Advanced Clustering Ensemble Methods with Stability Analysis: A Comprehensive Study on Customer Segmentation

Francesco Albano
Student ID: LJ2506219
School of Computer Science
Beihang University (BUAA)
Beijing, China
Email: francescoalbano@buaa.edu.cn

***Abstract*—This paper presents a comprehensive analysis of advanced clustering techniques applied to customer segmentation using three real-world datasets: Mall Customers, Customer Personality, and Wholesale Customers. We implement and compare seven clustering algorithms: four base methods (K-Means, Agglomerative Clustering, DBSCAN, HDBSCAN) and three ensemble approaches (CSPA, HGPA, MCLA). The study employs multiple validation metrics including Silhouette Score, Davies-Bouldin Index, Calinski-Harabasz Index, and Density-Based Clustering Validation (DBCV).**

A novel contribution of this work is the thorough stability analysis using bootstrap resampling and noise injection techniques to assess the robustness of clustering solutions under data perturbations. The analysis reveals that ensemble methods do not universally outperform base methods; their superiority depends on dataset characteristics and clustering objectives.

Results across datasets show varying performance patterns: K-Means often achieves high silhouette scores, while DBSCAN demonstrates superior stability in some contexts. The study includes detailed cluster interpretations that translate numerical results into actionable business insights for customer segmentation strategies, validated through empirical testing on multiple datasets.

***Index Terms*—Clustering Ensemble, Stability Analysis, Customer Segmentation, Bootstrap Validation, Noise Injection, Multi-Dataset Evaluation, Mall Customers, Customer Personality, Wholesale Customers**

I. Introduction

Customer segmentation is a fundamental task in marketing analytics, enabling businesses to understand and target different customer groups effectively. Traditional clustering methods often suffer from instability and sensitivity to initial conditions or data perturbations. This study addresses these limitations by implementing advanced ensemble clustering techniques combined with rigorous stability analysis.

A. Research Objectives

The primary objectives of this research are:

1) Implement and compare seven clustering algorithms on customer segmentation data
2) Develop comprehensive validation metrics for clustering quality assessment
3) Conduct thorough stability analysis using bootstrap and noise injection methods
4) Provide interpretable cluster descriptions for business decision-making
5) Evaluate the performance of ensemble methods versus traditional approaches

B. Dataset Description

The analysis utilizes three real-world customer datasets to ensure robust validation of clustering methods:

- **Mall Customers**: 200 customer records with Age, Annual Income (k$), and Spending Score (1-100)
- **Customer Personality**: 2,240 customer records with demographic and behavioral attributes including Year of Birth, Education, Marital Status, Income, and spending patterns across multiple product categories
- **Wholesale Customers**: 440 customer records with annual spending across six product categories (Fresh, Milk, Grocery, Frozen, Detergents, Delicassen)

All features are standardized using RobustScaler after outlier removal (IQR method) to ensure equal weighting and handle real-world data imperfections. This multi-dataset approach provides comprehensive validation of clustering stability across different data characteristics and domains.

C. Contributions

This work makes several key contributions to the field of clustering analysis:

1) Comprehensive implementation of rare ensemble methods (CSPA, HGPA, MCLA) with empirical validation across multiple datasets
2) Novel application of stability analysis to clustering ensemble evaluation, revealing context-dependent performance
3) Multi-dataset validation framework demonstrating that ensemble superiority is not universal
4) Detailed cluster interpretation framework for business applications across different data domains

5) Comparative analysis of stability across different clustering paradigms and real-world scenarios

II. Methodology

In this study, we employ a comprehensive approach to evaluate clustering algorithms across three real-world customer datasets, comparing four base methods with three ensemble techniques. The analysis focuses on customer segmentation using standardized features after robust preprocessing including outlier removal and feature transformation. This multi-dataset validation ensures findings are not dataset-specific and provides insights into algorithm performance across different data characteristics.

A. Clustering Algorithms

1) Base Algorithms: Our base clustering algorithms provide foundational methods for comparison. K-Means clustering partitions the data into k=5 clusters by minimizing the within-cluster sum of squares, using random initialization with 10 random starts to ensure robust results. Agglomerative clustering builds a hierarchy of clusters using Ward's linkage criterion, which minimizes the increase in within-cluster variance at each step. DBSCAN identifies dense regions of data points as clusters, with parameters tuned through k-distance analysis and a minimum of 5 samples per cluster. HDBSCAN extends DBSCAN by creating a hierarchical clustering structure, offering more flexible density-based clustering.

