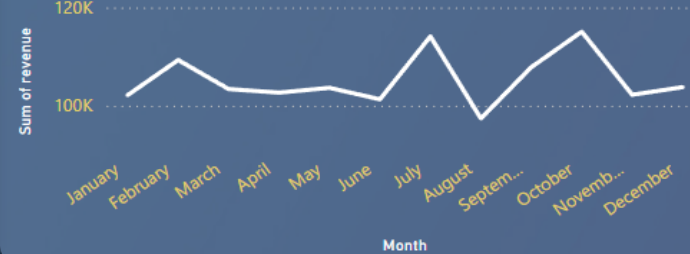
region

Central East North South West

Product Frequently Out of Stock



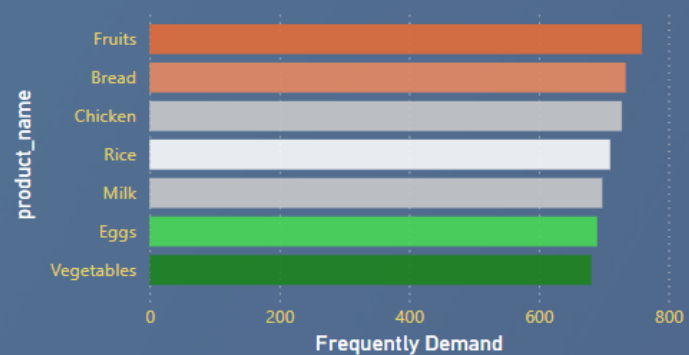
Revenue Trend by Month



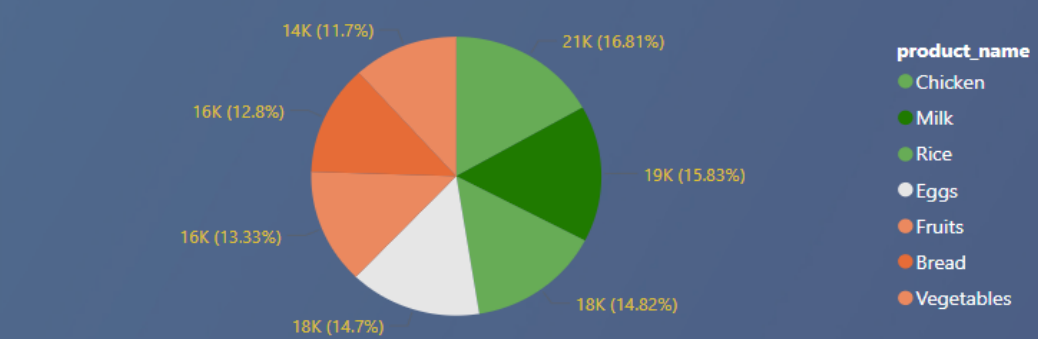
Top selling Category



Product Demand



Product Stock_level



2. SQL Assignment: Customer Purchase & Delivery Analysis - ([Project link](#))

Objective

- Understand customer retention, purchasing trends, and delivery performance.

Tasks

- A. Identify customers who **haven't placed an order in the last 60 days** but had at least 2 orders before.
 - B. Calculate the **average time between consecutive orders** for repeat customers.
 - C. Determine the **top 10% of customers by total spend** and their average order value.
 - D. Analyze **delivery time efficiency** by calculating the percentage of on-time deliveries per region.
- **Dataset:** orders.csv, delivery_performance.csv

[2(A)] Identify customers who **haven't placed an order in the last 60 days** but had at least 2 orders before.

- SQL Query(MS SQL Server)

```
SELECT distinct(customer_id)
FROM Orders
WHERE order_date < DATEADD(DAY, -60, GETDATE())
GROUP BY customer_id
HAVING COUNT(order_id) >= 2;
```

Sample Output

	customer_id
1	1257
2	1921
3	1589
4	1898
5	1234
6	1566
7	1804
8	1211
9	1374
10	1472
11	1543
12	1875
13	1042
14	1065
15	1397

[2(B)] Calculate the **average time between consecutive orders** for repeat customers.

◦ SQL Query(MS SQL Server)

```
WITH Order_Differences AS (  
    SELECT  
        customer_id,  
        order_id,  
        order_date,  
        LEAD(order_date) OVER (PARTITION BY customer_id ORDER BY order_date) AS next_order_date  
    FROM Orders  
)  
SELECT  
    customer_id,  
    AVG(DATEDIFF(DAY, order_date, next_order_date)) AS avg_days_between_orders  
FROM Order_Differences  
WHERE next_order_date IS NOT NULL  
GROUP BY customer_id;
```

Sample Output

	customer_id	avg_days_between_orders
1	1001	45
2	1002	225
3	1003	141
4	1005	24
5	1006	95
6	1007	133
7	1008	109
8	1009	30
9	1010	28
10	1011	49
11	1012	35
12	1013	78
13	1014	46
14	1015	54
15	1016	50

[2(C)] Determine the **top 10% of customers by total spend** and their average order value.

