Customer Shopping Behaviour Analysis

1. Project Overview

This project analyses customer shopping behaviour using transactional data from 3,900 purchases across various product categories. The goal is to uncover insights into spending patterns, customer segments, product preferences, and subscription behaviour to guide strategic business decisions.

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, Color) - 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.

[104]:

	Customer ID	Age	Gender	Item Purchased	Category	Purchase Amount (USD)	Location	Size	Color	Season	Review Rating	Subscription Status	Shipping Type	Discount Applied	Promo Code Used
count	3900.000000	3900.000000	3900	3900	3900	3900.000000	3900	3900	3900	3900	3863.000000	3900	3900	3900	3900
unique	NaN	NaN	2	25	4	NaN	50	4	25	4	NaN	2	6	2	2
top	NaN	NaN	Male	Blouse	Clothing	NaN	Montana	M	Olive	Spring	NaN	No	Free Shipping	No	No
freq	NaN	NaN	2652	171	1737	NaN	96	1755	177	999	NaN	2847	675	2223	2223
mean	1950.500000	44.068462	NaN	NaN	NaN	59.764359	NaN	NaN	NaN	NaN	3.750065	NaN	NaN	NaN	NaN
std	1125.977353	15.207589	NaN	NaN	NaN	23.685392	NaN	NaN	NaN	NaN	0.716983	NaN	NaN	NaN	NaN
min	1.000000	18.000000	NaN	NaN	NaN	20.000000	NaN	NaN	NaN	NaN	2.500000	NaN	NaN	NaN	NaN
25%	975.750000	31.000000	NaN	NaN	NaN	39.000000	NaN	NaN	NaN	NaN	3.100000	NaN	NaN	NaN	NaN
50%	1950.500000	44.000000	NaN	NaN	NaN	60.000000	NaN	NaN	NaN	NaN	3.800000	NaN	NaN	NaN	NaN
75%	2925.250000	57.000000	NaN	NaN	NaN	81.000000	NaN	NaN	NaN	NaN	4.400000	NaN	NaN	NaN	NaN
max	3900.000000	70.000000	NaN	NaN	NaN	100.000000	NaN	NaN	NaN	NaN	5.000000	NaN	NaN	NaN	NaN

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3900 entries, 0 to 3899
Data columns (total 18 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   Customer ID             3900 non-null   int64
 1   Age                     3900 non-null   int64
 2   Gender                  3900 non-null   object
 3   Item Purchased          3900 non-null   object
 4   Category                3900 non-null   object
 5   Purchase Amount (USD)   3900 non-null   int64
 6   Location                3900 non-null   object
 7   Size                    3900 non-null   object
 8   Color                   3900 non-null   object
 9   Season                  3900 non-null   object
 10  Review Rating           3863 non-null   float64
 11  Subscription Status     3900 non-null   object
 12  Shipping Type           3900 non-null   object
 13  Discount Applied        3900 non-null   object
 14  Promo Code Used         3900 non-null   object
 15  Previous Purchases      3900 non-null   int64
 16  Payment Method          3900 non-null   object
 17  Frequency of Purchases  3900 non-null   object
dtypes: float64(1), int64(4), object(13)
memory usage: 548.6+ KB
```

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 PostgreSQL and loaded the cleaned DataFrame into the database for SQL analysis.

4. Data Analysis using SQL (Business Transactions)

- **. Revenue by Gender** - Compared total revenue generated by male vs. female
- customers.

```sql
--q1. what is total revenue generated by male vs female ?
select gender ,
    sum(purchase_amount) as revenue
from customer
group by gender;
```

00 % No issues found

Results Messages

	gender	revenue
1	Male	157890
2	Female	75191

- **High-Spending Discount Users –** Identified customers who used discounts but still spent above the average purchase amount.

```sql
--q2. which customer used a discount but still spent more than average purchase amount
select customer_id,
    purchase_amount
from customer
where discount_applied='yes' and purchase_amount>=
    (select avg(purchase_amount) from customer);
```

100 % No issues found

Results Messages

	customer_id	purchase_amount
1	2	64
2	3	73
3	4	90
4	7	85
5	9	97
6	12	68
7	13	72
8	16	81
9	20	90
10	22	62

- **Top 5 Products by Rating** – Found products with the highest average review ratings Compared average purchase amounts between Standard and Express shipping.

