Cluster Analysis: Basic Concepts and Methods

Huan Sun, CSE@The Ohio State University

09/28/2017
Chapter 10. Cluster Analysis: Basic Concepts and Methods

- Cluster Analysis: An Introduction
- Partitioning Methods
- Hierarchical Methods
- Density- and Grid-Based Methods
- Evaluation of Clustering
- Summary
Clustering Algorithms

- K-means and its variants
- Hierarchical clustering
- Density-based clustering
Hierarchical Clustering

- Produces a set of nested clusters organized as a hierarchical tree
- Can be visualized as a dendrogram
  - A tree-like diagram that records the sequences of merges or splits
Hierarchical Clustering

- Two main types of hierarchical clustering
  - **Agglomerative:**
    - Start with the points as individual clusters
    - At each step, merge the closest pair of clusters until only one cluster (or k clusters) left
    - Build a bottom-up hierarchy of clusters
  - **Divisive:**
Agglomerative Clustering Algorithm

- More popular hierarchical clustering technique

- Basic algorithm is straightforward
  1. Compute the proximity matrix
  2. Let each data point be a cluster
  3. **Repeat**
     4. Merge the two closest clusters
     5. Update the proximity matrix
     6. **Until** only a single cluster remains

- **Key operation** is the computation of the proximity of two clusters
  - Different approaches to defining the distance/similarity between clusters distinguish the different algorithms
After some merging steps, we have some clusters