SQL Query(MS SQL Server)

```
WITH Customer_Spend AS (
    SELECT
        customer_id,
        SUM(total_amount) AS total_spend,
        COUNT(order_id) AS total_orders
    FROM Orders
    GROUP BY customer_id
),
Top_10_Percent_Customers AS (
    SELECT
        customer_id,
        total_spend,
        total_orders,
        PERCENT_RANK() OVER (ORDER BY total_spend DESC) AS spend_percentile
    FROM Customer_Spend
)
SELECT
    customer_id,
    total_spend,
    total_orders,
    total_spend / total_orders AS avg_order_value
FROM Top_10_Percent_Customers
WHERE spend_percentile <= 0.1
ORDER BY total_spend DESC;
```

Sample Output

	customer_id	total_spend	total_orders	avg_order_value
1	1161	3814.54170227051	13	293.426284790039
2	1770	3528.25869750977	11	320.750790682706
3	1578	3473.93367385864	11	315.812152168967
4	1200	3331.48054122925	11	302.861867384477
5	1916	3161.9059753418	9	351.322886149089
6	1118	3059.85415649414	10	305.985415649414
7	1661	3055.11183166504	9	339.456870185004
8	1678	3031.608959198	10	303.1608959198
9	1304	3027.02278137207	9	336.335864596897
10	1806	3016.5923614502	10	301.65923614502
11	1685	2991.16528320313	10	299.116528320313
12	1017	2986.73301696777	8	373.341627120972
13	1672	2956.48240661621	9	328.498045179579
14	1359	2915.6273651123	9	323.958596123589
15	1473	2900.80210113525	9	322.311344570584

[2(D)] Analyze **delivery time efficiency** by calculating the percentage of on-time deliveries per region.

SQL Query(MS SQL Server)

```
WITH Delivery_Stats AS (  
    SELECT  
        o.city,  
        COUNT(*) AS total_deliveries,  
        SUM(CASE WHEN dp.delivery_status = 'On Time' THEN 1 ELSE 0 END) AS on_time_deliveries  
    FROM delivery_performance dp  
    JOIN Orders o ON dp.customer_id = o.customer_id  
    GROUP BY o.city  
)  
SELECT  
    city,  
    total_deliveries,  
    on_time_deliveries,  
    (on_time_deliveries * 100.0 / total_deliveries) AS on_time_percentage  
FROM Delivery_Stats;
```

Output

	city	total_deliveries	on_time_deliveries	on_time_percentage
1	Los Angeles	5091	4235	83.186014535454
2	New York	4874	4083	83.771029954862
3	Houston	4937	4143	83.917358719870
4	Chicago	4933	4187	84.877356578147
5	San Francisco	5198	4408	84.801846864178

3. Python Assignment: Customer Segmentation & Demand Patterns **([Project link](#))**

Objective

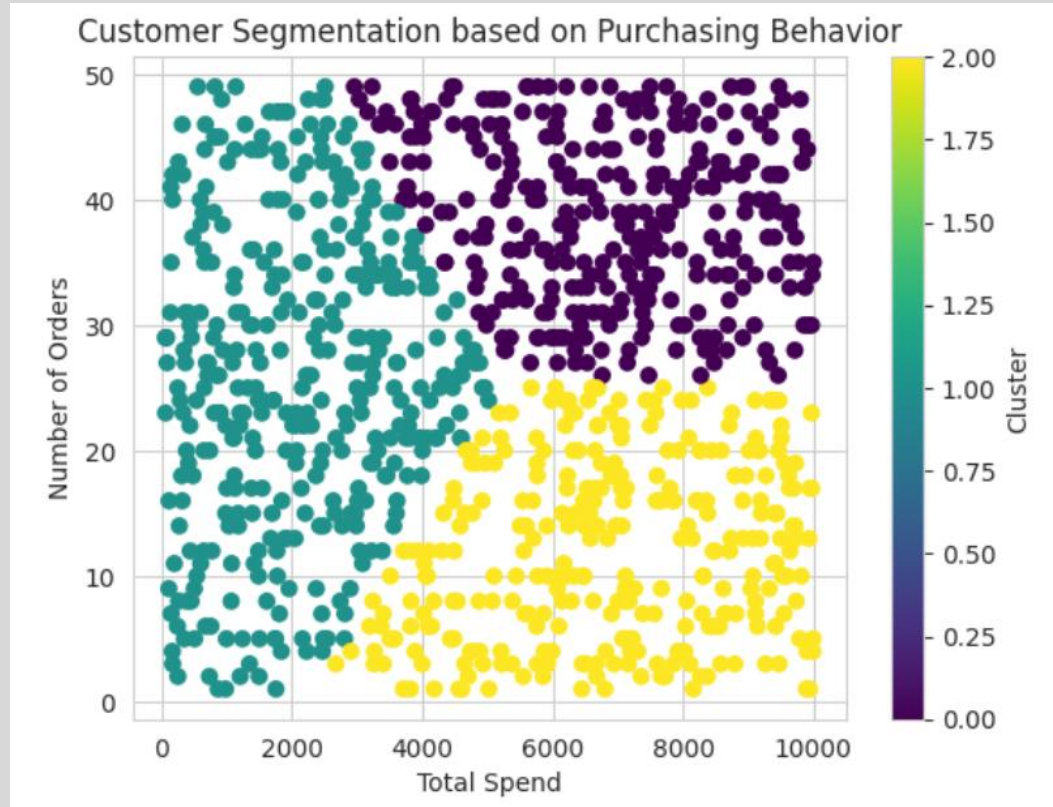
Segment customers based on purchasing behavior and detect demand trends.

