Customer Segmentation Report

1. Introduction

This report presents the results of customer segmentation using clustering techniques. The goal is to group customers into distinct clusters based on their profiles and transaction behaviors. DBSCAN was chosen as the primary clustering algorithm for its ability to identify arbitrary-shaped clusters and handle noise effectively. Additionally, Agglomerative Clustering and Gaussian Mixture Models (GMM) were used for comparison.

2. Methodology

2.1 Data Sources

- **Customers.csv**: Customer profile information.
- **Transactions.csv**: Transaction history, including transaction value and product details.
- **Products.csv**: Product categories linked to transactions.

2.2 Preprocessing Steps

- Aggregated transaction data to derive key features:
    - **AvgTransactionValue**: Average transaction value per customer.
    - **PurchaseFrequency**: Number of transactions per customer.
    - **DominantProductCategory**: Most frequently purchased product category per customer.
    - **DistinctProductCount**: Number of unique products purchased.
- Merged transaction features with customer profiles.
- Encoded categorical variables (e.g., Region, DominantProductCategory).
- Scaled numerical features using MinMaxScaler for normalization.

2.3 Clustering Algorithms

- **DBSCAN**:
    - Parameters Tuned: eps and min_samples.
    - Evaluation Metrics: Davies-Bouldin Index (DB Index) and Silhouette Score.
- **Gaussian Mixture Models (GMM)**:
    - Probabilistic clustering approach.
    - Evaluated using DB Index and Silhouette Score.
- **Agglomerative Clustering**:
    - Hierarchical clustering technique.
    - Evaluated using DB Index and Silhouette Score.
    -

- **K-Means Clustering**:
    - Evaluated using DB Index and Silhouette Score.

3. Results

3.1 Clustering Metrics

Algorithm	DB Index	Silhouette Score
DBSCAN (eps=0.4)	0.381	0.725
DBSCAN (eps=0.5)	0.514	0.651
GMM	1.396	0.273
Agglomerative	1.515	0.263
K-Means	1.428	0.257

The DBSCAN configuration with eps = 0.4 yielded the best results based on both DB Index and Silhouette Score.

GMM, Agglomerative Clustering, and K-Means showed relatively poor performance compared to DBSCAN.

3.2 DBSCAN Cluster Summary (eps = 0.4)

- **Number of Clusters**: 9 (excluding noise points).
- **Noise Points**: Represented by cluster -1.
- **Cluster Distribution**: Visualized using t-SNE (see Figure 1 below).

3.3 t-SNE Visualization

The clusters were visualized in two dimensions using t-SNE. Each point represents a customer, with colors indicating cluster assignments. Noise points (-1) are also shown.

(Figure 1: t-SNE visualization of DBSCAN clusters with eps = 0.4)

3.4 Key Insights

- Clusters are well-separated and compact, indicating clear customer segments.
- Noise points may represent unique customer behaviors or errors in the data.

4. Conclusion

The DBSCAN clustering with eps = 0.4 provided high-quality segmentation based on compactness (DB Index) and separation (Silhouette Score). Comparisons with GMM, Agglomerative Clustering, and K-Means confirmed the robustness of DBSCAN for this dataset. The analysis highlights distinct customer groups and noise points, offering a strong foundation for targeted business strategies.

Appendix

Clustering Parameters:

- DBSCAN: eps = 0.4, min_samples = 10
- GMM: Default settings with optimal cluster count derived from evaluation.
- Agglomerative: Default settings with optimal cluster count derived from evaluation.
- K-Means: Default settings with optimal cluster count derived from evaluation.

Metrics Definitions:

- **Davies-Bouldin Index**: Measures the average similarity ratio of intra-cluster to inter-cluster distances.
- **Silhouette Score**: Measures how similar a data point is to its own cluster compared to other clusters.

Code and Visualizations: The Python code used for clustering and visualizations is included in the attached script.