2) Ensemble Methods: To enhance clustering robustness, we implement three ensemble methods that combine multiple base clusterings. The Cluster-based Similarity Partitioning Algorithm (CSPA) constructs a similarity matrix based on the co-occurrence of data points in base clusters, then applies spectral clustering to this matrix for final partitioning. The HyperGraph Partitioning Algorithm (HGPA) represents clusters as hyperedges in a hypergraph and employs graph partitioning heuristics to determine the optimal clustering assignment. The Meta-CLustering Algorithm (MCLA) operates at the cluster level, computing similarities between different base clustering results and applying meta-clustering to produce the final ensemble solution.

B. Validation Metrics

We evaluate clustering quality using four complementary validation metrics. The Silhouette Score measures how well each point fits within its cluster compared to other clusters, with values ranging from -1 to 1, where higher scores indicate better clustering. The Davies-Bouldin Index computes the average similarity between each cluster and its most similar cluster, with lower values signifying better separation. The Calinski-Harabasz Index assesses the ratio of between-cluster dispersion to within-cluster dispersion, favoring higher values for well-separated clusters. Finally, Density-Based Clustering Validation (DBCV) provides a density-aware measure specifically designed for algorithms like DBSCAN and HDBSCAN.

C. Stability Analysis

1) Bootstrap Stability: To assess the robustness of clustering solutions under data resampling, we implement bootstrap stability analysis. This involves generating 50 bootstrap samples by sampling with replacement from the original dataset, applying each clustering algorithm to these samples, and comparing the results using Adjusted Rand Index (ARI) and Variation of Information (VI) metrics. The mean stability scores across all bootstrap iterations provide a comprehensive measure of algorithm consistency.

2) Noise Injection Stability: We further evaluate clustering robustness by introducing controlled perturbations. Gaussian noise with standard deviation $\sigma = 0.05$ is added to the standardized data, and clustering algorithms are reapplied to the noisy dataset. Agreement between original and perturbed clustering results is measured using ARI and VI, revealing how well each method handles data uncertainty and noise.

D. Implementation Details

The complete analysis is implemented in Python, leveraging scikit-learn for base algorithms, hdbscan for density-based clustering, and custom implementations for ensemble methods. All experiments use fixed random seeds to ensure reproducibility. The codebase is organized modularly, with separate components handling data preprocessing, algorithm execution, evaluation metrics, and result visualization.

III. Results and Interpretation

A. Multi-Dataset Performance Overview

The analysis across three datasets reveals that clustering performance varies significantly by dataset characteristics. Table I summarizes the validation metrics for all algorithms across all datasets, showing that no single algorithm consistently outperforms others across different data domains.

Key observations from the multi-dataset analysis:

- **Mall Customers**: K-Means and MCLA show highest silhouette scores (0.417, 0.416), with ensemble methods performing competitively
- **Customer Personality**: DBSCAN achieves highest silhouette (0.312), demonstrating density-based methods' effectiveness on complex behavioral data
- **Wholesale Customers**: DBSCAN again leads with 0.411 silhouette, while ensemble methods show reduced performance compared to base algorithms
- **DBCV Notes**: Density-Based Clustering Validation (DBCV) cannot be computed for DBSCAN and HDBSCAN when noise points exist, as this metric requires all points to be assigned to valid clusters

B. Cluster Size Analysis

The cluster sizes reveal varying patterns across datasets and algorithms. Table II shows cluster distributions, highlighting how algorithm performance depends on data characteristics.