Tasks

- A. Segment customers** into high-value, frequent, and occasional buyers using **K-Means clustering**.
- b. Analyze sales trends** to identify peak ordering periods.
- c. Visualize customer segments** and order patterns using graphs.

Dataset: customers.csv, sales_data.csv

[3(A)] Segment customers into high-value, frequent, and occasional buyers using **K-Mean Clustering**.



About :

Purple: Most valuable customers.

Yellow: Big spenders but not frequent shoppers.

Teal: Budget-conscious or occasional buyers.

Observation :

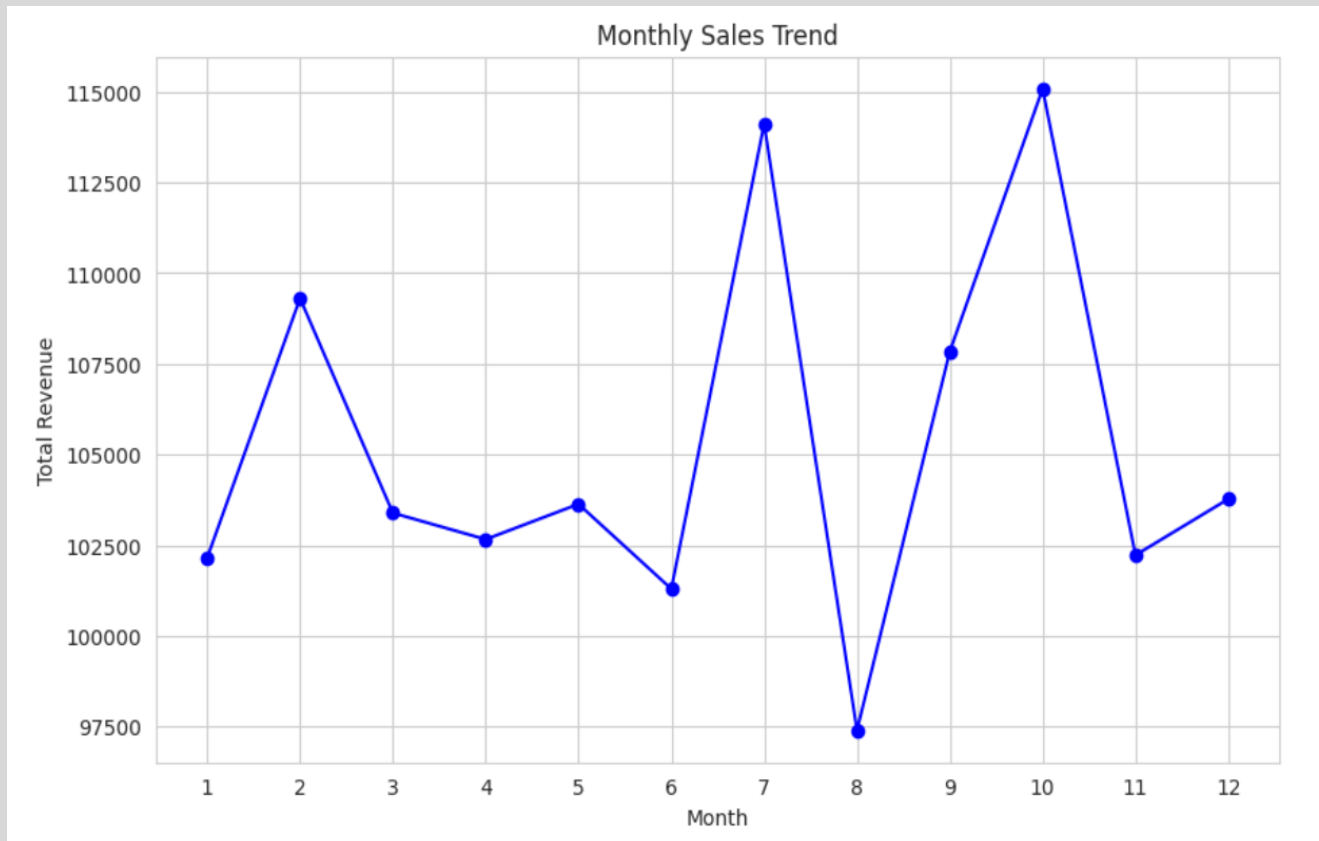
Yellow: Most premium customers.

