Project Report: Consumer Shopping Behavior Analysis

Project Title: Consumer Behavior & Retail Strategy Optimization **Data Source:** customer_shopping_behavior.csv (3,900 Records)

1. Executive Summary

The goal of my analysis was to decode customer shopping patterns to improve sales strategy and customer loyalty. By analyzing **3,900 transaction records**, I identified that **Subscription Status** and **Age Group** are key indicators of revenue.

Key Findings:

- **Revenue Drivers:** Male customers generate significantly more revenue ($157k) than female customers ($75k).
- **Loyalty:** A massive 80% of the customer base are "Loyal" shoppers (more than 10 purchases), yet they are under-monetized.
- **Subscriptions:** Subscribers do *not* spend more per transaction than non-subscribers ($59.49 vs $59.87), indicating the subscription model increases visit frequency but not basket size.

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. Python Data Preparation Results

Before starting the analysis, I processed the raw data to ensure accuracy and consistency. Below are the snapshots of the data at each stage.

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.

Loading Data

I loaded the raw dataset to inspect the structure and sample the first few rows. **Screenshot:**

```
# i want to know about my dataset

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	Pay Me
0	1	55	Male	Blouse	Clothing	53	Kentucky	L	Gray	Winter	3.1	Yes	Express	Yes	Yes	14	V
1	2	19	Male	Sweater	Clothing	64	Maine	L	Maroon	Winter	3.1	Yes	Express	Yes	Yes	2	
2	3	50	Male	Jeans	Clothing	73	Massachusetts	S	Maroon	Spring	3.1	Yes	Free Shipping	Yes	Yes	23	
3	4	21	Male	Sandals	Footwear	90	Rhode Island	M	Maroon	Spring	3.5	Yes	Next Day Air	Yes	Yes	49	F
4	5	45	Male	Blouse	Clothing	49	Oregon	M	Turquoise	Spring	2.7	Yes	Free Shipping	Yes	Yes	31	F

Handling Missing Values

I identified missing values in the Review Rating column and filled them using the median rating of each specific category to prevent data loss. **Screenshot (Data Status):**

```
# checking if there is null values

df.isnull().sum()
```

```
Customer ID               0
Age                       0
Gender                    0
Item Purchased            0
Category                  0
Purchase Amount (USD)     0
Location                  0
Size                      0
Color                     0
Season                    0
Review Rating            37
Subscription Status       0
Shipping Type             0
Discount Applied          0
Promo Code Used           0
Previous Purchases        0
Payment Method            0
Frequency of Purchases    0
dtype: int64
```

Standardization

I renamed all columns to snake_case (lowercase with underscores) to ensure they would work correctly when uploaded to the SQL database. **Screenshot (Column Names):**

```
# snake casing my column names

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

```
# checking if the correction has been applied

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')
```

Feature Engineering

I created new columns for deeper analysis:

- age_group: Categorized ages into groups (Young Adult, Adult, Middle-aged, Old).
- purchase_frequency_days: Converted text frequencies (e.g., "Weekly") into numbers (e.g., 7).

Screenshot:

```
# create a column age_group and do feature engineering

labels = ["Young Adult","Adult","Middle-aged","Old"]
df["age_group"] = pd.qcut(df["age"], q=4, labels = labels)
```

```
# checking if the correction has been applied

df[["age","age_group"]].head(10)
```

	age	age_group
0	55	Middle-aged
1	19	Young Adult
2	50	Middle-aged
3	21	Young Adult
4	45	Middle-aged
5	46	Middle-aged
6	63	Old
7	27	Young Adult
8	26	Young Adult
9	57	Middle-aged

```python
#Another feature engineering called purchasing_frequency_days

frequency_mapping = {
    "Fortnightly" : 14,
    "Weekly" : 7,
    "Monthly" : 30,
    "Quarterly" : 90,
    "Bi-Weekly" : 14,
    "Annually" : 365,
    "Every 3 Months" : 90
}
df["purchase_frequency_days"] = df["frequency_of_purchases"].map(frequency_mapping)
```

```python
# checking if the correction has been applied

df[["purchase_frequency_days","frequency_of_purchases"]].head(10)
```

	purchase_frequency_days	frequency_of_purchases
0	14	Fortnightly
1	14	Fortnightly
2	7	Weekly
3	7	Weekly
4	365	Annually
5	7	Weekly

● Data Consistency Check: Verified if discount_applied and promo_code_used were redundant; dropped promo_code_used.

