

Project Report

# Customer Shopping Behaviour Analysis

---

*Prepared By*

**Sadmanul Hoque**

## Project Overview

This project analyzes customer shopping behaviour using transactional data from nearly 4,000 purchases across various product categories. The goal is to uncover insights into purchase patterns, customer segments, product preferences to guide strategic business decisions.

## Data Cleaning using Python

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
1	55	Male	Blouse	Clothing	53	Kentucky	L	Gray	Winter	3.1	Yes	Express	Yes	Yes	14	Venmo	Fortnightly
2	19	Male	Sweater	Clothing	64	Maine	L	Maroon	Winter	3.1	Yes	Express	Yes	Yes	2	Cash	Fortnightly
3	50	Male	Jeans	Clothing	73	Massachusetts	S	Maroon	Spring	3.1	Yes	Free Shipping	Yes	Yes	23	Credit Card	Weekly
4	21	Male	Sandals	Footwear	90	Rhode Island	M	Maroon	Spring	3.5	Yes	Next Day Air	Yes	Yes	49	PayPal	Weekly
5	45	Male	Blouse	Clothing	49	Oregon	M	Turquoise	Spring	2.7	Yes	Free Shipping	Yes	Yes	31	PayPal	Annually

- Handled Missing data
- Rename Columns
- Feature Engineering

## Data Analysis using SQL

### Q1: Product Performance

```
WITH sales_data AS(
SELECT
    item_purchased AS product_name,
    SUM(purchase_amount) AS current_sales
FROM customer
GROUP BY
    item_purchased
)
SELECT
    product_name,
    current_sales,
    (SELECT AVG(current_sales) FROM sales_data) AS avarage_sales,
    current_sales - (SELECT AVG(current_sales) FROM sales_data) AS sales_difference,

CASE WHEN current_sales - (SELECT AVG(current_sales) FROM sales_data) > 0 THEN 'Profit'
      WHEN current_sales - (SELECT AVG(current_sales) FROM sales_data) < 0 THEN 'Loss'
      ELSE 'Avarage'
END performance

FROM sales_data;
```

	category	sales	overall_sales	sale_percent
1	Clothing	104264	233081	44.73%
2	Accessories	74200	233081	31.83%
3	Outerwear	18524	233081	7.95%
4	Footwear	36093	233081	15.49%

## Q2: Category Performance

```
WITH sales_data AS(
SELECT
    category,
    SUM(purchase_amount) AS sales
FROM customer
GROUP BY
    category
)
SELECT
    category,
    sales,
    (SELECT SUM(purchase_amount) FROM customer) AS overall_sales,
    CONCAT(ROUND((CAST(sales AS FLOAT) / (SELECT SUM(purchase_amount) FROM customer))
* 100, 2), '%') AS sale_percent
FROM sales_data;
```

	product_name	current_sales	avarage_sales	sales_difference	performance
1	T-shirt	9248	9323	-75	Loss
2	Backpack	8636	9323	-687	Loss
3	Belt	9635	9323	312	Profit
4	Sandals	9200	9323	-123	Loss
5	Boots	9018	9323	-305	Loss
6	Hoodie	8767	9323	-556	Loss
7	Shoes	9240	9323	-83	Loss
8	Jacket	9249	9323	-74	Loss
9	Blouse	10410	9323	1087	Profit
10	Pants	10090	9323	767	Profit
11	Shorts	9433	9323	110	Profit
12	Hat	9375	9323	52	Profit
13	Coat	9275	9323	-48	Loss
14	Scarf	9561	9323	238	Profit
15	Jeans	7548	9323	-1775	Loss
16	Socks	9252	9323	-71	Loss

### Q3: What is the total revenue generated by male vs female customers?

```
SELECT
    gender,
    CONCAT('$', SUM(purchase_amount)) AS total_revenue
FROM customer
GROUP BY gender;
```

Results Messages		
	gender	total_revenue
1	Male	\$157890
2	Female	\$75191

### Q4: Which Customers used discounts but still spent more than the average purchase amount?

```
SELECT
    customer_id,
    CONCAT('$', purchase_amount) AS spent
FROM
    customer
WHERE
    discount_applied = 'Yes'
    AND purchase_amount > (SELECT AVG(purchase_amount) FROM customer)
ORDER BY purchase_amount DESC;
```

	customer_id	spent
1	43	\$100
2	96	\$100
3	194	\$100
4	205	\$100
5	244	\$100
6	249	\$100
7	456	\$100
8	519	\$100
9	582	\$100
10	616	\$100
11	770	\$100
12	862	\$100
13	1209	\$100
14	1301	\$100
15	1406	\$100