(on less order Spend more Money)

Purple: Most Loyal and valuable customers.

[3(B)] Analyze sales trends to identify peak ordering periods.

Monthly wise Sales Trend



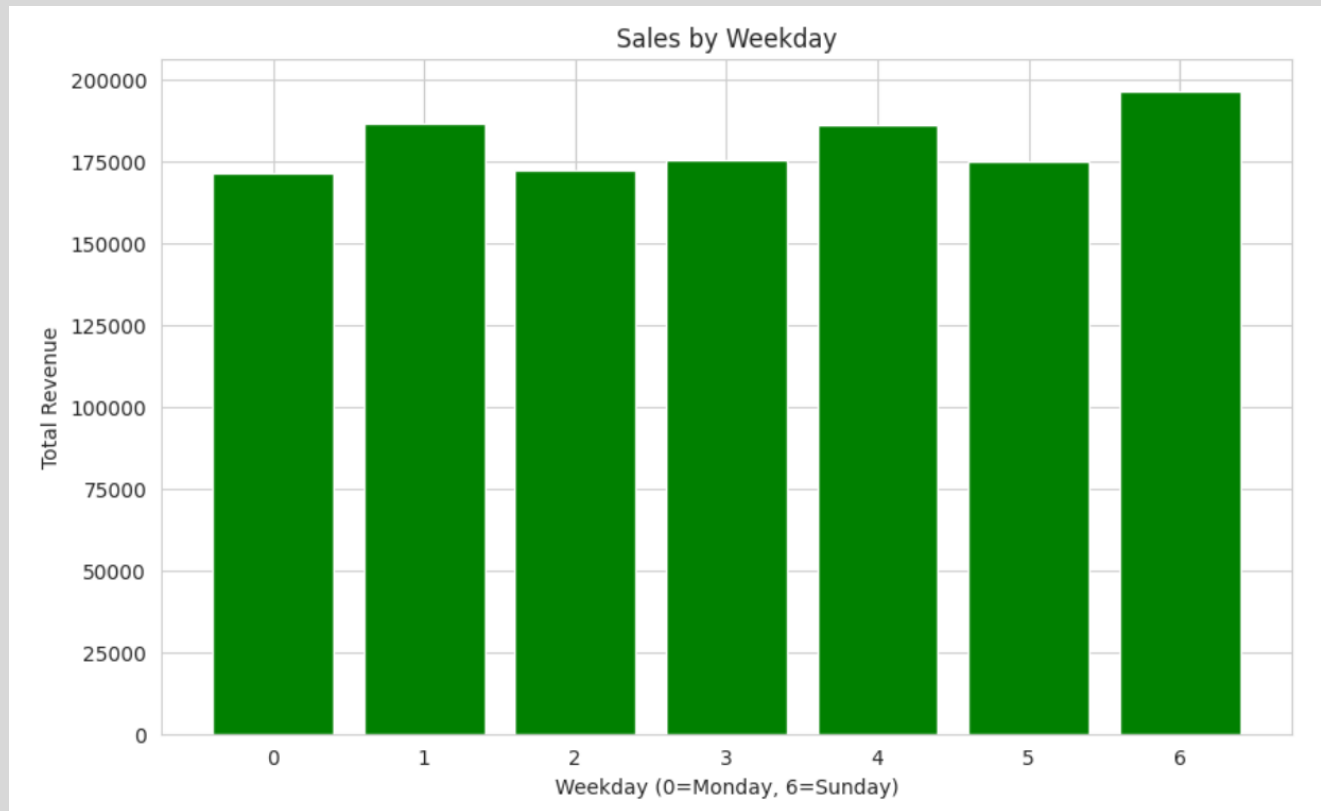
Observation:

Peak Sales: Highest revenue recorded in **July and October**.

Lowest Sales: **August** has the sharpest drop in revenue.

