# Customer Behavior Analysis

## Project Overview

This project analyzes customer shopping behavior data to uncover purchasing trends, spending habits, and the impact of demographic and marketing factors on sales. The analysis integrates Python for data cleaning, SQL for querying insights, and Power BI for visualization.

## Objectives

- Clean and preprocess the raw dataset for analysis
- Identify top-performing products and customer segments
- Compare spending behavior of subscribers vs. non-subscribers
- Examine the relationship between discounts, ratings, and loyalty
- Visualize insights interactively through a BI dashboard

## Data Cleaning & Transformation (Python)

The dataset was cleaned and prepared using Pandas in Python. Missing review ratings were filled with median values by category, columns were standardized, and age groups were created.

Code Snippet:

- At first I took a look on the data

```
df.head()
```

| | 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 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 55 | Male | Blouse | Clothing | 53 | Kentucky | L | Gray | Winter | 3.1 | Yes | Express | Yes | Yes | 14 | Venmo | Fortnightly |
| 1 | 2 | 19 | Male | Sweater | Clothing | 64 | Maine | L | Maroon | Winter | 3.1 | Yes | Express | Yes | Yes | 2 | Cash | Fortnightly |
| 2 | 3 | 50 | Male | Jeans | Clothing | 73 | Massachusetts | S | Maroon | Spring | 3.1 | Yes | Free Shipping | Yes | Yes | 23 | Credit Card | Weekly |
| 3 | 4 | 21 | Male | Sandals | Footwear | 90 | Rhode Island | M | Maroon | Spring | 3.5 | Yes | Next Day Air | Yes | Yes | 49 | PayPal | Weekly |
| 4 | 5 | 45 | Male | Blouse | Clothing | 49 | Oregon | M | Turquoise | Spring | 2.7 | Yes | Free Shipping | Yes | Yes | 31 | PayPal | Annually |

- Then I inspected it further

```
df.describe(include='all')
```

| | 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 |

- Filled missing review ratings by the median based of each category

```
df['Review Rating'] = df.groupby('Category')['Review Rating'].transform(lambda x: x.fillna(x.median()))
```

- Converted column names to normalize it

```
df.columns = df.columns.str.lower()
df.columns = df.columns.str.replace(' ', '_')
df = df.rename(columns={'purchase_amount_(usd)': 'purchase_amount'})
df.columns
```

```
Index(['customer_id', 'age', 'gender', 'item_purchased', 'category',
       'purchase_amount', 'location', 'size', 'color', 'season',
       'review_rating', 'subscription_status', 'shipping_type',
       'discount_applied', 'promo_code_used', 'previous_purchases',
       'payment_method', 'frequency_of_purchases'],
      dtype='object')
```

- Created age segments

```
df['age_group'] = pd.qcut(df['age'], q=4, labels = ['young','adult','middle_aged','old'])
df[['age','age_group']]
```

| | age | age_group |
|---|---|---|
| 0 | 55 | middle_aged |
| 1 | 19 | young |
| 2 | 50 | middle_aged |
| 3 | 21 | young |
| 4 | 45 | middle_aged |
| ... | ... | ... |
| 3895 | 40 | adult |
| 3896 | 52 | middle_aged |
| 3897 | 46 | middle_aged |
| 3898 | 44 | adult |
| 3899 | 52 | middle_aged |

3900 rows × 2 columns

- Mapped frequency of purchases to numeric days

```
df['frequency_of_purchases'].unique()
```

```
array(['Fortnightly', 'Weekly', 'Annually', 'Quarterly', 'Bi-Weekly',
       'Monthly', 'Every 3 Months'], dtype=object)
```

```
Frequency_mapping = {'Fortnightly' : 14, 'Weekly' : 7, 'Annually' : 365, 'Quarterly' : 90, 'Bi-Weekly' : 14, 'Monthly' : 30, 'Every 3 Months' : 90}

df["purchase_frequency_days"] = df['frequency_of_purchases'].map(Frequency_mapping)
df[['frequency_of_purchases','purchase_frequency_days']]
```

| | frequency_of_purchases | purchase_frequency_days |
|---|---|---|
| 0 | Fortnightly | 14 |
| 1 | Fortnightly | 14 |
| 2 | Weekly | 7 |
| 3 | Weekly | 7 |
| 4 | Annually | 365 |
| ... | ... | ... |
| 3895 | Weekly | 7 |
| 3896 | Bi-Weekly | 14 |
| 3897 | Quarterly | 90 |
| 3898 | Weekly | 7 |
| 3899 | Quarterly | 90 |