```sql
--q3 which are the top 5 products with highest average review rating?
select top 5  item_purchased,
    round(avg(review_Rating),2) as average_product_rating
from customer
group by item_purchased
order by average_product_rating desc;
```

100 % No issues found

Results Messages

	item_purchased	average_product_rating
1	Gloves	3.86
2	Sandals	3.84
3	Boots	3.82
4	Hat	3.8
5	Handbag	3.78

- **Shipping Type Comparison -** Compared average purchase amounts between

Standard and Express shipping.

```sql
--Q4 compare average  purchae amounts between standard and express shipping
select shipping_type,
    AVG(purchase_amount) as purchase_amount
from customer
where shipping_type in ('Standard','Express')
group by shipping_type;
```

100 % No issues found

Results Messages

	shipping_type	purchase_amount
1	Standard	58
2	Express	60

- **Subscribers vs. Non-Subscribers** – Compared average spend and total revenue across subscription status.


```sql
--Q4 compare average  purchae amounts between standard and express shipping
select shipping_type,
    AVG(purchase_amount) as purchase_amount
from customer
where shipping_type in ('Standard','Express')
group by shipping_type;
```

100 % No issues found

Results Messages

	shipping_type	purchase_amount
1	Standard	58
2	Express	60

- **Discount-Dependent Products** – Identified 5 products with the highest percentage of discounted purchases.

```sql
-- Q6 which 5 products have the highest percentage of purchase with discounts applied?
SELECT TOP 5
    item_purchased,
   FORMAT( ROUND(
        (CAST(SUM(CASE WHEN discount_applied = 'Yes' THEN 1 ELSE 0 END) AS DECIMAL(10,2))
        / COUNT(*) * 100),
    2),'#.00') AS discount_rate
FROM customer
GROUP BY item_purchased
ORDER BY discount_rate;
```

100 % No issues found

Results Messages

	item_purchased	discount_rate
1	Socks	32.70
2	Blouse	33.92
3	Sandals	36.88
4	Skirt	38.61
5	Handbag	39.87

- **Customer Segmentation** – Classified customers into New, Returning, and Loyal segments based on purchase history.

```sql
-- q7. segement customer into new ,returning and loyal based
   -- on their total numbers of previouse purchases and show the count of each segeme
with customer_type as
(
select customer_id,previous_purchases,
    case when previous_purchases=1 then 'New'
        when previous_purchases between 2 and  10 then 'returning'
        else   'Loyal' end as customer_segement
 from customer)
 select customer_segement,
    count(*) as number_of_customer
 from customer_type
 group by customer_segement
 order by number_of_customer desc
```

100 % No issues found

Results Messages

	customer_segement	number_of_customer
1	Loyal	3116
2	returning	701
3	New	83

- **Top 3 Products per Category** – Listed the most purchased products within each category

```sql
-- Q8 what are the top 3 most purchased products withing each category
with item_cout as(
select category,
   item_purchased,
   count(customer_id) as total_orders,
   row_number() over(partition by category order by count(customer_id) desc) as item_rank
from customer
group by category,item_purchased )
select  item_rank,
   category,item_purchased,
   total_orders
from item_cout
where item_rank<=3;
```

100 % No issues found

Results Messages

	item_rank	category	item_purchased	total_orders
1	1	Accessories	Jewelry	171
2	2	Accessories	Belt	161
3	3	Accessories	Sunglasses	161
4	1	Clothing	Blouse	171
5	2	Clothing	Pants	171
6	3	Clothing	Shirt	169
7	1	Footwear	Sandals	160
8	2	Footwear	Shoes	150
9	3	Footwear	Sneakers	145
10	1	Outerwear	Jacket	163
11	2	Outerwear	Coat	161

- **Repeat Buyers & Subscriptions** – Checked whether customers with >5 purchases are more likely to subscribe.

```
80    -- Q9  are the customers are repeat buyers(more than 5 previouse purchase ) also likely to subscrib
81    select subscription_status,
82       count(customer_id) as repeate_buyers
83    from customer
84    where previous_purchases>5
85    group by subscription_status
```

100 % — No issues found — Ln: 81, Ch: 1 (146

Results | Messages

	subscription_status	repeate_buyers
1	Yes	958
2	No	2518

- **Revenue by Age Group –** Calculated total revenue contribution of each age group

```
87    --q10  what is the revenue contribution of each age group?
88    select age_group,
89       sum(purchase_amount) as total_revenue
90    from customer
91    group by age_group
92    order  by total_revenue
93
```

100 % — No issues found

Results | Messages

	age_group	total_revenue
1	Senior	55763
2	Adult	55978
3	Middle Aged	59197
4	Young Adult	62143

- **Top 5 locations with highest sales-** Focus campaigns in top-performing cities & Ensure popular categories are stocked in these regions
-

```
95    --q11. What are the top 3 locations with the highest sales performance?
96     select top 5  location,
97       sum(purchase_amount) as total_revenue
98    from customer
99    group by location
100   order  by total_revenue desc
101
```

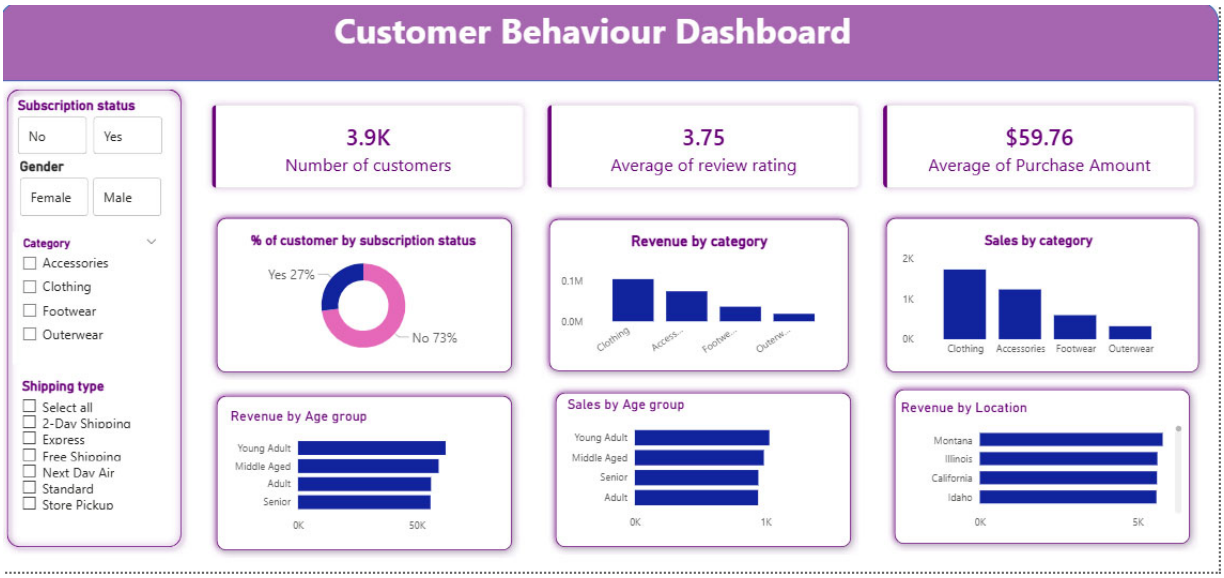
100 % — No issues found

Results | Messages

	location	total_revenue
1	Montana	5784
2	Illinois	5617
3	California	5605
4	Idaho	5587
5	Nevada	5514

5. Dashboard in Power BI

Finally, built an interactive dashboard in Power BI to present insights visually.



6. Business Recommendations

Boost Subscriptions

- Encourage more customers to subscribe by promoting exclusive benefits such as discounts, faster shipping, or loyalty rewards.
- Subscriptions increase recurring revenue and strengthen long-term customer relationships.

- **Customer Loyalty Programs**
  - Design reward systems for repeat buyers (e.g., points, cashback, or tiered benefits).
  - Helps move customers into the “Loyal” segment, reducing churn and increasing lifetime value.
- **Review Discount Policy**
  - Assess the balance between offering discounts to drive sales and maintaining healthy profit margins.
  - Identify products overly dependent on discounts and adjust pricing strategies to protect profitability.
- **Product Positioning**
  - Highlight top-rated and best-selling products in marketing campaigns.
  - Use customer reviews and ratings to build trust and attract new buyers.
- **Targeted Marketing**
  - Focus marketing efforts on high-revenue age groups and customers who prefer express shipping.
  - Personalize campaigns based on demographics, purchase frequency, and product preferences to maximize ROI.