CUSTOMER SHOPPING BEHAVIOUR ANALYSIS

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.

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

EXPLORATORY DATA ANALYSIS USING PYTHON

I 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.

4]:

		Age	Gender	Item Purchased	Category	Purchase Amount (USD)	Location	Size	Color	Season	Review Rating	Subscription Status	Shipping Type	Discount Applied	Promo Code Used	Previous Purchases	Payment Method	Fr Pr
0	1	55	Male	Blouse	Clothing	53	Kentucky	L	Gray	Winter	3.1	Yes	Express	Yes	Yes	14	Venmo	Fo
1	2	19	Male	Sweater	Clothing	64	Maine	L	Maroon	Winter	3.1	Yes	Express	Yes	Yes	2	Cash	Fo
2	3	50	Male	Jeans	Clothing	73	Massachusetts	S	Maroon	Spring	3.1	Yes	Free Shipping	Yes	Yes	23	Credit Card	
3	4	21	Male	Sandals	Footwear	90	Rhode Island	M	Maroon	Spring	3.5	Yes	Next Day Air	Yes	Yes	49	PayPal	
4	5	45	Male	Blouse	Clothing	49	Oregon	M	Turquoise	Spring	2.7	Yes	Free Shipping	Yes	Yes	31	PayPal	

Discount Applied	Promo Code Used	Previous Purchases	Payment Method	Frequency of Purchases
Yes	Yes	14	Venmo	Fortnightly
Yes	Yes	2	Cash	Fortnightly
Yes	Yes	23	Credit Card	Weekly
Yes	Yes	49	PayPal	Weekly
Yes	Yes	31	PayPal	Annually

- **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.

DATA ANALYSIS USING SQL (BUSINESS TRANSACTIONS)

We performed structured analysis in PostgreSQL to answer key business questions:

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

```sql
select gender, SUM (purchase_amount) as revenue
from customer
group by gender
```

	gender text	revenue numeric
1	Female	75191
2	Male	157890

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

```sql
select customer_id, purchase_amount
from customer
where discount_applied = 'Yes' and purchase_amount >= (select AVG(purchase_amount) from customer)
```

	customer_id bigint	purchase_amount bigint
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

Total rows: 839 Query complete 00:00:00.383

3. **Top 5 Products by Rating** – Found products with the highest average review ratings.

```
select item_purchased, ROUND(AVG(review_rating::numeric),2) as "Average Product Rating"
from customer
group by item_purchased
order by avg(review_rating) desc
limit 5;
```

	item_purchased text	Average Product Rating numeric
1	Gloves	3.86
2	Sandals	3.84
3	Boots	3.82
4	Hat	3.80
5	Skirt	3.78

4. **Shipping Type Comparison** – Compared average purchase amounts between Standard and Express shipping.

```
select shipping_type,
ROUND(AVG(purchase_amount),2)
from customer
where shipping_type in ('Standard','Express')
group by shipping_type
```

	shipping_type text	round numeric
1	Standard	58.46
2	Express	60.48

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

```sql
select subscription_status,
COUNT(customer_id) as total_customer,
ROUND(AVG(purchase_amount),2) as avg_spend,
ROUND(SUM(purchase_amount),2) as total_revenue
from customer
group by subscription_status
order by total_revenue, avg_spend desc;
```

	subscription_status text	total_customer bigint	avg_spend numeric	total_revenue numeric
1	Yes	1053	59.49	62645.00
2	No	2847	59.87	170436.00

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

```sql
select item_purchased,
ROUND(100*SUM(CASE WHEN discount_applied = 'Yes' THEN 1 ELSE 0 END)/COUNT(*),2) as discount_rate
from customer
group by item_purchased
order by discount_rate desc
limit 5;
```

	item_purchased text	discount_rate numeric
1	Hat	50.00
2	Sneakers	49.66
3	Coat	49.07
4	Sweater	48.17
5	Pants	47.37

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

```sql
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_segment
from customer
)

select customer_segment, count(*) as "Number of Customers"
from customer_type
group by customer_segment
```

	customer_segment text	Number of Customers bigint
1	Loyal	3116
2	New	83
3	Returning	701

