Laboratory assignment

Component 3

Authors: Ichim Stefan, Mirt Leonard
Group: 246/1

November 24, 2024

1 Unsupervised Learning Task

1.1 Clustering and DBSCAN Analysis

Clustering is a machine learning technique that groups similar data points together based on their characteristics or features in multidimensional space. DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a density-based clustering algorithm that identifies clusters based on the concentration of points in space.

The algorithm operates by examining the density relationships between points using two key parameters: epsilon (ϵ), which defines a neighborhood radius, and minPoints, which specifies the minimum number of points required to form a dense region. Points are classified as core points, border points, or noise based on these density criteria. Core points have at least minPoints within their ϵ-neighborhood, border points lie within the ϵ-neighborhood of a core point but have fewer neighbors, and noise points meet neither condition. Clusters are formed by connecting density-reachable core points and their associated border points.

1.1.1 Strengths

- **Shape Flexibility:** Finds clusters of any shape, unlike K-means' circular assumptions
- **Noise Handling:** Naturally identifies outliers
- **No Preset Clusters:** Discovers number of clusters automatically
- **Density-Based:** Works well with varying cluster sizes

1.1.2 Limitations

- **Parameter Sensitivity:** Requires careful tuning
- **Varying Densities:** Struggles with different density clusters
- **High Dimensions:** Performance degrades in high-dimensional spaces

1.2 Learning Framework for DBSCAN Clustering

1.2.1 Target Function

The target function $f : X \to Y$ maps each point $x_i \in \mathbb{R}^d$ to its true cluster label $y_i \in \{1, \ldots, k\} \cup \{-1\}$, where -1 represents noise points. Formally:

$$f(x) = \begin{cases} c_i \text{ if } x \text{ belongs to cluster } i \\ -1 \text{ if } x \text{ is noise} \end{cases}$$

1.2.2 Learning Hypothesis

DBSCAN approximates the target function with hypothesis $h_{\epsilon,m} : X \to Y$ parameterized by:

- ϵ: neighborhood radius
- m: minimum points threshold

The hypothesis function assigns cluster labels based on density-reachability criteria:

$$h_{\epsilon,m}(x) = \begin{cases} c_i \text{ if } x \text{ is density-connected to cluster } i \\ -1 \text{ if } x \text{ is not density-reachable} \end{cases}$$

1.2.3 Representation

The learned function is represented implicitly through:

- Core points: $\{x : |N_\epsilon(x)| \geq m\}$
- Border points: $\{x : |N_\epsilon(x)| < m \text{ but connected to core point}\}$
- Noise points: $\{x : |N_\epsilon(x)| < m \text{ and not connected}\}$

where $N_\epsilon(x)$ is the ϵ-neighborhood of point x.

1.2.4 Learning Algorithm

The DBSCAN learning process is deterministic and occurs through density-based region exploration:

1. **Initialization Phase:**
   - Mark all points as unvisited
   - Initialize empty cluster list C and noise list N
2. **Core Point Identification:**
   - For each unvisited point p:
     - Compute $N_\epsilon(p) = \{q : dist(p, q) \leq \epsilon\}$
     - If $|N_\epsilon(p)| \geq m$, mark p as core point
3. **Cluster Expansion:**
   - For each core point p not yet assigned to cluster:
     - Create new cluster C_k
     - Add p to C_k
     - For each point $q \in N_\epsilon(p)$:
       * If q unvisited: mark as visited, add to C_k
       * If q is core point: add $N_\epsilon(q)$ to processing queue
4. **Noise Identification:**
   - Any remaining unassigned points are marked as noise

Learning Characteristics:

- **Non-parametric Learning:** Does not assume underlying distribution of clusters

- **Instance-based Learning:** Clusters are formed based on local relationships between points
- **Single-pass Algorithm:** Each point is processed exactly once for core point determination
- **Time Complexity:** $O(n \log n)$ with spatial indexing, $O(n^2)$ without

The algorithm "learns" by discovering the inherent density structure of the data space. Unlike supervised learning algorithms, there is no explicit optimization of a loss function. Instead, learning occurs through the progressive discovery and expansion of dense regions, with the final cluster assignments emerging from the density-connectivity relationships between points. This makes DBSCAN particularly effective for datasets where clusters are defined by density rather than geometric distance to cluster centers.