<table>
<thead>
<tr>
<th>C1</th>
<th>C2</th>
<th>C3</th>
<th>C4</th>
<th>C5</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Proximity Matrix

```
p1      p2
p3      p4
```

```p9  p10  p11  p12```
We want to merge the two closest clusters (C2 and C5) and update the proximity matrix.

<table>
<thead>
<tr>
<th></th>
<th>C1</th>
<th>C2</th>
<th>C3</th>
<th>C4</th>
<th>C5</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Intermediate Situation
How do we update the proximity matrix?

Proximity Matrix

<table>
<thead>
<tr>
<th></th>
<th>C1</th>
<th>C5</th>
<th>C3</th>
<th>C4</th>
</tr>
</thead>
<tbody>
<tr>
<td>C1</td>
<td>?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C2 U C5</td>
<td>?</td>
<td>?</td>
<td>?</td>
<td>?</td>
</tr>
<tr>
<td>C3</td>
<td>?</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C4</td>
<td>?</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

After Merging

C1, C3, C4
How to Define Inter-Cluster Similarity

- MIN
- MAX
- Group Average
- Distance Between Centroids

Proximity Matrix

<table>
<thead>
<tr>
<th></th>
<th>p1</th>
<th>p2</th>
<th>p3</th>
<th>p4</th>
<th>p5</th>
<th>. .</th>
</tr>
</thead>
<tbody>
<tr>
<td>p1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>p2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>p3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>p4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>p5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Cluster Similarity: MIN or Single Link

- Similarity of two clusters is based on the two most similar (closest) points in the different clusters.
  - Determined by one pair of points, i.e., by one link in the proximity graph.
Cluster Similarity: MIN or Single Link

- Similarity of two clusters is based on the two most similar (closest) points in the different clusters
  - Determined by one pair of points, i.e., by one link in the proximity graph.

Let us define the distance between two points using Euclidean distance:

$$\delta(x, y) = \|x - y\|_2 = \left(\sum_{i=1}^{d} (x_i - y_i)^2\right)^{1/2}$$

Using single link, the distance between two clusters $C_i$ and $C_j$ is then:

$$\delta(C_i, C_j) = \min\{\delta(x, y) \mid x \in C_i, y \in C_j\}$$

The name comes from the observation that if we choose a line with the minimum distance to connect two points in two clusters, typically only a single link would exist.
Cluster Similarity: MIN or Single Link

- Similarity of two clusters is based on the **two most similar (closest)** points in the different clusters
  - Determined by one pair of points, i.e., by **one link in the proximity graph**.

What if we define the similarity (not distance) between two points?

Using single link, the similarity between two clusters $C_i$ and $C_j$ is then:

$$\text{Sim}(C_i, C_j) = \max\{\text{sim}(x, y) \mid x \in C_i, y \in C_j\}$$

The name comes from the observation that if we choose a line with the minimum distance to connect two points in two clusters, typically only a single link would exist.
Cluster Similarity: MIN or Single Link

- Similarity of two clusters is based on the **two most similar (closest)** points in the different clusters
  - Determined by one pair of points, i.e., by one link in the proximity graph.

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.00</td>
<td>0.90</td>
<td>0.10</td>
<td>0.65</td>
<td>0.20</td>
</tr>
<tr>
<td>2</td>
<td>0.90</td>
<td>1.00</td>
<td>0.70</td>
<td>0.60</td>
<td>0.50</td>
</tr>
<tr>
<td>3</td>
<td>0.10</td>
<td>0.70</td>
<td>1.00</td>
<td>0.40</td>
<td>0.30</td>
</tr>
<tr>
<td>4</td>
<td>0.65</td>
<td>0.60</td>
<td>0.40</td>
<td>1.00</td>
<td>0.80</td>
</tr>
<tr>
<td>5</td>
<td>0.20</td>
<td>0.50</td>
<td>0.30</td>
<td>0.80</td>
<td>1.00</td>
</tr>
</tbody>
</table>

Similarity matrix (e.g., by cosine similarity)
Cluster Similarity: MIN or Single Link

- Similarity of two clusters is based on the **two most similar (closest)** points in the different clusters.
  - Determined by one pair of points, i.e., by **one link in the proximity graph**.

<table>
<thead>
<tr>
<th></th>
<th>I1</th>
<th>I2</th>
<th>I3</th>
<th>I4</th>
<th>I5</th>
</tr>
</thead>
<tbody>
<tr>
<td>I1</td>
<td>0.00</td>
<td><strong>0.10</strong></td>
<td>0.90</td>
<td>0.35</td>
<td>0.80</td>
</tr>
<tr>
<td>I2</td>
<td>0.10</td>
<td>0.00</td>
<td>0.30</td>
<td>0.40</td>
<td>0.50</td>
</tr>
<tr>
<td>I3</td>
<td>0.90</td>
<td>0.30</td>
<td>0.00</td>
<td>0.30</td>
<td>0.70</td>
</tr>
<tr>
<td>I4</td>
<td>0.35</td>
<td>0.40</td>
<td>0.60</td>
<td>0.00</td>
<td>0.20</td>
</tr>
<tr>
<td>I5</td>
<td>0.80</td>
<td>0.50</td>