Moderate Sales: Sales relatively stable in **March, April, and May**

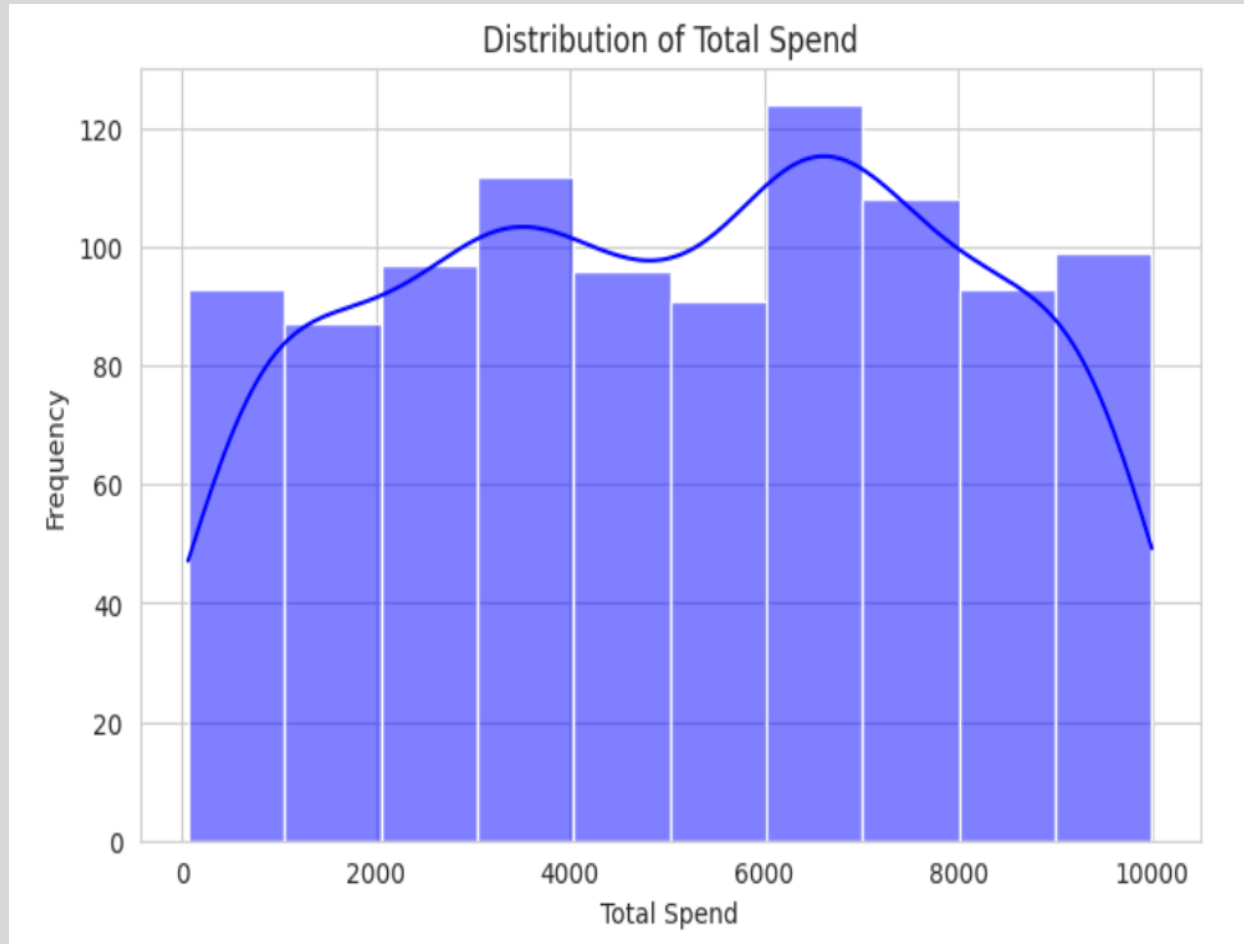
Days wise Sales Trend



observation:

- **Weekend sales (Friday-Sunday) seem to be the highest**, indicating higher customer activity.
- **Mon, wed and Thursday may need promotional strategies** to boost revenue.

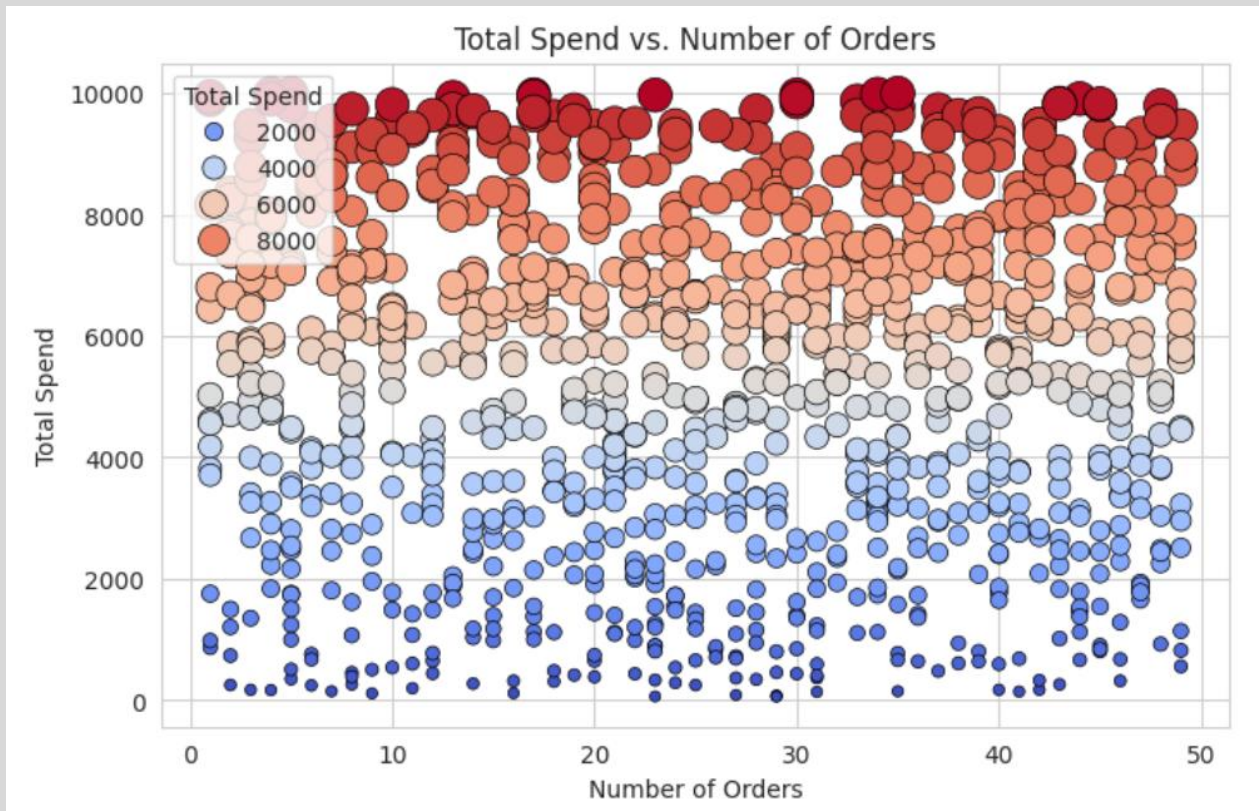
[3(C)] Visualize customer segments and order patterns using graphs.



Observation:

Most spend amounts range between 2,000 and 8,000, meaning the majority of transactions fall in this range.

Total Spend vs. Number of orders



Observation:

>> A **cluster** of big spenders (**red bubbles**) may be seen at 30+ orders, indicating that **regular customers** typically spend much more.

>> With less than 10 orders, there are a lot of lower-spending **customers** who can be new or infrequent purchasers.

4. Business Case Study: Discount Impact Analysis - ([Project link](#))

Objective

- Analyze the impact of discount strategies on profitability and retention.

Tasks

- Compare **customer spending behavior before and after** discounts.
- Identify **customer segments most responsive** to discounts.
- Recommend a strategy to **maximize revenue while maintaining profitability**.

Dataset: `discount_campaign.csv`

[4(A)] Compare **customer spending behavior before and after** discounts.

Summary analysis

discount_analysis

	discount_applied	avg_order_count_before	avg_order_count_after	avg_order_count_change	avg_spend_before	avg_spend_after	avg_spend_change
0	No	9.826772	15.724409	214.285376	2538.720034	5092.106617	307.252098
1	Yes	10.130081	15.939024	180.674804	2607.674516	5181.592918	530.997234

Average Order Count Before & After Discount



About

>> Before applying discounts, the average order count was around **10**.
>> After applying discounts, the average order count **jumped to approximately 16**.

OBSERVATION

>> Even customers who didn't receive discounts showed increased order counts, but the growth is noticeably smaller compared to those who received discounts.

Average spend Before & After Discount



About

>> Before applying discounts, average spend was around **\$2,500**.

>> After applying discounts, average spend **jumped to over \$5,000**.

OBSERVATION

>> Even customers who didn't receive a **discount showed increased spending**, but the effect is more pronounced for those who received discounts.

Conclusion:

1. Impact on Order Count

>> **Customers who received discounts had a slight increase in order count** after the discount.

>> However, the number of orders placed by customers without discounts also **increased significantly**, indicating that variables other than **discounts may also affect sales**.

2. Impact on Total Spending

>> **Both groups** saw an **increase in total spending**, but the percentage **rise in spending for customers without discounts** was **higher** than that of **customers** who did.

