

Customer Shopping Behavior Analysis Report

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 behavior 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 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

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 `df.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	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

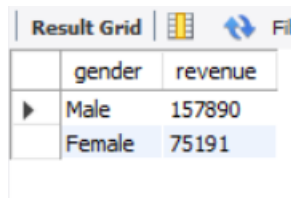
Promo Code Used	Previous Purchases	Payment Method	Frequency of Purchases
3900	3900.000000	3900	3900
2	NaN	6	7
No	NaN	PayPal	Every 3 Months
2223	NaN	677	584
NaN	25.351538	NaN	NaN
NaN	14.447125	NaN	NaN
NaN	1.000000	NaN	NaN
NaN	13.000000	NaN	NaN
NaN	25.000000	NaN	NaN
NaN	38.000000	NaN	NaN
NaN	50.000000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

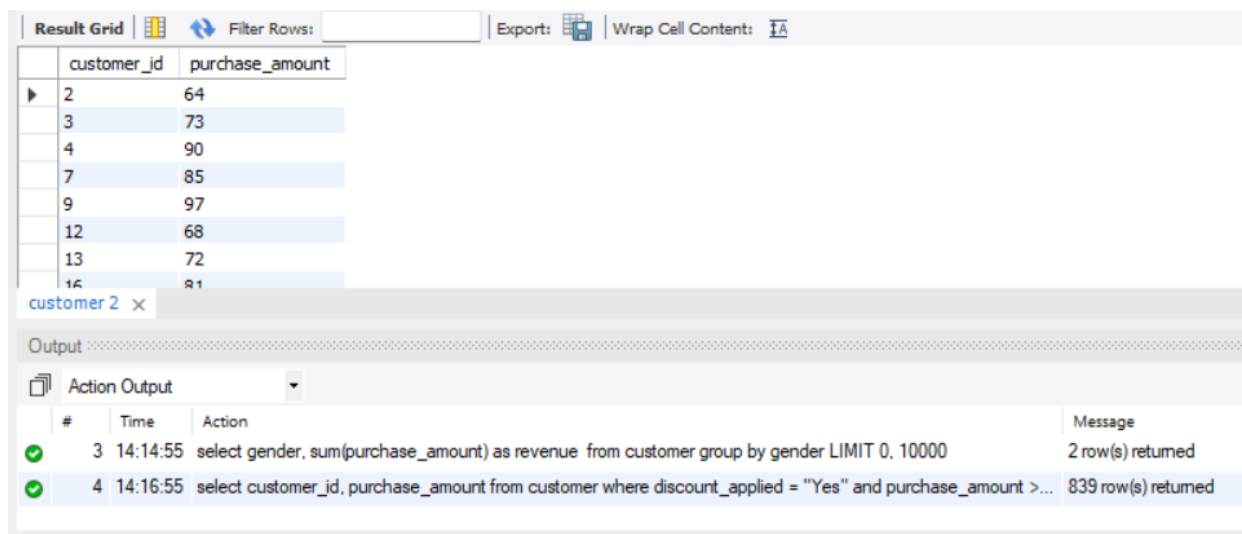
1. **Revenue by Gender** – Compared total revenue generated by male vs. female customers.



A screenshot of a PostgreSQL query result grid. The grid has two columns: 'gender' and 'revenue'. The first row shows 'Male' with a revenue of 157890. The second row shows 'Female' with a revenue of 75191. The grid is titled 'Result Grid' and has a 'Filter Rows' button.

gender	revenue
Male	157890
Female	75191

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



A screenshot of a PostgreSQL query result grid. The grid has two columns: 'customer_id' and 'purchase_amount'. The first row shows customer_id 2 with a purchase_amount of 64. The second row shows customer_id 3 with a purchase_amount of 73. The third row shows customer_id 4 with a purchase_amount of 90. The fourth row shows customer_id 7 with a purchase_amount of 85. The fifth row shows customer_id 9 with a purchase_amount of 97. The sixth row shows customer_id 12 with a purchase_amount of 68. The seventh row shows customer_id 13 with a purchase_amount of 72. The eighth row shows customer_id 14 with a purchase_amount of 81. The grid is titled 'Result Grid' and has a 'Filter Rows' button. Below the grid, there is an 'Output' section with a table showing the results of the queries. The first row shows a successful query: 'select gender, sum(purchase_amount) as revenue from customer group by gender LIMIT 0, 10000' with a message '2 row(s) returned'. The second row shows a successful query: 'select customer_id, purchase_amount from customer where discount_applied = "Yes" and purchase_amount >...' with a message '839 row(s) returned'.

customer_id	purchase_amount
2	64
3	73
4	90
7	85
9	97
12	68
13	72
14	81

#	Time	Action	Message
3	14:14:55	select gender, sum(purchase_amount) as revenue from customer group by gender LIMIT 0, 10000	2 row(s) returned
4	14:16:55	select customer_id, purchase_amount from customer where discount_applied = "Yes" and purchase_amount >...	839 row(s) returned



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



A screenshot of a PostgreSQL query result grid. The grid has two columns: 'item_purchased' and 'avg_review_rating'. The first row shows 'Gloves' with an avg_review_rating of 3.86. The second row shows 'Sandals' with an avg_review_rating of 3.84. The third row shows 'Boots' with an avg_review_rating of 3.82. The fourth row shows 'Hat' with an avg_review_rating of 3.8. The fifth row shows 'Handbag' with an avg_review_rating of 3.78. The grid is titled 'Result Grid' and has a 'Filter Rows' button.

item_purchased	avg_review_rating
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.