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

```sql
with item_counts 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_counts
where item_rank <= 3;
```

	item_rank bigint	category text	item_purchased text	total_orders bigint
1	1	Accessori...	Jewelry	171
2	2	Accessori...	Sunglasses	161
3	3	Accessori...	Belt	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

Total rows: 11 | Query complete 00:00:00.259

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

```sql
select subscription_status,
count (customer_id) as repeat_buyers
from customer
where previous_purchases > 5
group by subscription_status
```

	subscription_status text	repeat_buyers bigint
1	No	2518
2	Yes	958

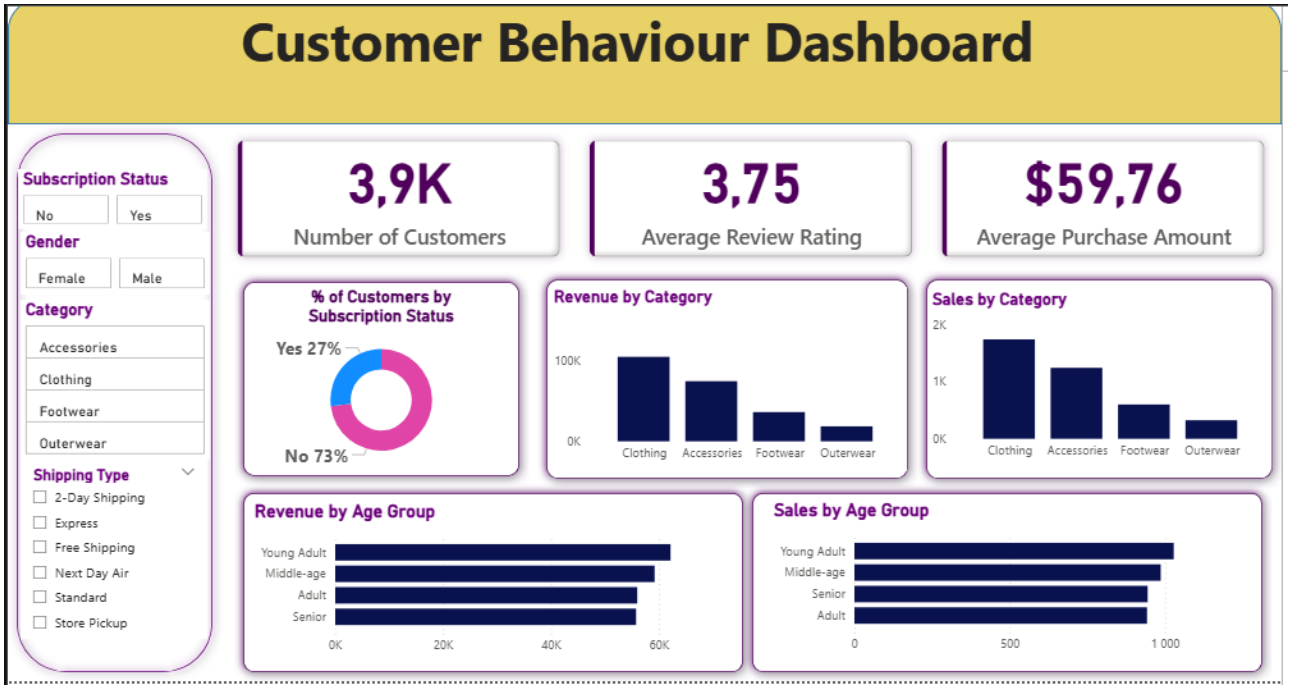
10. **Revenue by Age Group** – Calculated total revenue contribution of each age group

```sql
select age_group,
SUM(purchase_amount) as total_revenue
from customer
group by age_group
order by total_revenue desc;
```

	age_group text	total_revenue numeric
1	Young Adult	62143
2	Middle-age	59197
3	Adult	55978
4	Senior	55763

DASHBOARD IN POWER BI

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



Business Recommendations

- **Boost Subscriptions** – Promote exclusive benefits for subscribers.
- **Customer Loyalty Programs** – Reward repeat buyers to move them into the “Loyal” segment.
- **Review Discount Policy** – Balance sales boosts with margin control.
- **Product Positioning** – Highlight top-rated and best-selling products in campaigns.
- **Targeted Marketing** – Focus efforts on high-revenue age groups and express-shipping users.