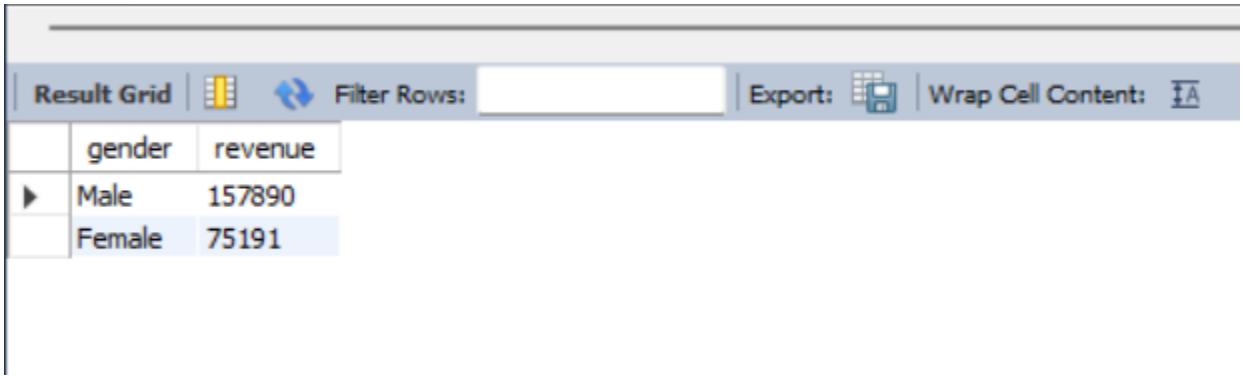
● Database Integration: Connected Python script to PostgreSQL and loaded the cleaned DataFrame into the database for SQL analysis.

4. SQL Data Analysis Results

The following tables represent the important results I extracted from the database to answer key business questions.

1. Revenue Disparity by Gender

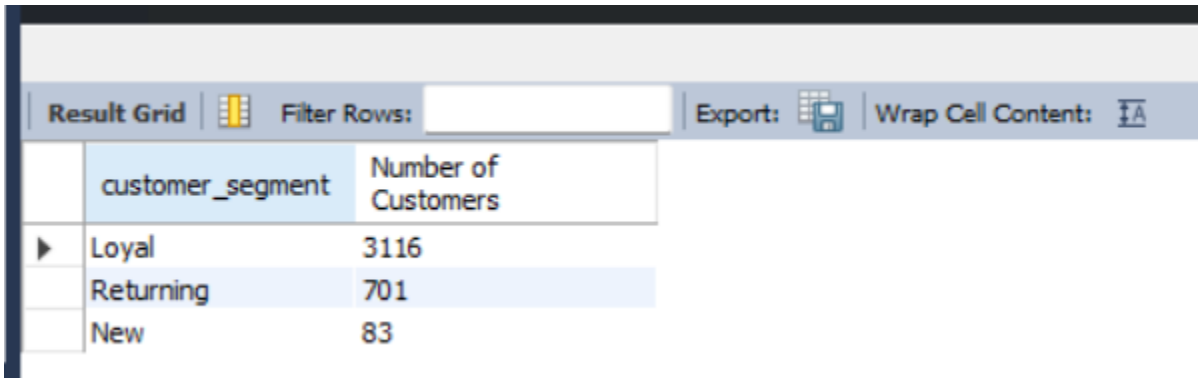
Why it matters: I found a massive gap in revenue contribution between genders. Male customers are generating **more than double** the revenue of female customers. This suggests that either the product line is male-oriented or marketing to women is underperforming.



