Detailed Report of Clustering Results

Clustering Metrics and Insights:

1. Cluster Composition:

Total Number of Clusters: 5

Cluster Size Breakdown:

- Cluster 0: 12 customers
- Cluster 1: 59 customers
- Cluster 2: 71 customers (Largest cluster)
- Cluster 3: 36 customers
- Cluster 4: 21 customers

2. Clustering Performance Metrics:

Davies-Bouldin Index (DB Index)

- **Value**: 0.9245800474301245
- The Davies-Bouldin Index measures the average similarity between each cluster and its most similar cluster.

3. Cluster Size Analysis:

Cluster Distribution Insights

The cluster sizes reveal interesting patterns:

- **Cluster 2 (71 customers)**: Represents the majority of the customer base
- **Cluster 1 (59 customers)**: Second largest group
- **Cluster 3 (36 customers)**: Mid-sized cluster
- **Cluster 4 (21 customers)**: Little Smaller group
- **Cluster 0 (12 customers)**: Smallest cluster, might represent unique or outlier customers

4. Implications of Cluster Sizes:

Customer Segmentation Considerations

1. **Large Segments**
    - Clusters 1 and 2 contain the large no. of customers.
    - These groups likely represent the most typical customer behaviours.
2. **Small Segments**

- Clusters 0 and 4 are smaller.
- May represent Unique customer types with potential high-value or high-risk groups from the rest.

5. Methodological Notes:

Feature Engineering of aggregate customer data:

Code:

```
customer_data = merged.groupby('CustomerID').agg({
        'TotalValue': 'sum',
        'TransactionID': 'count',
        'ProductID': 'nunique'
    }).reset_index()
```

Clustering Approach:

- **Algorithm**: K-Means Clustering

```
# K-Means clustering
num_clusters = 5
kmeans = KMeans(n_clusters=num_clusters, random_state=42)
clusters = kmeans.fit_predict(scaled_data)
customer_data['Cluster'] = clusters


# Evaluating clusters
db_index = davies_bouldin_score(scaled_data, clusters)
print(f"DB Index: {db_index}")
```

- **Preprocessing**: StandardScaler for feature normalization

```
# Standardize the data
scaler = StandardScaler()
scaled_data = scaler.fit_transform(customer_data.drop(columns=['CustomerID']))
```

- **Number of Clusters**: Determined through initial analysis – taken 5

6. Clustering visualizations:

1. K-Means clustering of customers:

```
plt.figure(figsize=(8, 6))
sns.scatterplot(
   x=scaled_data[:, 0], y=scaled_data[:, 1],
   hue=clusters, palette="Set2", s=100
)
plt.title("Customer Clusters (K-Means)")
plt.xlabel("Feature 1")
plt.ylabel("Feature 2")
plt.legend(title="Cluster")
plt.show()
```

Plot:

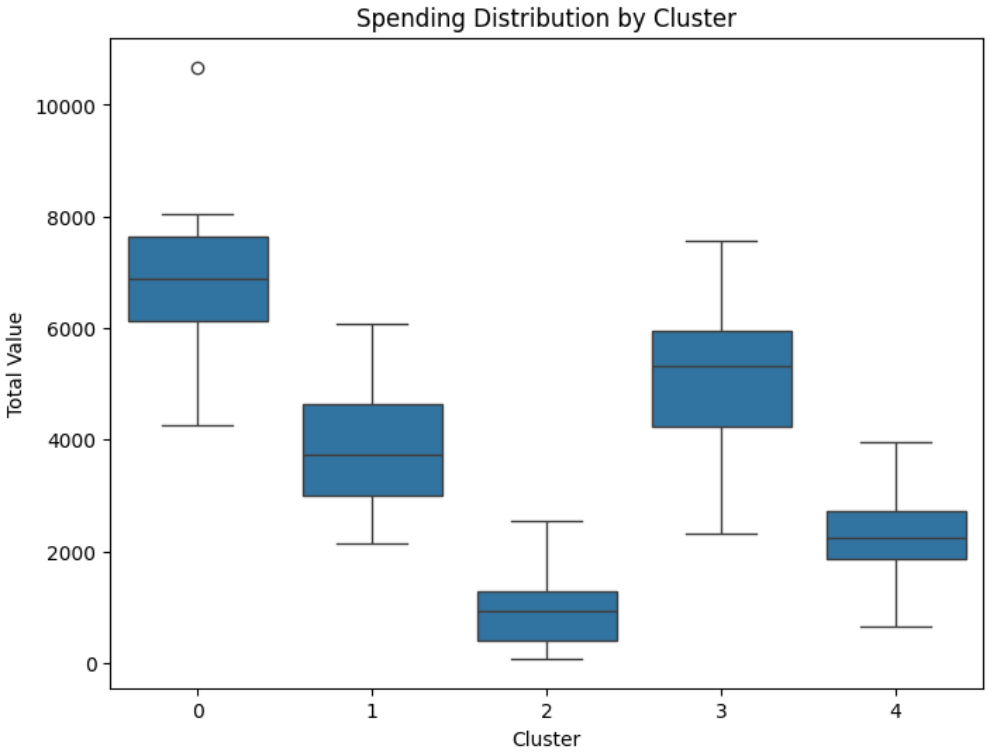
Insights:

- Cluster 1 and 4 are the dominant groups and may represent average or majority customer behaviours.
- Cluster 2 contain inactive or low-engagement customers who are comparatively little lower in number.
- Cluster 3 consists of moderately active or mid-spending customers, who are moderately populated.

2. **Spending Distribution by Cluster:**

```
plt.figure(figsize=(8, 6))
sns.boxplot(data=customer_data, x='Cluster', y='TotalValue')
plt.title("Spending Distribution by Cluster")
plt.xlabel("Cluster")
plt.ylabel("Total Value")
plt.show()
```

Plot:



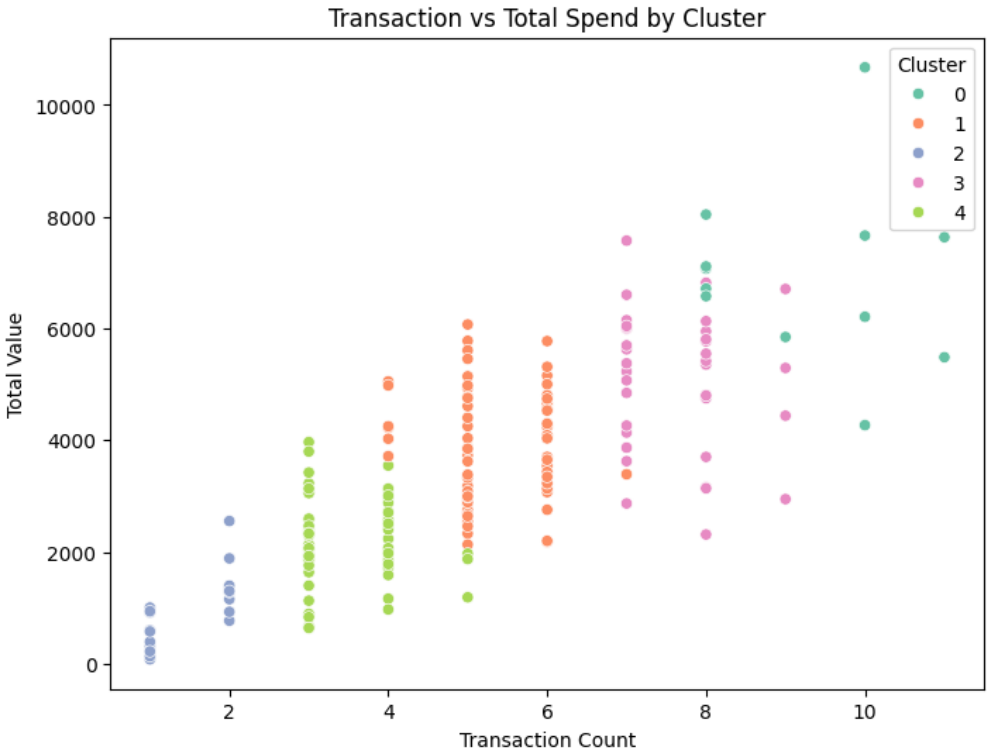
Insights:

- Cluster 0 has the highest spending customers, with a median total value above 6000.
- Cluster 2 has the lowest median spending, suggesting that customers in this cluster spend the least on average.
- Clusters 1, 3, and 4 show more moderate spending levels, their box sizes suggest moderate variability in spending within each of these groups.
- There's a single data point plotted as a circle above the box for Cluster 0. This is an outlier, representing an individual in this cluster with exceptionally high spending compared to the rest of the group.

3. **Transaction vs Total Spend by Cluster:**

```
plt.figure(figsize=(8, 6))
sns.scatterplot(x=customer_data['TransactionID'], y=customer_data['TotalValue'],
hue=customer_data['Cluster'], palette="Set2")
plt.title("Transaction vs Total Spend by Cluster")
plt.xlabel("Transaction Count")
plt.ylabel("Total Value")
plt.legend(title="Cluster")
plt.show()
```

Plot:



Insights:

- There's a positive correlation between transaction count and total value. As the number of transactions increases, the total spending tends to increase as well.
- Cluster 0 generally exhibits higher total spending, even with a moderate number of transactions.
- Cluster 2 consistently shows lower spending across all transaction counts. Even with a higher number of transactions, their total spending remains relatively low, indicating smaller purchase sizes.