Customer Shopping Behaviour Analysis

1. Project Overview

This project analyzes customer shopping behavior 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 behaviours, ultimately guiding 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 behavior (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

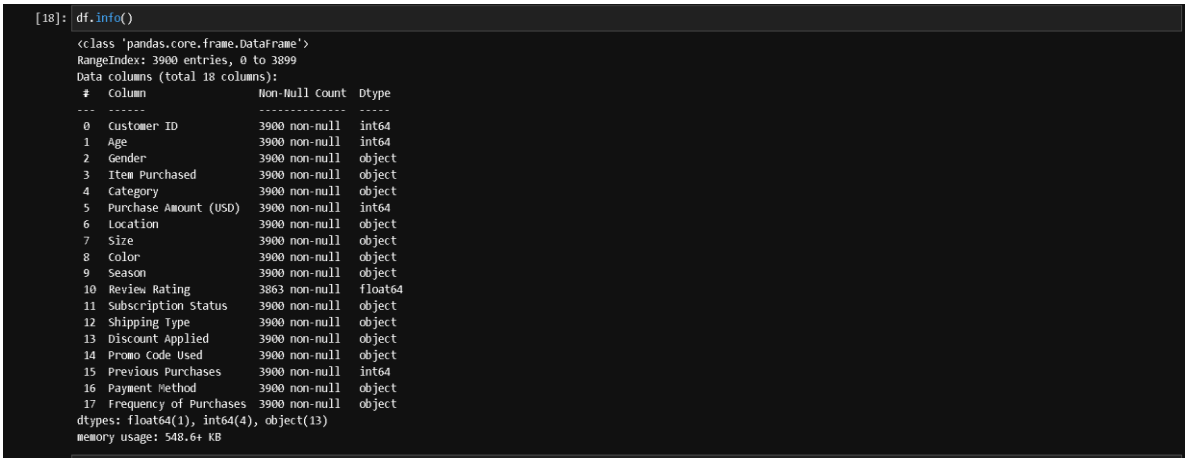
We began with data preparation and cleaning in python:

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

```
[19]: df.describe(include = 'all')
```

[19]:

	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	Previous Purchases	Payment Method	Frequency of Purchases
count	3900.000000	3900.000000	3900	3900	3900	3900.000000	3900	3900	3900	3900	3863.000000	3900	3900	3900	3900	3900.000000	3900	3900
unique	NaN	NaN	2	25	4	NaN	50	4	25	4	NaN	2	6	2	2	NaN	6	7
top	NaN	NaN	Male	Blouse	Clothing	NaN	Montana	M	Olive	Spring	NaN	No	Free Shipping	No	No	NaN	PayPal	Every 3 Months
freq	NaN	NaN	2652	171	1737	NaN	96	1755	177	999	NaN	2847	675	2223	2223	NaN	677	584
mean	1950.500000	44.068462	NaN	NaN	NaN	59.764359	NaN	NaN	NaN	NaN	3.750065	NaN	NaN	NaN	NaN	25.351538	NaN	NaN
std	1125.977353	15.207589	NaN	NaN	NaN	23.685392	NaN	NaN	NaN	NaN	0.716983	NaN	NaN	NaN	NaN	14.447125	NaN	NaN
min	1.000000	18.000000	NaN	NaN	NaN	20.000000	NaN	NaN	NaN	NaN	2.500000	NaN	NaN	NaN	NaN	1.000000	NaN	NaN
25%	975.750000	31.000000	NaN	NaN	NaN	39.000000	NaN	NaN	NaN	NaN	3.100000	NaN	NaN	NaN	NaN	13.000000	NaN	NaN
50%	1950.500000	44.000000	NaN	NaN	NaN	60.000000	NaN	NaN	NaN	NaN	3.800000	NaN	NaN	NaN	NaN	25.000000	NaN	NaN
75%	2925.250000	57.000000	NaN	NaN	NaN	81.000000	NaN	NaN	NaN	NaN	4.400000	NaN	NaN	NaN	NaN	38.000000	NaN	NaN
max	3900.000000	70.000000	NaN	NaN	NaN	100.000000	NaN	NaN	NaN	NaN	5.000000	NaN	NaN	NaN	NaN	50.000000	NaN	NaN



```
[18]: df.info()

<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)

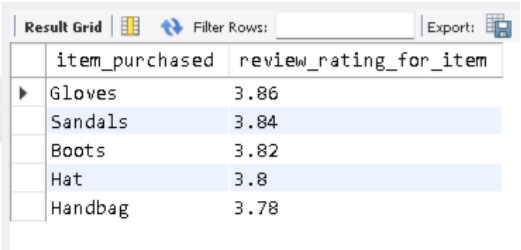
We performed structured analysis in PostgreSQL to answer key business questions:

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

	gender	revenue
1	gender	revenue
2	Male	157890
3	Female	75191
4		

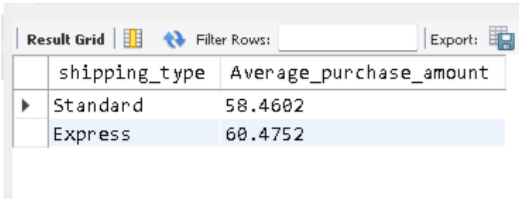
2. **High-Spending Discount Users** – Identified customers who used discounts but still spent above the average purchase amount.

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



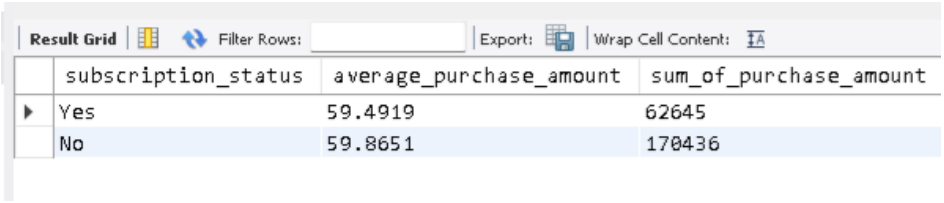
item_purchased	review_rating_for_item
Gloves	3.86
Sandals	3.84
Boots	3.82
Hat	3.8
Handbag	3.78

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



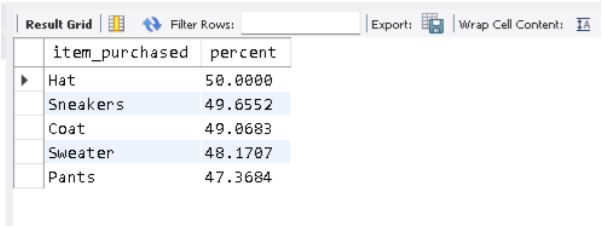
shipping_type	Average_purchase_amount
Standard	58.4602
Express	60.4752

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



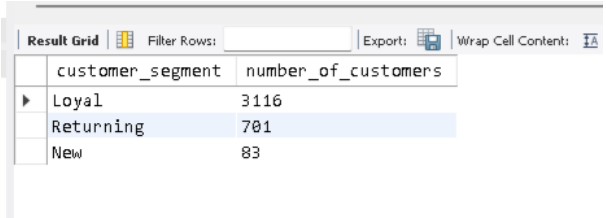
subscription_status	average_purchase_amount	sum_of_purchase_amount
Yes	59.4919	62645
No	59.8651	170436

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



item_purchased	percent
Hat	50.0000
Sneakers	49.6552
Coat	49.0683
Sweater	48.1707
Pants	47.3684

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



customer_segment	number_of_customers
Loyal	3116
Returning	701
New	83

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

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

category	item_purchased	each_item_purchase_amount	position
Accessories	Jewelry	10010	1
Accessories	Sunglasses	9649	2
Accessories	Belt	9635	3
Clothing	Blouse	10410	1
Clothing	Shirt	10332	2
Clothing	Dress	10320	3
Footwear	Shoes	9240	1
Footwear	Sandals	9200	2
Footwear	Boots	9018	3
Outerwear	Coat	9275	1
Outerwear	Jacket	9249	2

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

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

subscription_status	repeated_buyers
Yes	958
No	2518

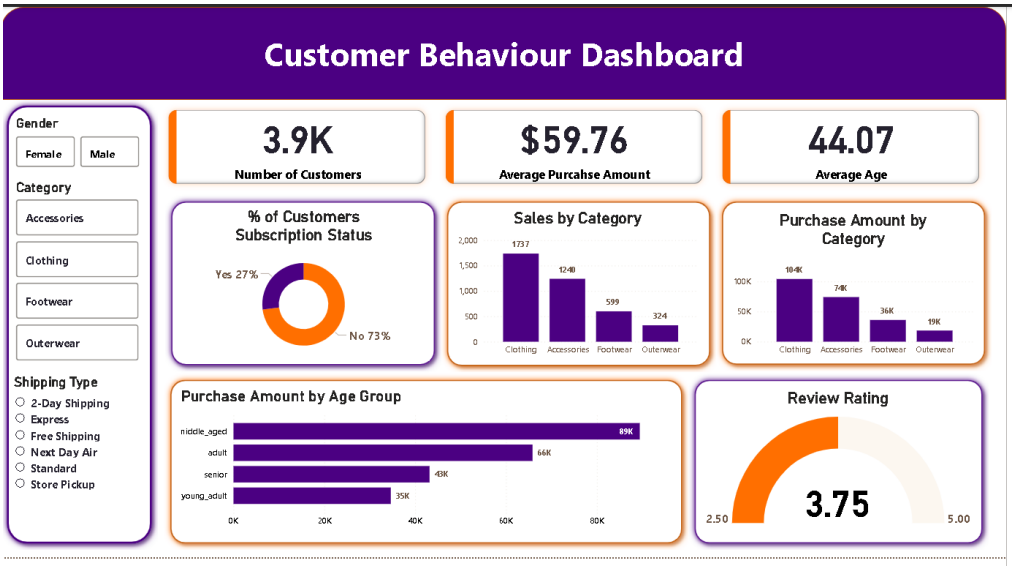
10. **Revenue by Age Group** – Calculated total revenue contribution of each age group.

Result Grid | Filter Rows: | Export:

age_group	total_revenue_by_age_group
middle_aged	89445
adult	65842
senior	43164
young_adult	34630

5. Dashboard in Power BI

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



6. 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.