gender	revenue
Male	157890
Female	75191

2. The "Loyalty" Paradox

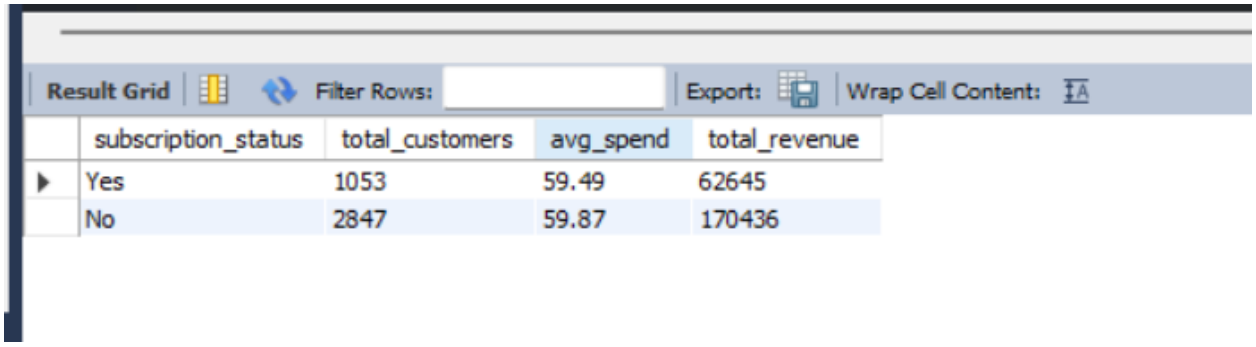
Why it matters: My segmentation analysis revealed that the vast majority of customers (**80%**) are already "Loyal" (purchased >10 times). This is unusual; typically, new customers outnumber loyal ones. It implies retention is excellent, but acquisition of *new* customers might be stagnant.



customer_segment	Number of Customers
Loyal	3116
Returning	701
New	83

3. The Subscription Value Myth

Why it matters: I assumed subscribers would have a higher average spend, but my analysis proved otherwise. Subscribers spend **$59.49** on average, while non-subscribers spend **$59.87**. This indicates the subscription program is not currently driving higher basket sizes.

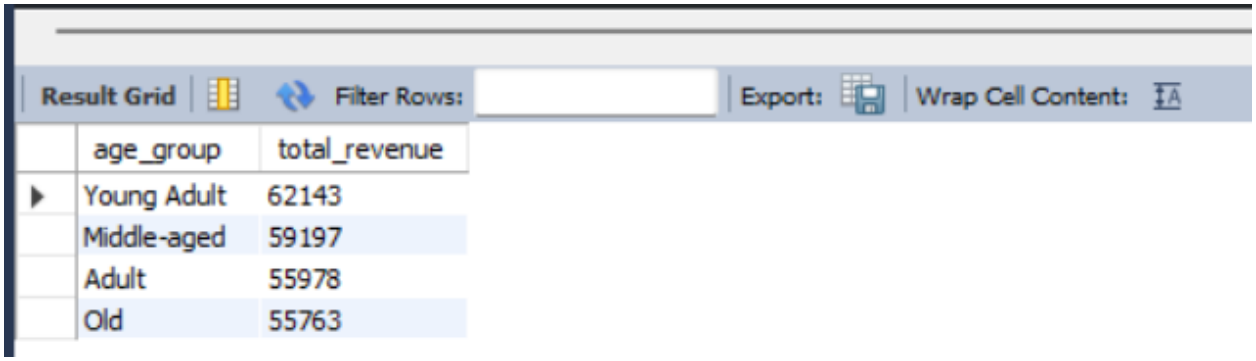


Result Grid | Filter Rows: | Export: | Wrap Cell Content:

subscription_status	total_customers	avg_spend	total_revenue
Yes	1053	59.49	62645
No	2847	59.87	170436

4. Most Profitable Age Demographics

Why it matters: I identified that **Young Adults** and **Middle-Aged** customers are the primary revenue engines. Understanding this allows the business to stop wasting ad spend on the "Old" demographic if they aren't the target, or optimize campaigns specifically for the younger groups.