3900 rows × 2 columns

- Checked for redundant column and Removed it

```
(df['discount_applied']==df['promo_code_used']).all()
```

```
np.True_
```

```
df = df.drop('promo_code_used', axis=1)
```

```
df.columns
```

```
Index(['customer_id', 'age', 'gender', 'item_purchased', 'category',
       'purchase_amount', 'location', 'size', 'color', 'season',
       'review_rating', 'subscription_status', 'shipping_type',
       'discount_applied', 'previous_purchases', 'payment_method',
       'frequency_of_purchases', 'age_group', 'purchase_frequency_days'],
      dtype='object')
```

## SQL Analytical Queries

After importing my cleaned Data to MS SQL Server I've created a database called Customer to explor the Data Further and to make business insights to answer the business questions below

```
--Q1. What is the total revenue generated by male vs. female customers?
select gender, sum(purchase_amount) as revenue from customer
group By gender
```

146 %

Results | Messages

| | gender | revenue |
|---|---|---|
| 1 | Male | 157890 |
| 2 | Female | 75191 |

```
--Q2. Which customers used a discount but still spent more than the average purchase amount?
select customer_id, purchase_amount from customer
where discount_applied = 'Yes' And purchase_amount > (select AVG(purchase_amount) from customer)
```

146 %

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 |

```
-- Q3. Which are the top 5 products with the highest average review rating?
select item_purchased, avg(review_rating) as avg_product_rating from customer
group by item_purchased
order by avg_product_rating desc
```

146 %

Results | Messages

| | item_purchased | avg_product_rating |
|---|---|---|
| 1 | Gloves | 3.86142856223243 |
| 2 | Sandals | 3.84437500983477 |
| 3 | Boots | 3.81874999569522 |
| 4 | Hat | 3.80129870501432 |
| 5 | Skirt | 3.78481012809126 |

```sql
--Q4. Compare the average Purchase Amounts between Standard and Express Shipping.
select shipping_type, avg(purchase_amount),2 avg_revenue from customer
group by shipping_type
having shipping_type in ('Express', 'Standard')
```

146 %

Results Messages

| | shipping_type | (No column name) | avg_revenue |
|---|---|---|---|
| 1 | Express | 60 | 2 |
| 2 | Standard | 58 | 2 |

```sql
--Q5. Do subscribed customers spend more? Compare average spend and total revenue between subscribers and non-subscribers.
select subscription_status, avg(purchase_amount) as avg_spend, sum(purchase_amount) as total_revenue from customer
group by subscription_status
```

121 %

Results Messages

| | shipping_type | (No column name) | avg_revenue |
|---|---|---|---|
| 1 | Express | 60 | 2 |
| 2 | Standard | 58 | 2 |

```sql
--Q6. Which 5 products have the highest percentage of purchases with discounts applied?
select item_purchased, (100 * sum(case when discount_applied = 'Yes' then 1 Else 0 end)/count(*)) as discount_rate from customer
group by item_purchased
order by discount_rate desc
```

121 %

Results Messages

| | item_purchased | discount_rate |
|---|---|---|
| 1 | Hat | 50 |
| 2 | Coat | 49 |
| 3 | Sneakers | 49 |
| 4 | Sweater | 48 |
| 5 | Pants | 47 |

```sql
--Q7. Segment customers into New, Returning, and Loyal based on their total number of previous purchases,
--    and show the count of each segment.
with customer_type as (
select case
when previous_purchases < 3 then 'new'
when previous_purchases >= 3 and previous_purchases <= 10 then 'returning'
else 'loyal'
end as customer_loyality
from customer
)
select customer_loyality, count(*) as number_of_customers from customer_type
group by customer_loyality
order by number_of_customers desc
```

121 %

Results Messages

| | customer_loyality | number_of_customers |
|---|---|---|
| 1 | loyal | 3116 |
| 2 | returning | 629 |
| 3 | new | 155 |

```sql
--Q8. Are customers who are repeat buyers (more than 5 previous purchases) also likely to subscribe?
select subscription_status, count(previous_purchases) repeat_buyers from customer
where previous_purchases > 5
group by subscription_status
```