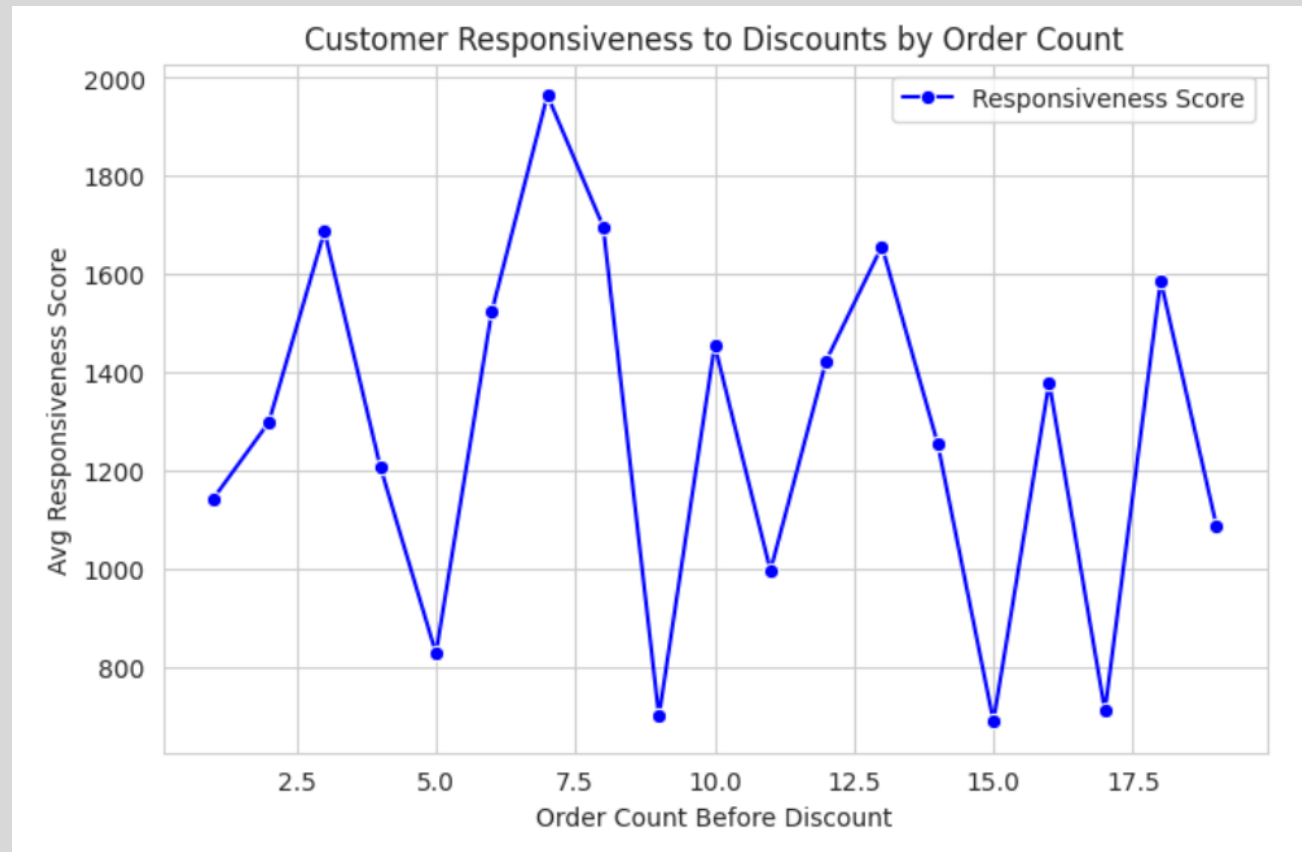
3. Effectiveness of the Discount Strategy

>> The **discount strategy helped increase engagement (more orders) but did not significantly boost revenue**.

>> Discounts may not have been the primary factor for the rise in purchases, as non-discounted clients displayed higher overall expenditure and a larger order count increase.

>> Applying discounts widely may not be as beneficial as a more focused discount strategy. For example, concentrating on the intended audience.

[4(B)] Identify **customer segments most responsive** to discounts.

