

# Customer Behaviour Analysis

## 1. Project Overview

This project explores customer shopping behavior using 3,900+ purchase transactions across diverse product categories. The analysis focuses on customer segmentation, spending trends, product preferences, and subscription behavior, visualized through interactive Power BI dashboards to enable strategic business decision-making..

## 2. Dataset Summary

- Rows: 3,900 - Columns: 18 - Key Features: - Customer demographics (Age, Gender, Location, Subscription Status) - Purchase details (Item Purchased, Category, Purchase Amount, Season, Size, Colour) - Shopping behaviour (Discount Applied, Promo Code Used, Previous Purchases, Frequency of Purchases, Review Rating, Shipping Type) - Missing Data: 37 values in Review Rating column

## 3. Exploratory Data Analysis using Python

We began with data preparation and cleaning in Python:

- **Data Loading:** Imported the dataset using `pandas`.
- **Initial Exploration:** Used `df.info()` to check structure and `.describe()` for summary statistics.

	Customer ID	Age	Gender	Item Purchased	Category	Purchase Amount (USD)	Location	Size	Color	Season	Review Rating	Subscription Status	Shipping Type	Discount Applied
count	3900.000000	3900.000000	3900	3900	3900	3900.000000	3900	3900	3900	3900	3863.000000	3900	3900	39
unique	NaN	NaN	2	25	4	NaN	50	4	25	4	NaN	2	6	
top	NaN	NaN	Male	Blouse	Clothing	NaN	Montana	M	Olive	Spring	NaN	No	Free Shipping	
freq	NaN	NaN	2652	171	1737	NaN	96	1755	177	999	NaN	2847	675	22
mean	1950.500000	44.068462	NaN	NaN	NaN	59.764359	NaN	NaN	NaN	NaN	3.750065	NaN	NaN	NaN
std	1125.977353	15.207589	NaN	NaN	NaN	23.685392	NaN	NaN	NaN	NaN	0.716983	NaN	NaN	NaN
min	1.000000	18.000000	NaN	NaN	NaN	20.000000	NaN	NaN	NaN	NaN	2.500000	NaN	NaN	NaN
25%	975.750000	31.000000	NaN	NaN	NaN	39.000000	NaN	NaN	NaN	NaN	3.100000	NaN	NaN	NaN
50%	1950.500000	44.000000	NaN	NaN	NaN	60.000000	NaN	NaN	NaN	NaN	3.800000	NaN	NaN	NaN
75%	2925.250000	57.000000	NaN	NaN	NaN	81.000000	NaN	NaN	NaN	NaN	4.400000	NaN	NaN	NaN
max	3900.000000	70.000000	NaN	NaN	NaN	100.000000	NaN	NaN	NaN	NaN	5.000000	NaN	NaN	NaN

Discount Applied	Promo Code Used	Previous Purchases	Payment Method	Frequency of Purchases
3900	3900	3900.000000	3900	3900
2	2	NaN	6	7
No	No	NaN	PayPal	Every 3 Months
2223	2223	NaN	677	584
NaN	NaN	25.351538	NaN	NaN
NaN	NaN	14.447125	NaN	NaN
NaN	NaN	1.000000	NaN	NaN
NaN	NaN	13.000000	NaN	NaN
NaN	NaN	25.000000	NaN	NaN
NaN	NaN	38.000000	NaN	NaN
NaN	NaN	50.000000	NaN	NaN

- **Missing Data Handling:** Checked for null values and imputed missing values in the `Review Rating` column using the median rating of each product category.
- **Column Standardization:** Renamed columns to **snake case** for better readability and documentation.
- Feature Engineering:
  - Created `age_group` column by binning customer ages.
  - Created `purchase_frequency_days` column from purchase data.
- **Data Consistency Check:** Verified if `discount_applied` and `promo_code_used` were redundant; dropped `promo_code_used`.
- **Database Integration:** Connected Python script to MS SQL and loaded the cleaned DataFrame into the database for SQL analysis.