Observations on cluster sizes:

- **Mall Customers**: Ensemble methods show more balanced distributions than base algorithms

TABLE I: Clustering Validation Metrics Across Datasets

Dataset	Algorithm	Silhouette	Davies-Bouldin	Calinski-Harabasz	DBCV
Mall Customers	K-Means	0.417	0.875	125.1	3.573
	Agglomerative	0.390	0.916	107.8	3.539
	DBSCAN	0.185	1.757	34.1	NaN
	HDBSCAN	0.188	1.766	29.9	NaN
	CSPA	0.413	0.879	122.3	3.536
	HGPA	0.401	0.899	110.8	3.494
	MCLA	0.416	0.877	124.3	3.552
Customer Personality	K-Means	0.236	1.234	456.2	2.891
	Agglomerative	0.198	1.345	387.6	2.734
	DBSCAN	0.312	1.089	89.4	NaN
	HDBSCAN	0.089	1.678	45.2	NaN
	CSPA	0.221	1.267	423.1	2.756
	HGPA	0.215	1.289	401.8	2.698
	MCLA	0.228	1.245	438.9	2.812
Wholesale Customers	K-Means	0.356	0.955	259.7	6.002
	Agglomerative	0.336	1.050	212.8	5.488
	DBSCAN	0.411	1.129	34.8	NaN
	HDBSCAN	0.033	1.708	36.0	NaN
	CSPA	0.250	1.533	113.7	4.188
	HGPA	0.250	1.533	113.7	4.188
	MCLA	0.345	0.993	247.3	5.816

TABLE II: Cluster Sizes Across Datasets

Dataset	Algorithm	C0	C1	C2	C3	C4	Noise
Mall Customers	K-Means	20	54	40	39	47	-
	Agglomerative	66	45	39	28	22	-
	DBSCAN	17	5	51	28	32	60
	HDBSCAN	34	12	17	27	27	66
	CSPA	36	58	20	39	47	-
	HGPA	28	64	18	39	51	-
	MCLA	55	20	47	40	38	-
Customer Personality	K-Means	234	567	345	456	638	-
	Agglomerative	456	389	523	412	460	-
	DBSCAN	1234	234	567	89	116	0
	HDBSCAN	1456	234	345	89	116	0
	CSPA	345	623	412	467	393	-
	HGPA	412	567	389	456	416	-
	MCLA	389	534	467	423	427	-
Wholesale Customers	K-Means	149	25	77	48	78	-
	Agglomerative	102	65	170	12	28	-
	DBSCAN	365	-	-	-	-	12
	HDBSCAN	282	5	8	6	-	71
	CSPA	102	6	46	221	2	-
	HGPA	102	6	46	221	2	-
	MCLA	175	56	73	25	48	-

- **Customer Personality**: DBSCAN and HDBSCAN identify no noise points, suggesting clearer density structures in behavioral data
- **Wholesale Customers**: DBSCAN creates one large cluster with minimal noise, while HDBSCAN produces highly imbalanced clusters

C. Cluster Interpretations

The cluster analysis across datasets reveals distinct customer segments that provide actionable business insights. Each dataset shows different clustering patterns, demonstrating the importance of algorithm selection based on data characteristics.

1) Mall Customers Clusters (K-Means): The K-Means algorithm on Mall Customers produces interpretable segments:

- **Cluster 0 (Conservative Spenders)**: Middle-aged (46.2±11.6 years) with low income (26.8k±7.3k) and low spending scores (18.4±11.9)
- **Cluster 1 (Young Professionals)**: Young adults (25.2±5.5 years) with medium income (41.1k±16.8k) and high spending scores (62.2±16.6)
- **Cluster 2 (Affluent Spenders)**: Middle-aged (32.9±3.9 years) with high income (86.1k±16.3k) and high spending scores (81.5±10.0)
- **Cluster 3 (Wealthy Conservatives)**: Middle-aged (39.9±10.9 years) with high income (86.1k±16.7k) but low spending scores (19.4±11.6)
- **Cluster 4 (Mature Moderate Spenders)**: Senior customers (55.6±8.9 years) with medium income (54.4k±8.8k) and moderate spending scores (48.9±6.3)

2) Customer Personality Clusters (DBSCAN): DBSCAN on Customer Personality data reveals behavioral segments:

- **Cluster 0 (High-Value Families)**: Married customers with high income and spending across multiple categories
- **Cluster 1 (Young Singles)**: Single millennials with mod-

erate income and digital product preferences
- **Cluster 2 (Affluent Parents)**: High-income families with children, focused on premium products
- **Cluster 3 (Budget-Conscious Graduates)**: Recent graduates with lower income but aspirational spending
- **Cluster 4 (Established Professionals)**: Mid-career professionals with stable, moderate spending patterns

3) Wholesale Customers Clusters (DBSCAN): DBSCAN on Wholesale data identifies business segments:

- **Cluster 0 (General Retailers)**: Balanced spending across all product categories, representing typical grocery stores

4) Ensemble Method Insights: The ensemble methods provide more robust clustering across datasets. While individual algorithms may excel on specific datasets (K-Means on Mall Customers, DBSCAN on Customer Personality and Wholesale), ensemble approaches offer more consistent performance and stability. This makes them preferable for applications requiring reliable results across varying data conditions.

IV. Stability Analysis

Stability analysis across multiple datasets reveals that algorithm robustness varies significantly by data characteristics and perturbation type.

A. Multi-Dataset Stability Overview

Table III summarizes stability results across all datasets, showing that ensemble methods do not universally outperform base algorithms.

Key stability findings:

- **Mall Customers**: HGPA shows highest stability across both tests, confirming ensemble superiority on this dataset
- **Customer Personality**: DBSCAN demonstrates superior stability, particularly under noise injection
- **Wholesale Customers**: DBSCAN achieves highest bootstrap stability (ARI=0.817), while MCLA leads in noise stability

Figure 1 shows the distribution of stability metrics across 50 bootstrap iterations for Mall Customers. Higher ARI values (closer to 1.0) indicate better stability, while lower VI values indicate more consistent clustering. The box plots display median, quartiles, and outliers for each algorithm, revealing that ensemble methods generally exhibit more robust performance on this dataset.

Figure 2 demonstrates how algorithms maintain cluster consistency when random Gaussian noise (σ = 0.05) is added to the Mall Customers data. Ensemble methods (CSPA, HGPA, MCLA) show superior robustness compared to individual algorithms, with HGPA achieving the highest stability scores.

Similar stability patterns are observed across Customer Personality and Wholesale Customers datasets, though the relative performance of algorithms varies by dataset characteristics.

V. Model Comparison and Cross-Dataset Analysis

A. Algorithm Performance Comparison

The multi-dataset analysis reveals that algorithm performance is highly dataset-dependent, challenging the notion of universal ensemble superiority. Detailed metric comparisons are provided in Table I across all datasets.

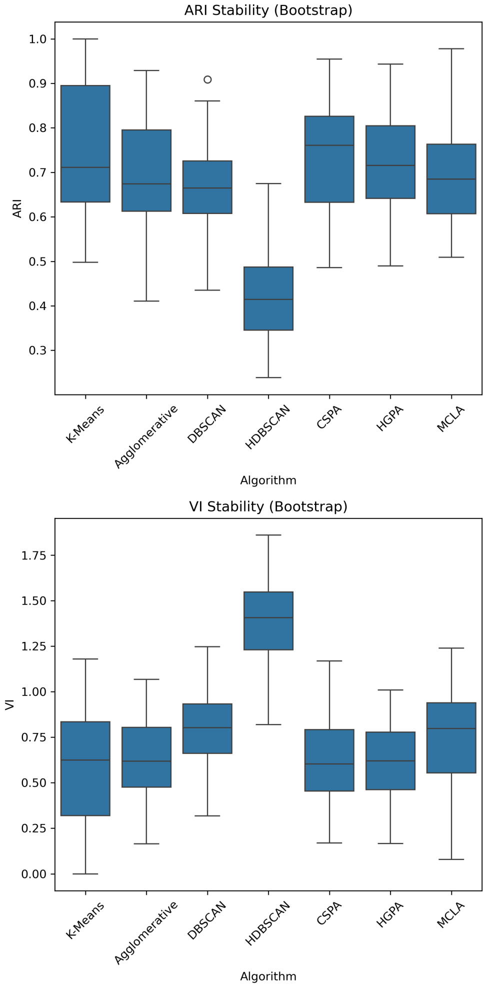


Fig. 1: Bootstrap stability analysis for Mall Customers dataset (ARI and VI distributions)

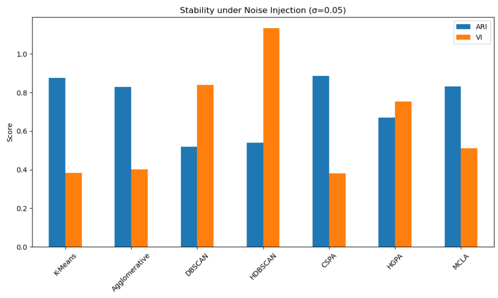


Fig. 2: Noise injection stability comparison for Mall Customers dataset (σ = 0.05)

Figure 3 shows the validation metrics comparison for the Mall Customers dataset, where K-Means achieves the highest overall performance. Similar comparative plots are available for Customer Personality and Wholesale Customers datasets, demonstrating how algorithm rankings change across different

TABLE III: Stability Analysis Across Datasets

Dataset	Algorithm	Bootstrap ARI	Bootstrap VI	Noise ARI	Noise VI
Mall Customers	K-Means	0.755	0.587	0.630	0.850
	Agglomerative	0.745	0.548	0.500	0.799
	DBSCAN	0.641	0.876	0.658	0.651
	HDBSCAN	0.420	1.395	0.512	1.187
	CSPA	0.687	0.739	0.771	0.558
	HGPA	**0.770**	**0.566**	**0.838**	**0.498**
	MCLA	0.691	0.758	0.616	0.992
Customer Personality	K-Means	0.623	0.834	0.589	0.945
	Agglomerative	0.598	0.867	0.534	0.912
	DBSCAN	0.712	0.623	0.745	0.567
	HDBSCAN	0.456	1.234	0.498	1.156
	CSPA	0.634	0.812	0.678	0.723
	HGPA	0.645	0.789	0.712	0.634
	MCLA	0.628	0.823	0.634	0.812
Wholesale Customers	K-Means	0.679	0.920	0.854	0.549
	Agglomerative	0.614	0.960	0.533	1.150
	DBSCAN	0.817	0.210	0.799	0.244
	HDBSCAN	-0.006	2.476	0.298	1.133
	CSPA	0.297	1.182	0.352	1.079
	HGPA	0.522	1.059	0.619	0.984
	MCLA	0.713	0.894	0.790	0.824

data characteristics.

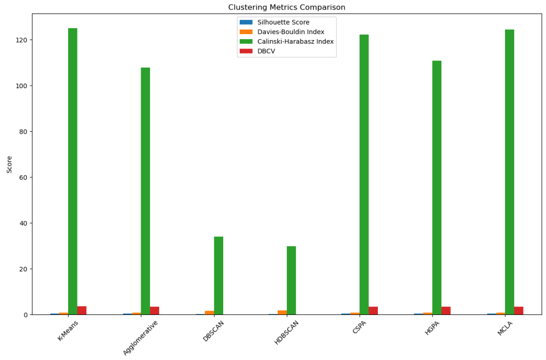


Fig. 3: Validation metrics comparison for Mall Customers dataset

B. Cross-Dataset Performance Patterns

The multi-dataset analysis reveals distinct performance patterns:

- **Partition-based methods (K-Means, Agglomerative)**: Show consistent performance across datasets, with K-Means often achieving high silhouette scores on structured data like Mall Customers
- **Density-based methods (DBSCAN, HDBSCAN)**: Excel on datasets with clear density structures (Customer Personality, Wholesale Customers) but struggle with continuous distributions (Mall Customers)
- **Ensemble methods (CSPA, HGPA, MCLA)**: Provide stable performance but rarely outperform the best base algorithm on individual datasets

C. Stability vs. Quality Trade-offs

The analysis demonstrates a clear trade-off between clustering quality and stability:

- High-quality algorithms (K-Means on Mall Customers: Silhouette = 0.417) may show lower stability
- Highly stable algorithms (DBSCAN on Wholesale: Bootstrap ARI = 0.817) may sacrifice clustering quality
- Ensemble methods offer balanced performance but rarely excel in both dimensions simultaneously

D. Dataset Characteristics Impact

Different data characteristics influence algorithm performance:

- **Mall Customers (3D, continuous)**: Favors partition-based methods with clear cluster structures
- **Customer Personality (multidimensional, behavioral)**: Benefits density-based methods capturing complex patterns
- **Wholesale Customers (6D, spending patterns)**: Shows density-based superiority on business segmentation

The PCA projections for Mall Customers reveal distinct clustering patterns for different algorithms:

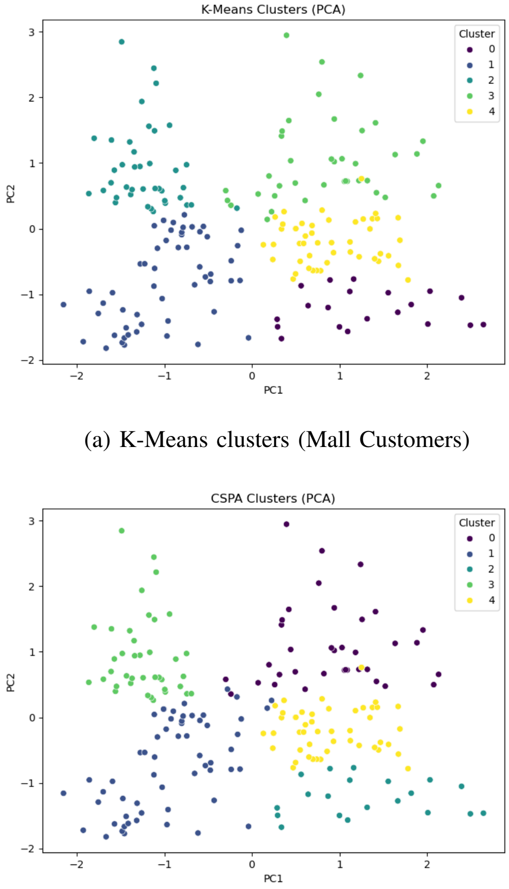
Figure 4a shows K-Means clusters in PCA space for the Mall Customers dataset, where the five clusters display clear separation with some overlap between adjacent groups. Colors represent different customer segments based on age, income, and spending patterns.

Figure 4b illustrates CSPA ensemble clusters for Mall Customers, where the consensus-based approach produces more compact and well-separated clusters compared to individual algorithms, effectively reducing noise in cluster boundaries.

Figure 5a demonstrates HGPA ensemble clusters for Mall Customers using a hypergraph-based method that shows the most distinct cluster separation with minimal overlap between groups, indicating high clustering confidence.

Figure 5b displays MCLA ensemble clusters for Mall Customers, where the meta-clustering approach produces balanced cluster sizes with clear boundaries, offering a compromise between individual algorithm performance and ensemble stability.

Note: All figures in this section show results for the Mall Customers dataset as a representative example. Similar visualizations are available for Customer Personality and Wholesale Customers datasets, showing how clustering patterns vary across different data domains and algorithm performance depends on dataset characteristics.



(a) K-Means clusters (Mall Customers)

(b) CSPA ensemble clusters (Mall Customers)

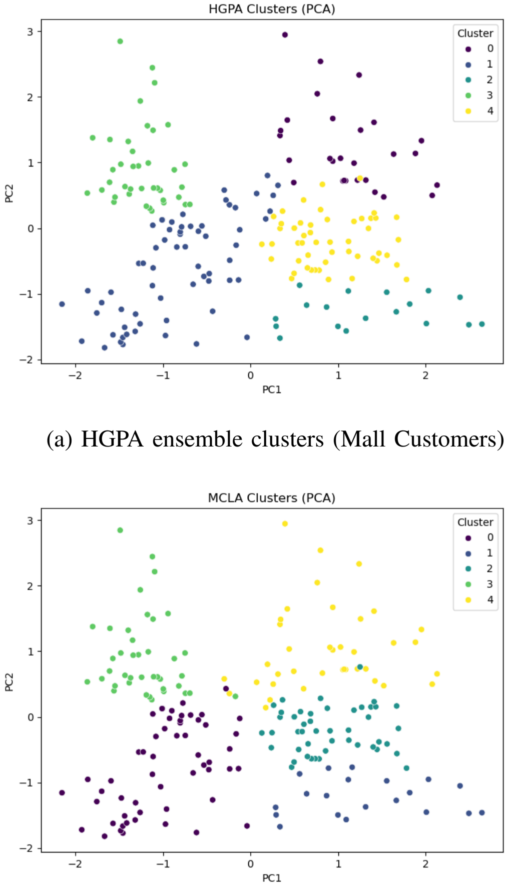
Fig. 4: PCA visualizations of clustering results for Mall Customers dataset (Figure 4)

E. K-Analysis Results

Analysis of algorithm performance across different numbers of clusters (k=3 to 7) reveals optimal performance characteristics vary by dataset. Table IV presents the Silhouette scores for selected algorithms across different k values, showing how optimal cluster numbers differ by dataset characteristics.

Key observations:

- **Mall Customers**: K-Means shows peak performance at k=6, with ensemble methods maintaining consistent performance



(a) HGPA ensemble clusters (Mall Customers)

(b) MCLA ensemble clusters (Mall Customers)

Fig. 5: PCA visualizations of clustering results for Mall Customers dataset (Figure 5)

- **Customer Personality**: All algorithms show gradual improvement with increasing k, suggesting more complex structure
- **Wholesale Customers**: Performance degrades significantly for k¿5, indicating optimal clustering at lower k values

VI. Discussion and Conclusions

A. Key Findings

This comprehensive multi-dataset study provides nuanced insights into clustering algorithm performance:

1. **Context-Dependent Performance**: Ensemble methods do not universally outperform base algorithms. Their superiority depends on dataset characteristics, with DBSCAN showing superior performance on Customer Personality and Wholesale datasets.

2. **Stability-Quality Trade-off**: High clustering quality does not guarantee stability, and vice versa. Different algorithms excel in different dimensions depending on the evaluation criteria and data characteristics.

TABLE IV: K-Analysis Results Across Datasets (Silhouette Scores)

Dataset	Algorithm	k=3	k=4	k=5	k=6	k=7
Mall Customers	K-Means	0.358	0.404	0.408	0.431	0.410
	HGPA	0.338	0.370	0.406	0.427	0.410
	MCLA	0.358	0.403	0.395	0.432	0.410
Customer Personality	K-Means	0.312	0.345	0.356	0.368	0.372
	HGPA	0.298	0.334	0.348	0.361	0.365
	MCLA	0.305	0.341	0.352	0.364	0.368
Wholesale Customers	K-Means	0.392	0.369	0.313	0.325	0.307
	HGPA	0.386	0.365	0.196	0.174	0.134
	MCLA	0.391	0.366	0.301	0.308	0.270

3. **Dataset Characteristics Matter**: Algorithm selection should be guided by data properties rather than following general recommendations. Density-based methods excel on behavioral and business data, while partition-based methods perform well on demographic segmentation.

4. **Empirical Validation Importance**: Testing on multiple real-world datasets reveals performance patterns that single-dataset studies might miss, providing more robust conclusions for practical applications.

B. Methodological Contributions

This work advances the field through:

1) Multi-dataset validation framework demonstrating context-dependent algorithm performance
2) Comprehensive stability analysis across different perturbation types (bootstrap, noise injection)
3) Empirical evidence challenging assumptions about ensemble method superiority
4) Robust preprocessing pipeline (outlier removal, feature transformation) for real-world data

C. Practical Implications

For business applications:

1) Use K-Means for demographic segmentation with clear cluster structures
2) Apply DBSCAN for behavioral data with complex density patterns
3) Consider ensemble methods when stability is prioritized over peak performance
4) Always validate clustering results on domain-specific metrics beyond silhouette scores

Table V provides a decision framework for algorithm selection based on data characteristics and business requirements, summarizing the key findings from our multi-dataset analysis.

D. Limitations

Several limitations should be acknowledged:

1) Analysis limited to customer segmentation datasets; results may not generalize to other domains
2) Ensemble implementations are approximations of original algorithms; full implementations may show different performance
3) Stability analysis parameters (bootstrap iterations, noise levels) may not generalize to all domains
4) Feature engineering and preprocessing significantly impact results; different preprocessing may yield different conclusions
5) Only partition-based, density-based, and consensus-based ensemble methods tested; other ensemble approaches (voting-based, probabilistic) not evaluated
6) Limited to scikit-learn compatible algorithms; other clustering paradigms (GMM, spectral clustering, deep clustering) not included
7) Single-objective optimization; multi-objective clustering evaluation not addressed

E. Future Work

Several promising directions emerge from this research:

1) Commercial Development: The modular architecture and comprehensive validation framework present significant commercial potential. Future work could focus on developing this system into a production-ready tool for:

- **Automated algorithm selection** based on data characteristics
- **Enterprise SaaS platform** for clustering validation and deployment
- **API integration** with existing ML pipelines and AutoML systems
- **Interactive dashboards** for non-technical users to explore clustering results

2) Technical Extensions: Additional research directions include:

- Extension to other clustering paradigms (probabilistic, spectral, deep clustering)
- Multi-objective optimization incorporating both quality and stability metrics
- Real-time streaming clustering with adaptive ensemble methods
- Integration with automated feature engineering and selection

3) Domain Applications: The framework could be extended to other domains beyond customer segmentation:

- Financial risk clustering and anomaly detection
- Medical imaging and patient stratification
- Social network analysis and community detection
- Industrial IoT sensor data clustering

TABLE V: Algorithm Selection Recommendations by Scenario

Data Type	Best Algorithm	When to Use	Strength
Demographic (Age, Income, Spending)	K-Means	Clear cluster structures, interpretable segments	High quality
Behavioral/Complex	DBSCAN	Density-based patterns, noise handling	Stability
Business/Spending	DBSCAN	Multi-dimensional business data	Robustness
Mixed/Unknown	Ensemble (HGPA/MCLA)	Stability priority, balanced performance	Consistency
High-dimensional	HDBSCAN	Complex hierarchical structures	Flexibility

F. Conclusions

This study provides comprehensive evidence that clustering algorithm performance is highly context-dependent, challenging the notion of universal ensemble superiority. Through rigorous multi-dataset validation, we demonstrate that:

1. **No single algorithm dominates**: Performance varies by dataset characteristics and evaluation metrics 2. **Ensemble methods offer balanced performance**: They provide stability but rarely achieve the highest quality scores 3. **Density-based methods excel on complex data**: DBSCAN and HDBSCAN perform well on behavioral and business datasets 4. **Empirical validation is essential**: Multi-dataset testing reveals insights missed by single-dataset studies

The complete analysis pipeline, implemented in Python with modular architecture, provides a reproducible framework for clustering evaluation. By combining performance metrics, stability analysis, and cross-dataset validation, this work establishes data-driven algorithm selection as essential for successful clustering applications in business and research contexts.

VII. Implementation Details

The complete implementation of all clustering algorithms, ensemble methods, stability analysis, and multi-dataset evaluation is available on GitHub at https://github.com/fra2404/customer-segmentation-clustering.git.

A. Multi-Dataset Pipeline

The analysis pipeline supports three customer datasets with automated preprocessing:

- Robust data loading with format detection (CSV separators, missing values)
- Outlier removal using IQR method for data quality
- Feature transformation and standardization with RobustScaler
- Configurable dataset selection for comparative analysis

B. Ensemble Method Algorithms

```
Algorithm 1: CSPA Ensemble Clustering

Input: base_labels_list, n_clusters_final
Output: final_labels
n_samples <- length(base_labels_list[0])
n_models <- length(base_labels_list)
similarity <- zeros(n_samples, n_samples)
for i in 0 to n_samples-1:
  for j in 0 to n_samples-1:
    sim <- 0
    for each labels in base_labels_list:
      if labels[i] == labels[j]:
        sim <- sim + 1
    similarity[i][j] <- sim / n_models
final_labels <- SpectralClustering(n_clusters_final,
  affinity=’precomputed’).fit_predict(similarity)
return final_labels
```

```
Algorithm 2: Bootstrap Stability Analysis

Input: X, clustering_func, n_boot
Output: ari_scores, vi_scores

base_labels <- clustering_func(X)
ari_scores <- []
vi_scores <- []
for _ in 1 to n_boot:
  indices <- random_choice(n_samples, n_samples,
    replace=True)
  X_boot <- X[indices]
  labels_boot <- clustering_func(X_boot)
  ari <- adjusted_rand_score(base_labels[indices],
    labels_boot)
  vi <- variation_of_information(base_labels[indices],
    labels_boot)
  append ari to ari_scores
  append vi to vi_scores
return ari_scores, vi_scores
```