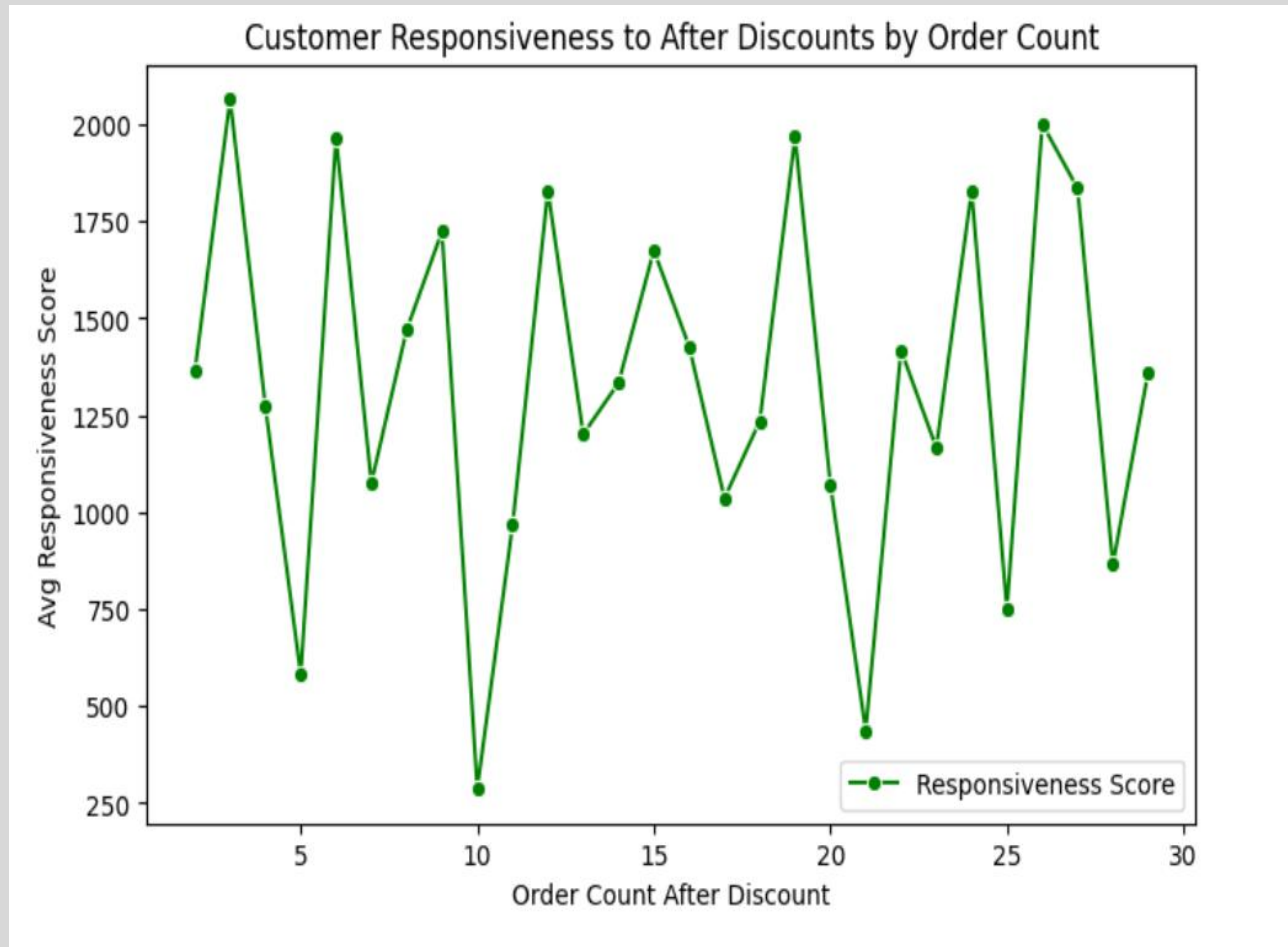


About

>> The responsiveness score fluctuates significantly with the order count before discounts.

>> The highest responsiveness score is around 2000 at approximately 8 orders.

>> There are several drops in responsiveness, indicating inconsistency in customer behavior before discounts.



About

>> The responsiveness score also fluctuates but covers a wider range of order counts (up to 30).

>> The responsiveness score reaches around 2000 multiple times, suggesting stronger engagement post-discount.

>> There are more frequent peaks, indicating an overall increase in responsiveness after the discount.

Factor	Before Discount (Blue Graph)	After Discount (Green Graph)
Order Count Range	Up to Approx 18 orders	Up to Approx 30 orders
Responsiveness Peaks	Peaks around 2000 but inconsistent	Peaks around 2000 more frequently
Variability	Highly fluctuating, sharp drops	Still fluctuating, but more stable peaks
Overall Trend	Unstable engagement	More consistent responsiveness
Impact of Discounts	Customers engage irregularly before discounts	Discounts lead to higher, more frequent engagement

Conclusion:

- **Discounts increase customer engagement**, leading to **higher and more frequent peaks** in responsiveness.
- **Order counts increase after discounts**, meaning customers place more orders when discounts are available.
- The **fluctuations before discounts suggest inconsistency**, whereas after discounts, customers respond more predictably.
- This indicates that **offering discounts strategically can drive higher engagement and order volume**.

[4(C)] Recommend a strategy to maximize revenue while maintaining profitability.

Recommendation:

- ☐ **Refine discount strategies** to focus on the right customer segments rather than applying discounts across the board.

1. Segmented Discounting Approach

A. for High-Responsiveness Segments:

- >> Use **personalized offers based on past purchase behavior** to maximize effectiveness.
- >> Consider **loyalty-based incentives** instead of direct discounts to sustain long-term engagement.

B. Low-Responsiveness Segments:

- >> Avoid heavy discounting, as these customers **do not show a strong response**.

2. Optimize Discount Structures

Instead of generic discounts, implement strategies that **incentivize higher spending**:

- ❑ **Find external factors** influencing spending behavior, as discounts alone did not drive higher revenues.

Thank You