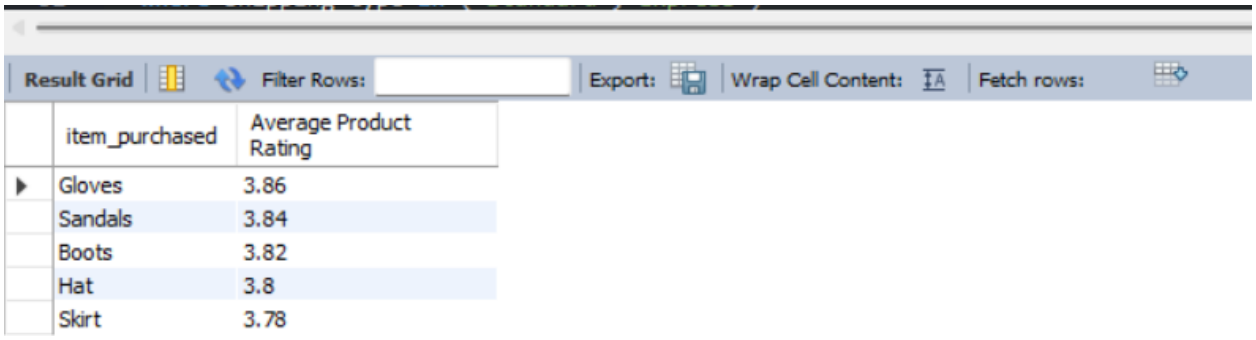


Result Grid | Filter Rows: | Export: | Wrap Cell Content:

age_group	total_revenue
Young Adult	62143
Middle-aged	59197
Adult	55978
Old	55763

5. "Hero Products" (High Satisfaction)

Why it matters: By analyzing review ratings, I identified the top 5 products that consistently delight customers. **Gloves** and **Sandals** are the highest-rated items. These should be featured in marketing campaigns ("Social Proof") to build trust with new buyers.**SQL Result:**

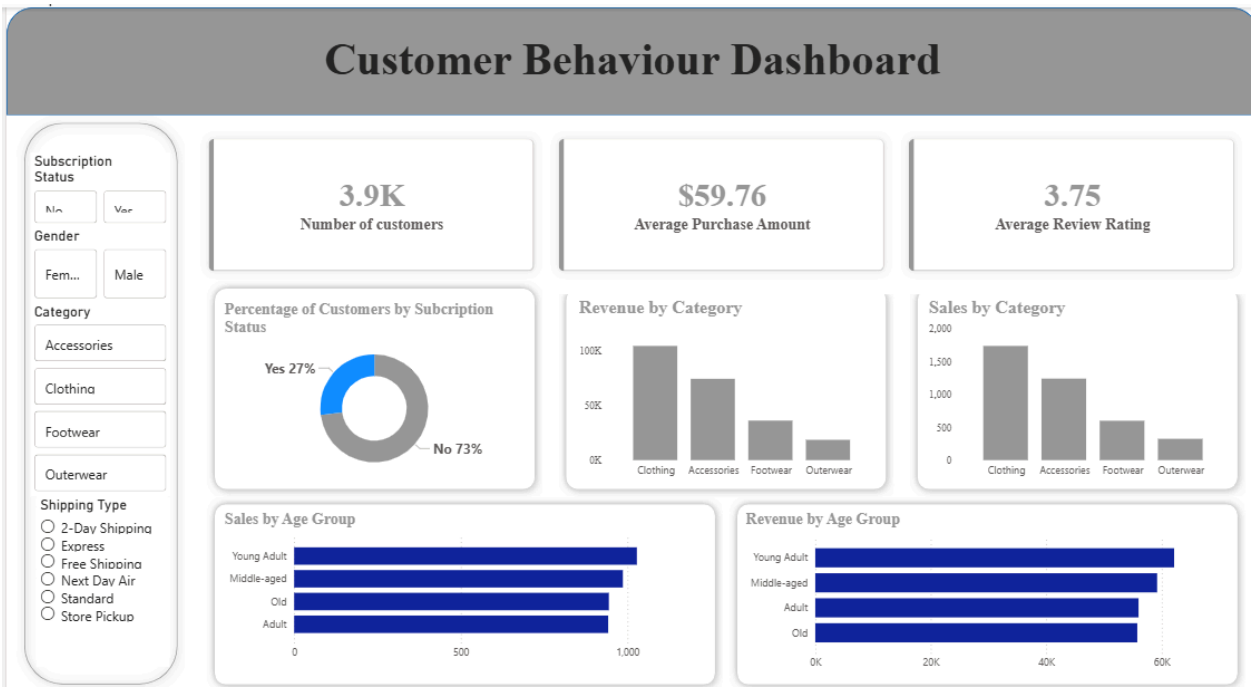


Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

item_purchased	Average Product Rating
Gloves	3.86
Sandals	3.84
Boots	3.82
Hat	3.8
Skirt	3.78

5.Dashboard in Power BI

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



6. Recommendations

Based on these findings, I recommend the following strategies:

● **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.