Result Grid			 Filter Rows:	
	shipping_type	Average_purchase_amt		
▶	Express	60.48		
	Standard	58.46		

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

Result Grid



Filter Rows:

Export:


Wrap Cell Content

	subscription_status	Total_customer	Average_spend	Total_revenue
▶	No	2847	59.87	170436
	Yes	1053	59.49	62645

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

Result Grid			 Filter Rows:
	item_purchased	discount_rate	
▶	Hat	50.00	
	Sneakers	49.66	
	Coat	49.07	
	Sweater	48.17	
	Pants	47.37	

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

Result Grid		 Filter Rows:
	total_count	customer_type
▶	3116	Loyal
	701	Returning
	83	New

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

	item_rank	category	item_purchased	total_orders
▶	1	Accessories	Jewelry	171
	2	Accessories	Sunglasses	161
	3	Accessories	Belt	161
	1	Clothing	Blouse	171
	2	Clothing	Pants	171
	3	Clothing	Shirt	169
	1	Footwear	Sandals	160
	2	Footwear	Shoes	150
	3	Footwear	Sneakers	145
	1	Outerwear	Jacket	163
	2	Outerwear	Coat	161

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

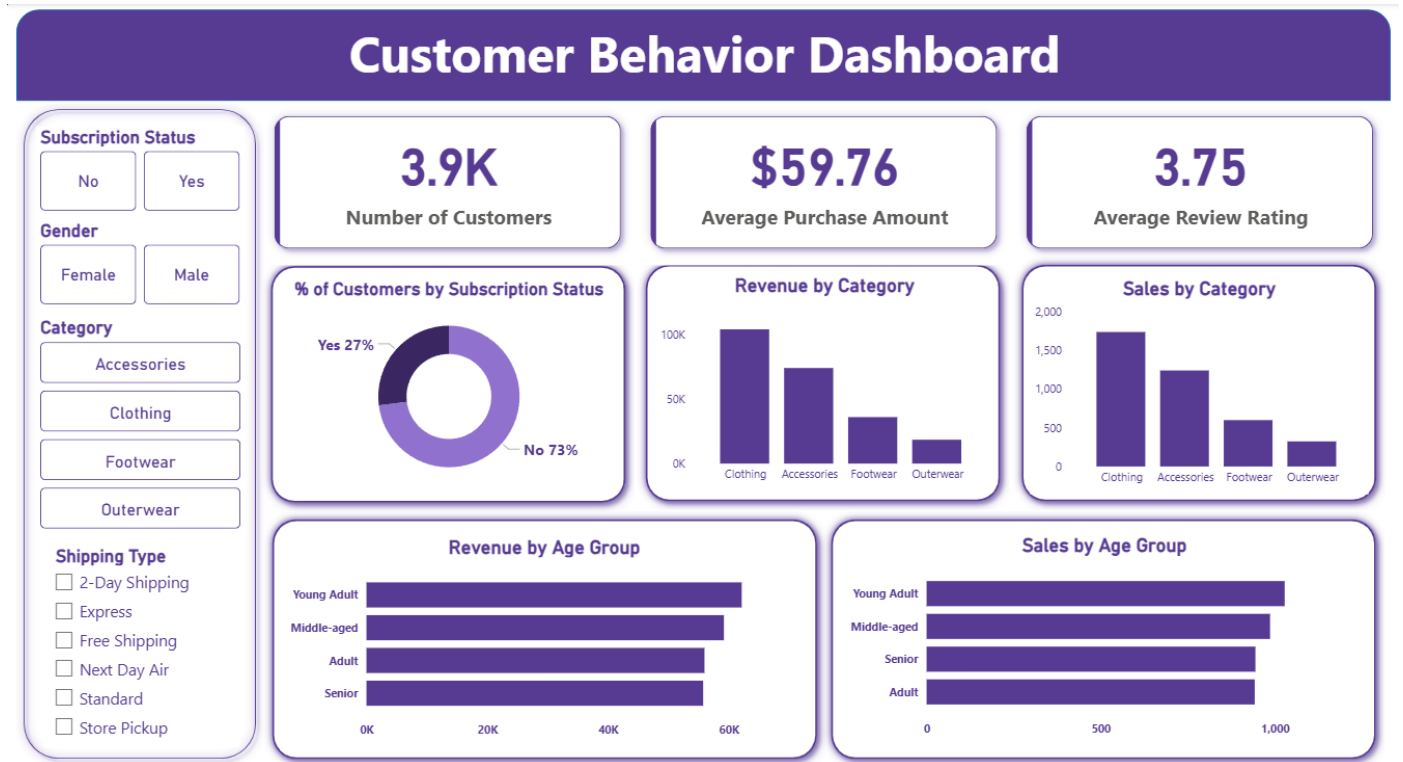
	repeat_buyers	subscription_status
▶	958	Yes
	2518	No

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

	age_group	revenue_contribution
▶	Young Adult	62143
	Middle-aged	59197
	Adult	55978
	Senior	55763

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.