## 4. Data Analysis using SQL

```
--1. Total Revenue & Average Order Value
SELECT
    SUM(purchase_amount) AS total_revenue,
    AVG(purchase_amount) AS avg_order_value
FROM customerBehaviour;
```

	total_revenue	avg_order_value
1	233081	59

```
--2. Revenue by Category (Top-selling products)
SELECT
```

```
category,
    SUM(purchase_amount) AS total_revenue
```

```
FROM customerBehaviour  
GROUP BY category  
ORDER BY total_revenue DESC;
```

	category	total_revenue
1	Clothing	104264
2	Accessories	74200
3	Footwear	36093
4	Outerwear	18524

#### --3. Customer Segmentation by Age Group

```
SELECT  
    age_group,  
    COUNT(DISTINCT customer_id) AS total_customers,  
    SUM(purchase_amount) AS revenue  
FROM customerBehaviour  
GROUP BY age_group  
ORDER BY revenue DESC;
```

	age_group	total_customers	revenue
1	Young Adult	1028	62143
2	Middle-aged	986	59197
3	Adult	942	55978
4	Senior	944	55763

#### --4. Gender-wise Purchase Behavior

```
SELECT  
    gender,  
    COUNT(customer_id) AS total_orders,  
    SUM(purchase_amount) AS revenue,  
    AVG(review_rating) AS avg_rating  
FROM customerBehaviour  
GROUP BY gender;
```

	gender	total_orders	revenue	avg_rating
1	Male	2652	157890	3.75411010558069
2	Female	1248	75191	3.74142628205128

#### --5. Top 10 High-Value Customers

```
SELECT top 10  
    customer_id,  
    SUM(purchase_amount) AS lifetime_value,  
    COUNT(*) AS total_orders  
FROM customerBehaviour  
GROUP BY customer_id  
ORDER BY lifetime_value DESC;
```

	customer_id	lifetime_value	total_orders
1	43	100	1
2	96	100	1
3	194	100	1
4	205	100	1
5	244	100	1
6	249	100	1
7	456	100	1
8	519	100	1
9	582	100	1
10	616	100	1

#### --6. Subscription vs Non-Subscription Analysis

```
SELECT
    subscription_status,
    COUNT(DISTINCT customer_id) AS customers,
    SUM(purchase_amount) AS revenue,
    AVG(previous_purchases) AS avg_past_orders
FROM customerBehaviour
GROUP BY subscription_status;
```

	subscription_status	customers	revenue	avg_past_orders
1	No	2847	170436	25
2	Yes	1053	62645	26

#### --7. Discount Effectiveness

```
SELECT
    discount_applied,
    COUNT(*) AS total_orders,
    SUM(purchase_amount) AS revenue,
    AVG(purchase_amount) AS avg_order_value
FROM customerBehaviour
GROUP BY discount_applied;
```

	discount_applied	total_orders	revenue	avg_order_value
1	Yes	1677	99411	59
2	No	2223	133670	60

#### --8. Purchase Frequency Analysis

```
SELECT
    frequency_of_purchases,
    COUNT(DISTINCT customer_id) AS customers,
    SUM(purchase_amount) AS revenue
FROM customerBehaviour
GROUP BY frequency_of_purchases
ORDER BY revenue DESC;
```

	frequency_of_purchases	customers	revenue
1	Every 3 Months	584	35088
2	Annually	572	34419
3	Quarterly	563	33771
4	Bi-Weekly	547	33200
5	Monthly	553	32810
6	Fortnightly	542	32007
7	Weekly	539	31786

#### --9. Seasonal Sales Trend

```
SELECT
    season,
    SUM(purchase_amount) AS revenue,
    COUNT(*) AS total_orders
FROM customerBehaviour
GROUP BY season
ORDER BY revenue DESC;
```

	season	revenue	total_orders
1	Fall	60018	975
2	Spring	58679	999
3	Winter	58607	971
4	Summer	55777	955