<td>0.70</td>
<td>0.20</td>
<td>0.00</td>
</tr>
</tbody>
</table>

Note: if using distance measure, the smaller, the closer.

Distance matrix (e.g., by Euclidean distance, Manhattan distance)

See problem III in the sample midterm!
Cluster Similarity: MIN or Single Link

- Similarity of two clusters is based on the **two most similar (closest)** points in the different clusters
  - Determined by one pair of points, i.e., by **one link in the proximity graph**.

<table>
<thead>
<tr>
<th>{I1, I2}</th>
<th>I3</th>
<th>I4</th>
<th>I5</th>
</tr>
</thead>
<tbody>
<tr>
<td>{I1, I2}</td>
<td>1.00</td>
<td>0.70</td>
<td>0.65</td>
</tr>
<tr>
<td>I3</td>
<td>0.70</td>
<td>1.00</td>
<td>0.40</td>
</tr>
<tr>
<td>I4</td>
<td>0.65</td>
<td>0.40</td>
<td>1.00</td>
</tr>
<tr>
<td>I5</td>
<td>0.50</td>
<td>0.30</td>
<td>0.80</td>
</tr>
</tbody>
</table>

Update **similarity matrix** with new cluster \( \{I_1, I_2\} \)
Cluster Similarity: MIN or Single Link

- Similarity of two clusters is based on the **two most similar (closest)** points in the different clusters.
  - Determined by one pair of points, i.e., by **one link in the proximity graph**.

<table>
<thead>
<tr>
<th></th>
<th>I1</th>
<th>I2</th>
<th>I3</th>
<th>I4</th>
<th>I5</th>
</tr>
</thead>
<tbody>
<tr>
<td>{I1, I2}</td>
<td>1.00</td>
<td>0.70</td>
<td>0.65</td>
<td>0.50</td>
<td></td>
</tr>
<tr>
<td>I3</td>
<td>0.70</td>
<td>1.00</td>
<td>0.40</td>
<td>0.30</td>
<td></td>
</tr>
<tr>
<td>I4</td>
<td>0.65</td>
<td>0.40</td>
<td>1.00</td>
<td>0.80</td>
<td></td>
</tr>
<tr>
<td>I5</td>
<td>0.50</td>
<td>0.30</td>
<td>0.80</td>
<td>1.00</td>
<td></td>
</tr>
</tbody>
</table>

Update similarity matrix with new cluster \{I1, I2\}.
Cluster Similarity: MIN or Single Link

- Similarity of two clusters is based on the two most similar (closest) points in the different clusters.
  - Determined by one pair of points, i.e., by one link in the proximity graph.

<table>
<thead>
<tr>
<th></th>
<th>{I1,I2}</th>
<th>I3</th>
<th>{I4,I5}</th>
</tr>
</thead>
<tbody>
<tr>
<td>{I1,I2}</td>
<td>1.00</td>
<td>0.70</td>
<td>0.65</td>
</tr>
<tr>
<td>I3</td>
<td>0.70</td>
<td>1.00</td>
<td>0.40</td>
</tr>
<tr>
<td>{I4,I5}</td>
<td>0.65</td>
<td>0.40</td>
<td>1.00</td>
</tr>
</tbody>
</table>

Update similarity matrix with new cluster \{I1, I2\} and \{I4, I5\}
Cluster Similarity: MIN or Single Link

- Similarity of two clusters is based on the two most similar (closest) points in the different clusters
  - Determined by one pair of points, i.e., by one link in the proximity graph.

{I1, I2, I3} 1.00 0.65
{I4, I5} 0.65 1.00

Only two clusters are left.

Diagram showing the clusters and their connections.
Hierarchical Clustering: MIN

Distance/dissimilarity between clusters

We can see a sequence of nested clusters
Strength of MIN

• Can handle non-elliptical shapes
Limitations of MIN

- Sensitive to noise and outliers
Another example: Single Link

Distance matrix

<table>
<thead>
<tr>
<th>δ</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>B</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>1</td>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Initialization
Another example: Single Link

Distance matrix

\[ \begin{array}{c|ccccc} \delta & B & C & D & E \\ \hline A & 1 & 3 & 2 & 4 \\ B & 3 & 2 & 3 \\ C & & 1 & 3 \\ D & & & 5 \end{array} \]

First Cluster

\[ \begin{array}{c}
A \\
B \\
C \\
D \\
E
\end{array} \]
Another example: Single Link

<table>
<thead>
<tr>
<th>$\delta$</th>
<th>C</th>
<th>D</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>$AB$</td>
<td>3</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>C</td>
<td></td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>D</td>
<td></td>
<td></td>
<td>5</td>
</tr>
</tbody>
</table>

Update distance matrix

Second Cluster
Another example: Single Link

Update distance matrix

<table>
<thead>
<tr>
<th></th>
<th>CD</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>AB</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>CD</td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>
Another example: Single Link

Update distance matrix
Cluster Similarity: MAX or Complete Linkage

- Similarity of two clusters is based on the two least similar (most distant) points in the different clusters
  - Determined by all pairs of points in the two clusters

Let us define the distance between two points using Euclidean distance:

\[ \delta(x, y) = \|x - y\|_2 = \left( \sum_{i=1}^{d} (x_i - y_i)^2 \right)^{1/2} \]

Using MAX link, the distance between two clusters \( C_i \) and \( C_j \) is then:

\[ \delta(C_i, C_j) = \max\{\delta(x, y) \mid x \in C_i, y \in C_j\} \]
Cluster Similarity: MAX or Complete Linkage

- Similarity of two clusters is based on the two least similar (most distant) points in the different clusters
  - Determined by all pairs of points in the two clusters

What if we define the similarity (not distance) between two points?

Using MAX link, the similarity between two clusters $C_i$ and $C_j$ is then:

$$\text{Sim}(C_i, C_j) = \min\{\text{sim}(x, y) \mid x \text{ in } C_i, y \text{ in } C_j\}$$
Cluster Similarity: MAX or Complete Linkage

- Similarity of two clusters is based on **the two least similar (most distant)** points in the different clusters
  - Determined by all pairs of points in the two clusters

<table>
<thead>
<tr>
<th></th>
<th>I1</th>
<th>I2</th>
<th>I3</th>
<th>I4</th>
<th>I5</th>
</tr>
</thead>
<tbody>
<tr>
<td>I1</td>
<td>1.00</td>
<td>0.90</td>
<td>0.10</td>
<td>0.65</td>
<td>0.20</td>
</tr>
<tr>
<td>I2</td>
<td>0.90</td>
<td>1.00</td>
<td>0.70</td>
<td>0.60</td>
<td>0.50</td>
</tr>
<tr>
<td>I3</td>
<td>0.10</td>
<td>0.70</td>
<td>1.00</td>
<td>0.40</td>
<td>0.30</td>
</tr>
<tr>
<td>I4</td>
<td>0.65</td>
<td>0.60</td>
<td>0.40</td>
<td>1.00</td>
<td>0.80</td>
</tr>
<tr>
<td>I5</td>
<td>0.20</td>
<td>0.50</td>
<td>0.30</td>
<td>0.80</td>
<td>1.00</td>
</tr>
</tbody>
</table>

Similarity matrix (e.g., by cosine similarity)
Cluster Similarity: MAX or Complete Linkage

- Similarity of two clusters is based on **the two least similar (most distant) points in the different clusters**
  - Determined by all pairs of points in the two clusters

<table>
<thead>
<tr>
<th></th>
<th>I3</th>
<th>I4</th>
<th>I5</th>
</tr>
</thead>
<tbody>
<tr>
<td>I1, I2</td>
<td>1.00 0.10</td>
<td>0.60 0.20</td>
<td></td>
</tr>
<tr>
<td>I3</td>
<td>0.10 1.00</td>
<td>0.40 0.30</td>
<td></td>
</tr>
<tr>
<td>I4</td>
<td>0.60 0.40</td>
<td>1.00 0.80</td>
<td></td>
</tr>
<tr>
<td>I5</td>
<td>0.20 0.30</td>
<td>0.80 1.00</td>
<td></td>
</tr>
</tbody>
</table>

How do we get this number? How to update the similarity matrix?

Update similarity matrix with new cluster \{I1, I2\}
Cluster Similarity: MAX or Complete Linkage

- Similarity of two clusters is based on the two least similar (most distant) points in the different clusters
  - Determined by all pairs of points in the two clusters

At each step, you will always choose the most similar clusters to merge, regardless of the measure you choose.
Cluster Similarity: MAX or Complete Linkage

- Similarity of two clusters is based on the two least similar (most distant) points in the different clusters
  - Determined by all pairs of points in the two clusters

<table>
<thead>
<tr>
<th></th>
<th>I3</th>
<th>I4</th>
<th>I5</th>
</tr>
</thead>
<tbody>
<tr>
<td>{I1,I2}</td>
<td>1.00</td>
<td>0.10</td>
<td>0.60</td>
</tr>
<tr>
<td>I3</td>
<td>0.10</td>
<td>1.00</td>
<td>0.40</td>
</tr>
<tr>
<td>I4</td>
<td>0.60</td>
<td>0.40</td>
<td>1.00</td>
</tr>
<tr>
<td>I5</td>
<td>0.20</td>
<td>0.30</td>
<td>0.80</td>
</tr>
</tbody>
</table>

After this, which two clusters should be merged next?
Cluster Similarity: MAX or Complete Linkage

- Similarity of two clusters is based on the two least similar (most distant) points in the different clusters
  - Determined by all pairs of points in the two clusters

\[
\begin{array}{c|cccc}
\{1,2\} & 1 & 0.10 & 0.60 & 0.20 \\
\{1,2\} & 0.10 & 1 & 0.40 & 0.30 \\
3 & 0.60 & 0.40 & 1 & 0.80 \\
4 & 0.20 & 0.30 & 0.80 & 1
\end{array}
\]

Merge \{3\} with \{4,5\}, why?
Hierarchical Clustering: MAX

A sequence of nested clusters

Dendrogram

NestedClusters
Strength of MAX

- Less susceptible to noise and outliers

Original Points

Two Clusters
Limitations of MAX

- Tends to break large clusters
- Biased towards globular clusters
Cluster Similarity: Group Average

- Proximity of two clusters is the average of pairwise proximity between points in the two clusters.

\[
\text{proximity}(\text{Cluster}_i, \text{Cluster}_j) = \frac{\sum_{p_i \in \text{Cluster}_i, p_j \in \text{Cluster}_j} \text{proximity}(p_i, p_j)}{|\text{Cluster}_i| \times |\text{Cluster}_j|}
\]

- Need to use average connectivity for scalability since total proximity favors large clusters

<table>
<thead>
<tr>
<th></th>
<th>I1</th>
<th>I2</th>
<th>I3</th>
<th>I4</th>
<th>I5</th>
</tr>
</thead>
<tbody>
<tr>
<td>I1</td>
<td>1.00</td>
<td>0.90</td>
<td>0.10</td>
<td>0.65</td>
<td>0.20</td>
</tr>
<tr>
<td>I2</td>
<td>0.90</td>
<td>1.00</td>
<td>0.70</td>
<td>0.60</td>
<td>0.50</td>
</tr>
<tr>
<td>I3</td>
<td>0.10</td>
<td>0.70</td>
<td>1.00</td>
<td>0.40</td>
<td>0.30</td>
</tr>
<tr>
<td>I4</td>
