Customer_Shopping_Behaviour_Analysis

December 25, 2025

[35]:
```
import pandas as pd
from sqlalchemy import create_engine
import psycopg2
```

[36]:
```
df = pd.read_csv('customer_shopping_behavior.csv')
```

[37]:
```
df.head()
```

[37]:

	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

[38]:
```
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
```

[39]:
```
df.describe()
```

[39]:

	Customer ID	Age	Purchase Amount (USD)	Review Rating	\
count	3900.000000	3900.000000	3900.000000	3863.000000	
mean	1950.500000	44.068462	59.764359	3.750065	
std	1125.977353	15.207589	23.685392	0.716983	
min	1.000000	18.000000	20.000000	2.500000	
25%	975.750000	31.000000	39.000000	3.100000	
50%	1950.500000	44.000000	60.000000	3.800000	
75%	2925.250000	57.000000	81.000000	4.400000	
max	3900.000000	70.000000	100.000000	5.000000	

	Previous Purchases
count	3900.000000
mean	25.351538
std	14.447125
min	1.000000
25%	13.000000
50%	25.000000
75%	38.000000
max	50.000000

[40]:
```
df.isnull().sum()
```

[40]:
```
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
```

[41]:
```
df['Review Rating'] = df.groupby('Category')['Review Rating'].transform(lambda␣
↪x:x.fillna(x.median()))
```

[42]:
```
df.isnull().sum()
```

[42]:
```
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              0
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
```

[43]:
```
df.columns = df.columns.str.lower()
df.columns = df.columns.str.replace(' ','_')
df = df.rename(columns={'purchase_amount_(usd)':'purchase_amount'})
```

[44]:
```
df.columns
```

[44]:
```
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')
```

[45]:
```
# Create a columns age_group
labels = ['Young Adult','Adult', 'Middle-aged', 'Senior']
df ['age_group'] = pd.qcut(df['age'], q=4, labels = labels)
```

[46]:
```
df[['age','age_group']].head(10)
```

[46]:

	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	Senior
7	27	Young Adult
8	26	Young Adult
9	57	Middle-aged

[47]:
```
# Create Columns Purchased _frequency_days
frequency_mapping = {
    'Fortnightly': 14,
    'Weekly': 7,
    'Monthly': 30,
    'Quarterly': 90,
    'Bi-Weekly': 14,
    'Annually': 365,
    'Every 3 Months': 90
}

df['Purchased _frequency_days'] = df['frequency_of_purchases'].
 ↪map(frequency_mapping)
```

[48]:
```
df[['Purchased _frequency_days','frequency_of_purchases']].head(10)
```

[48]:
```
   Purchased _frequency_days frequency_of_purchases
0                         14            Fortnightly
1                         14            Fortnightly
2                          7                 Weekly
3                          7                 Weekly
4                        365                Annually
5                          7                 Weekly
6                         90              Quarterly
7                          7                 Weekly
8                        365                Annually
9                         90              Quarterly
```

[49]:
```
df[['discount_applied','promo_code_used']].head(10)
```

[49]:
```
  discount_applied promo_code_used
0              Yes             Yes
1              Yes             Yes
2              Yes             Yes
3              Yes             Yes
4              Yes             Yes
5              Yes             Yes
6              Yes             Yes
7              Yes             Yes
8              Yes             Yes
9              Yes             Yes
```

[50]:
```
(df['discount_applied'] ==df['promo_code_used']).all()
```

[50]:
```
np.True_
```

[51]:
```
df = df.drop('promo_code_used', axis=1)
```

[52]:
```
df.columns
```

[52]:
```
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', 'Purchased _frequency_days'],
      dtype='object')
```

[54]:
```
from sqlalchemy import create_engine, text

# Step 1: Connection parameters
username = "postgres"
password = "Dharmikp1908"
host = "localhost"
port = 5432
```

```
database = "customer_behaviour"

# Step 2: Create engine
engine = create_engine(
    f"postgresql+psycopg2://{username}:{password}@{host}:{port}/{database}"
)

# Step 3: Test connection
try:
    with engine.connect() as conn:
        result = conn.execute(text("SELECT 1"))
        print("Connection successful:", result.scalar() == 1)
except Exception as e:
    print("Connection failed:", e)

# Step 4: Load DataFrame into PostgreSQL (only if connection works)
table_name = "customer"
df.to_sql(table_name, engine, if_exists="replace", index=False)
print(f"Data successfully loaded into table '{table_name}' in database␣
 ↪'{database}'.")
```

```
Connection successful: True
Data successfully loaded into table 'customer' in database 'customer_behaviour'.
```

[ ]:

[ ]:

[ ]:

[ ]:

[ ]: