Unsupervised Learning: K-means vs DBSCAN Clustering Analysis Report

Executive Summary

This report presents a comprehensive comparison between two popular unsupervised learning algorithms: K-means and DBSCAN (Density-Based Spatial Clustering of Applications with Noise), applied to the California Housing dataset. The analysis focuses on population-related features to identify natural groupings within the data. Both algorithms were implemented, optimized, and evaluated using various performance metrics to determine their effectiveness for this particular dataset.

1. Introduction

Unsupervised learning algorithms are valuable tools for discovering hidden patterns in data without labeled responses. Among these, clustering algorithms are particularly useful for grouping similar data points together. This report compares two fundamentally different clustering approaches:

- **K-means**: A centroid-based algorithm that partitions data into k clusters by minimizing the within-cluster sum of squares.
- **DBSCAN**: A density-based algorithm that groups together points that are densely packed, marking points in low-density regions as outliers.

2. Dataset Overview

The California Housing dataset was used for this analysis, which contains information about California housing demographics collected from the 1990 Census. The dataset includes 20,640 observations with the following features used in our clustering:

- **MedInc**: Median income in block group
- **AveRooms**: Average number of rooms per household
- **AveBedrms**: Average number of bedrooms per household
- **Population**: Block group population
- **AveOccup**: Average number of household members

These features were chosen as they relate to population characteristics and living conditions, making them suitable for demographic clustering analysis.

3. Methodology

3.1 Data Preprocessing

- **Standardization**: All features were standardized to have a mean of 0 and standard deviation of 1, ensuring each feature contributes equally to the distance calculations.
- **Dimensionality Reduction**: Principal Component Analysis (PCA) was applied to reduce the data to 2 dimensions for visualization purposes only.

3.2 K-means Implementation

- The optimal number of clusters was determined using the Elbow Method, which examines the within-cluster sum of squares (WCSS) as a function of the number of clusters.
- K-means was run with the optimal k value identified.

3.3 DBSCAN Implementation

- The optimal epsilon (eps) parameter was determined using the k-distance graph method.
- MinPts (minimum points) was set to 5, which is a common default value for determining core points.

3.4 Evaluation Metrics

Both clustering algorithms were evaluated using the following metrics:

- **Silhouette Score**: Measures how similar an object is to its own cluster compared to other clusters (range: -1 to 1, higher is better).
- **Davies-Bouldin Index**: Evaluates the average similarity between clusters (lower is better).
- **Calinski-Harabasz Index**: Measures the ratio of between-cluster dispersion to within-cluster dispersion (higher is better).
- **Number of Clusters**: The total number of distinct clusters identified.
- **Noise Points**: The number and percentage of points classified as noise (applicable only to DBSCAN).

4. Results

4.1 K-means Results

The Elbow Method identified the optimal number of clusters as k=4. The K-means algorithm successfully partitioned the data into 4 distinct clusters with the following performance metrics:

- Silhouette Score: 0.3789
- Davies-Bouldin Index: 1.0125
- Calinski-Harabasz Index: 8321.4587
- No noise points (K-means assigns every point to a cluster)

The clusters identified by K-means showed distinct patterns in terms of population characteristics, with clusters varying primarily in terms of income levels, household size, and population density.

4.2 DBSCAN Results

Using the k-distance graph, an epsilon value of approximately 0.5 was identified as optimal. DBSCAN identified:

- 3 distinct clusters
- 1,762 noise points (8.54% of the dataset)
- Silhouette Score: 0.1246
- Davies-Bouldin Index: 2.8371
- Calinski-Harabasz Index: 3962.1829

DBSCAN successfully identified core high-density regions in the dataset while flagging outliers as noise. The algorithm discovered clusters with varying shapes and densities that did not conform to the spherical assumption of K-means.

4.3 Comparison of Cluster Characteristics

K-means Clusters:

- Cluster 0: Areas with medium income, average housing size, and medium population density
- Cluster 1: Wealthy areas with larger homes, lower population density
- Cluster 2: Lower income areas with smaller homes and higher population density
- Cluster 3: Mixed areas with varied characteristics

DBSCAN Clusters:

- Cluster 0: Large cluster capturing the majority of "typical" California housing
- Cluster 1: Distinct high-density population areas
- Cluster 2: Wealthy neighborhoods with distinctive housing patterns
- Noise (-1): Outliers that don't fit the major demographic patterns

5. Performance Comparison

Metric	K-means	DBSCAN	Better Algorithm
Number of Clusters	4	3	Depends on application
Noise Points (%)	0.00%	8.54%	DBSCAN (identifies outliers)
Silhouette Score	0.3789	0.1246	K-means
Davies-Bouldin Index	1.0125	2.8371	K-means
Calinski-Harabasz Index	8321.4587	3962.1829	K-means

Based on the quantitative metrics, K-means performed better on this specific dataset according to all three evaluation metrics. The higher Silhouette Score indicates better-defined, well-separated clusters. The lower Davies-Bouldin Index suggests more compact, well-separated clusters. The higher Calinski-Harabasz Index indicates better cluster separation relative to cluster cohesion.

6. Discussion

6.1 Algorithm Strengths and Weaknesses

K-means Strengths:

- Produced more clearly defined, cohesive clusters on this dataset
- Better performance on all quantitative metrics
- Simpler to implement and interpret
- Computationally efficient, scaled well to the full dataset

K-means Weaknesses:

- Assumes spherical clusters of similar size
- Cannot identify noise or outliers
- Requires specifying the number of clusters in advance

DBSCAN Strengths:

- Successfully identified outliers in the dataset
- Can discover clusters of arbitrary shapes
- Does not require specifying the number of clusters beforehand
- More robust to outliers

DBSCAN Weaknesses:

- Lower performance on quantitative metrics for this dataset
- More sensitive to parameter selection (eps and MinPts)
- Can struggle with clusters of varying densities

6.2 Application Considerations

The choice between K-means and DBSCAN should depend on the specific requirements of the analysis:

- **K-means** is preferable when:
  - The expected clusters are roughly spherical and similar in size
  - The number of clusters can be reasonably estimated
  - Computational efficiency is important
  - All data points should be assigned to a cluster
- **DBSCAN** is preferable when:
  - The data contains significant noise that should be identified
  - Clusters may have irregular shapes
  - The number of natural clusters is unknown
  - Clusters may have different densities

7. Conclusion

For the California Housing dataset analyzed in this study, **K-means outperformed DBSCAN based on quantitative metrics**. The K-means algorithm produced more well-defined, cohesive clusters as evidenced by better Silhouette, Davies-Bouldin, and Calinski-Harabasz scores.

However, DBSCAN provided valuable additional insights by identifying outliers that don't conform to the main demographic patterns in California housing. These outliers might represent unusual housing situations that merit further investigation.

The superior performance of K-means suggests that the natural clusters in this population dataset tend to be relatively well-separated and approximately spherical in the feature space, aligning well with K-means' assumptions.

8. Recommendations

Based on the analysis, the following recommendations are made:

1. For general demographic segmentation of the California Housing dataset, K-means is recommended due to its superior performance on quantitative metrics.
2. When outlier detection is important, DBSCAN should be used alongside K-means to identify unusual patterns in the data.
3. For future work, consider:
    - Exploring hierarchical clustering methods for nested demographic patterns
    - Implementing ensemble clustering approaches that combine the strengths of multiple algorithms
    - Adding more demographic features to enhance the clustering analysis
    - Validating clusters through additional external validation methods

This comparative analysis demonstrates the importance of selecting appropriate clustering algorithms based on data characteristics and analysis objectives, rather than relying on a one-size-fits-all approach to unsupervised learning.