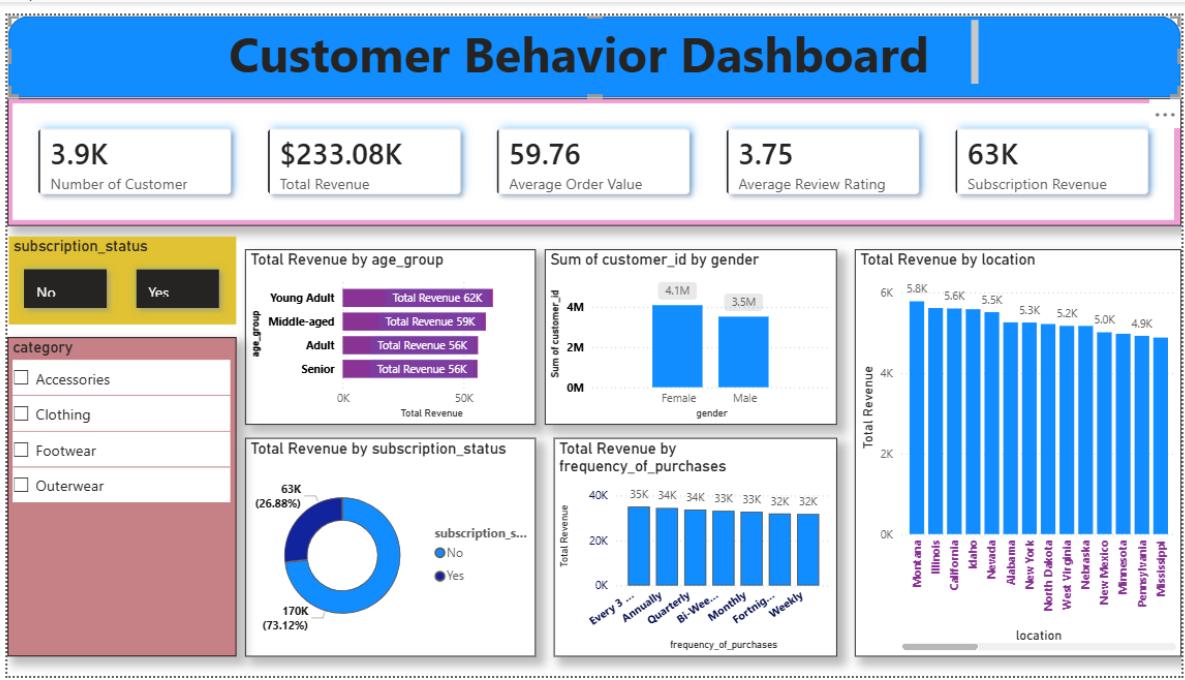
#### --10. Shipping Type Performance

```
SELECT
    shipping_type,
    COUNT(*) AS orders,
    AVG(review_rating) AS avg_rating
FROM customerBehaviour
GROUP BY shipping_type
ORDER BY avg_rating DESC;
```

	shipping_type	orders	avg_rating
1	Standard	654	3.81896024464832
2	Express	646	3.77306501547988
3	2-Day Shipping	627	3.76570972886762
4	Next Day Air	648	3.71975308641975
5	Free Shipping	675	3.71703703703704
6	Store Pickup	650	3.70723076923077

## 5. Dashboard in Power BI

Finally, an interactive Power BI dashboard was developed to visually communicate key insights, enabling stakeholders to explore trends, compare segments, and make data-driven decisions.



## 6. Business Recommendations

- 1 Focus marketing campaigns on **Young Adult and Middle-aged customers** to maximize revenue contribution.
- 2 Increase **subscription adoption** by offering exclusive benefits like discounts and faster delivery.
- 3 Prioritize inventory and promotions for **high-revenue product categories** to improve profitability.
- 4 Launch **location-specific sales strategies** in top-performing regions to boost regional revenue.
- 5 Encourage repeat purchases through **loyalty programs and personalized offers** to increase customer lifetime value.
- 6 Improve **average order value** using product bundling and cross-selling strategies.
- 7 Enhance **customer experience and shipping quality** to raise review ratings above 4.0.