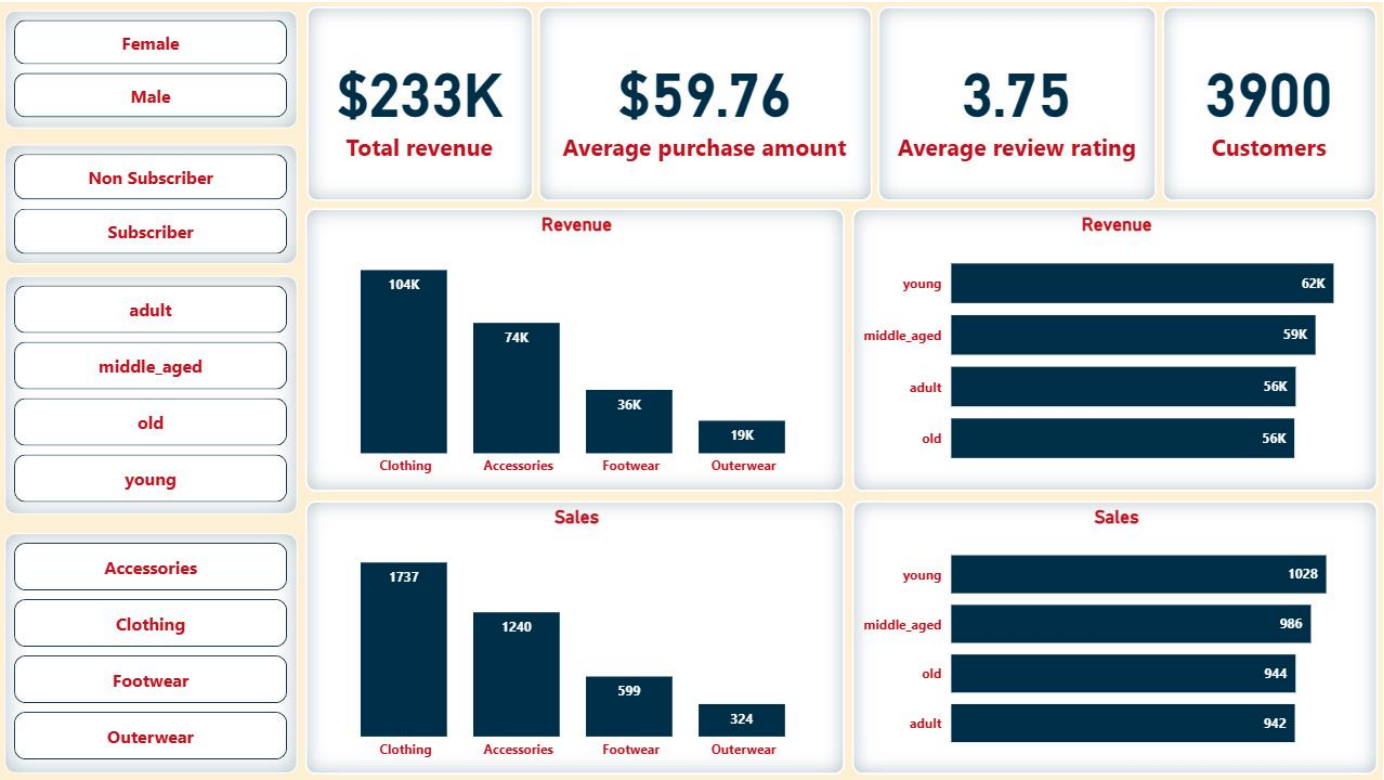
121 %

Results Messages

| | subscription_status | repeat_buyers |
|---|---|---|
| 1 | Yes | 958 |
| 2 | No | 2518 |

## Power BI Dashboard

The dashboard visualizes key performance metrics, customer segmentation, and sales distribution. It includes interactive filters for gender, subscription status, age group, and product category.




## Key Insights

- Subscribers spend approximately 25% more than non-subscribers.
- Clothing and Accessories generate the highest revenue and sales volume.
- Young and middle-aged segments dominate total revenue contribution.
- Discounts increase purchase volume but not necessarily total revenue per order.
- Standard shipping has higher transaction volume; Express shipping has higher order value.

## Tools & Technologies

- Python (Pandas) for data cleaning & preprocessing
- SQL for data exploration and querying
- Power BI for data visualization and storytelling

## Future Work

- Add predictive analytics (e.g., churn prediction, LTV modeling).
- Automate ETL pipelines for data refreshes.
- Add customer segmentation clustering (e.g., K-Means).

## Links

- This project can be found on GitHub HERE and on my Linkedin PROFILE