**Q5: Which are the top 5 products with the highest average review rating?**

```
SELECT TOP 5
    item_purchased,
    AVG(review_rating) AS avg_rating
FROM customer
GROUP BY item_purchased
ORDER BY
    AVG(review_rating) DESC;
```

	Results	Messages
	item_purchased	avg_rating
1	Gloves	3.86142856223243
2	Sandals	3.84437500983477
3	Boots	3.81874999569522
4	Hat	3.80129870501432
5	Skirt	3.78481012809126

**Q6: Compare the average purchase amounts between standard and express shippings.**

```
SELECT
    shipping_type,
    CONCAT('$', AVG(purchase_amount)) AS avg_purchase_amount
FROM customer
WHERE shipping_type IN('Standard', 'Express')
GROUP BY
    Shipping_type;
```

	Results	Messages
	shipping_type	avg_purchase_amount
1	Express	\$60
2	Standard	\$58

### Q7: Do subscribing customers spend more?

```
SELECT
    subscription_status,
    CONCAT('$', AVG(purchase_amount)) AS spend,
    CONCAT('$', SUM(purchase_amount)) AS total_revenue
FROM customer
GROUP BY subscription_status;
```

	subscription_status	spend	total_revenue
1	Yes	\$59	\$62645
2	No	\$59	\$170436

### Q8: Which product has the highest percentage of the purchases with discount applied?

```
SELECT TOP 5
    item_purchased,
    CONCAT((SUM(CASE WHEN discount_applied = 'Yes' THEN 1 ELSE 0 END) * 100 / COUNT(*)),
    '%') sale_percent
FROM customer
GROUP BY item_purchased
ORDER BY sale_percent DESC;
```

	item_purchased	sale_percent
1	Hat	50%
2	Coat	49%
3	Sneakers	49%
4	Sweater	48%
5	Pants	47%

**Q9: Segment customer into vip, regular, new based on the total previous purchases show the count segment**

```
WITH segment AS(
SELECT
    customer_id,
    previous_purchases,
    CASE WHEN previous_purchases >= 10 THEN 'VIP'
         WHEN previous_purchases >= 5 THEN 'Regular'
         ELSE 'New'
    END customer_segment
FROM customer)
SELECT
    customer_segment,
    COUNT(customer_id) AS total_customer
FROM segment
GROUP BY
    customer_segment
ORDER BY COUNT(customer_id) DESC;
```

	customer_segment	total_customer
1	VIP	3192
2	Regular	371
3	New	337

**Q10: What are the top 3 most purchased products within each category?**

```
WITH sales_data AS(
SELECT
    category,
    item_purchased AS product_name,
    COUNT(item_purchased) AS sale_count
FROM customer
GROUP BY
    item_purchased,
    category
),
product_ranking AS(
SELECT
```



```

category,
product_name,
sale_count,
ROW_NUMBER() OVER (PARTITION BY category ORDER BY sale_count DESC) AS ranking
FROM sales_data
)
SELECT
ranking,
category,
product_name,
sale_count
FROM product_ranking
WHERE
ranking <= 5
GROUP BY
category,
product_name,
sale_count,
ranking;

```

Results		Messages		
	ranking	category	product_name	sale_count
1	1	Accessories	Jewelry	171
2	2	Accessories	Sunglasses	161
3	3	Accessories	Belt	161
4	4	Accessories	Scarf	157
5	5	Accessories	Hat	154
6	1	Clothing	Pants	171
7	2	Clothing	Blouse	171
8	3	Clothing	Shirt	169
9	4	Clothing	Dress	166
10	5	Clothing	Sweater	164
11	1	Footwear	Sandals	160
12	2	Footwear	Shoes	150
13	3	Footwear	Sneakers	145
14	4	Footwear	Boots	144
15	1	Outerwear	Jacket	163
16	2	Outerwear	Coat	161

**Q11: Are customers who have more than 5 previous purchases are likely to subscribe?**

```
SELECT
    subscription_status,
    COUNT(customer_id) AS total_customers
FROM customer
WHERE
    previous_purchases > 5
GROUP BY
    subscription_status;
```

	subscription_status	total_customers
1	Yes	958
2	No	2518

**Q12: What is the revenue contribution of each age group?**

```
SELECT
    age_group,
    CONCAT('$', SUM(purchase_amount)) AS total_revenue
FROM customer
GROUP BY
    age_group
ORDER BY SUM(purchase_amount) DESC
;
```

	age_group	total_revenue
1	young_adult	\$62143
2	middle_aged	\$59197
3	adult	\$55978
4	senior	\$55763