<td>0.65</td>
<td>0.60</td>
<td>0.40</td>
<td>1.00</td>
<td>0.80</td>
</tr>
<tr>
<td>I5</td>
<td>0.20</td>
<td>0.50</td>
<td>0.30</td>
<td>0.80</td>
<td>1.00</td>
</tr>
</tbody>
</table>
Hierarchical Clustering: Group Average

A sequence of nested clusters

Dendrogram

Nested Clusters
Hierarchical Clustering: Group Average

- Compromise between Single and Complete Link

**Strengths**
- Less susceptible to noise and outliers

**Limitations**
- Biased towards globular clusters
Hierarchical Clustering: Time and Space requirements

- $O(N^2)$ space since it uses the proximity matrix.
  - $N$ is the number of points.

- $O(N^3)$ time in many cases
  - There are $N$ steps and at each step the size, $N^2$, proximity matrix must be updated and searched
  - Complexity can be reduced to $O(N^2 \log(N))$ time for some approaches
Hierarchical Clustering: Problems and Limitations

- Once a decision is made to combine two clusters, it cannot be undone

- No objective function is directly minimized

- Different schemes have problems with one or more of the following:
  - Sensitivity to noise and outliers
  - Difficulty handling different sized clusters and convex shapes
  - Breaking large clusters
Divisive Clustering

- DIANA (Divisive Analysis) (Kaufmann and Rousseeuw, 1990)
  - Implemented in some statistical analysis packages, e.g., Splus
- Inverse order of AGNES: Eventually each node forms a cluster on its own

- Divisive clustering is a top-down approach
  - The process starts at the root with all the points as one cluster
  - It recursively splits the higher level clusters to build the dendrogram
  - Can be considered as a global approach
  - More efficient when compared with agglomerative clustering
More on Algorithm Design for Divisive Clustering

- Choosing which cluster to split
  - Check the sums of squared errors of the clusters and choose the one with the largest value

- Splitting criterion: Determining how to split
  - One may use Ward’s criterion to chase for greater reduction in the difference in the SSE criterion as a result of a split
  - For categorical data, Gini-index can be used

- Handling the noise
  - Use a threshold to determine the termination criterion (do not generate clusters that are too small because they contain mainly noises)
Clustering Algorithms

- K-means and its variants
- Hierarchical clustering
- Density-based clustering
Density-Based Clustering Methods

- Clustering based on density (a local cluster criterion), such as density-connected points

- Major features:
  - Discover clusters of arbitrary shape
  - Handle noise
  - One scan (only examine the local region to justify density)
  - Need density parameters as termination condition

- Several interesting studies:
  - **DBSCAN**: Ester, et al. (KDD’96)
  - **OPTICS**: Ankerst, et al (SIGMOD’99)
  - **DENCLUE**: Hinneburg & D. Keim (KDD’98)
  - **CLIQUE**: Agrawal, et al. (SIGMOD’98) (also, grid-based)
DBSCAN

- DBSCAN (M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, KDD’96)
  - Discovers clusters of arbitrary shape: Density-Based Spatial Clustering of Applications with Noise
DBSCAN

- DBSCAN (M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, KDD’96)
  - Discovers clusters of arbitrary shape: Density-Based Spatial Clustering of Applications with Noise

- DBSCAN is a density-based algorithm.
  - Density = number of points within a specified radius (Eps)
  - A point is a **core point** if it has more than a specified number of points (MinPts) within Eps
    - These are points that are at the interior of a cluster
  - A **border point** has fewer than MinPts within Eps, but is in the neighborhood of a core point
  - A **noise point** is any point that is not a core point or a border point.
DBSCAN

- DBSCAN is a density-based algorithm.
  - Density = number of points within a specified radius (Eps)
  
- A point is a core point if it has more than a specified number of points (MinPts) within Eps
  - These are points that are at the interior of a cluster
  
- A border point has fewer than MinPts within Eps, but is in the neighborhood of a core point
  
- A noise point is any point that is not a core point or a border point.
DBSCAN: Core, Border, and Noise Points

1. A point is a core point if it has more than a specified number of points (MinPts) within Eps.

2. A border point has fewer than MinPts within Eps, but is in the neighborhood of a core point.

3. A noise point is any point that is not a core point or a border point.
DBSCAN Algorithm

- Eliminate noise points
- Perform clustering on the remaining points

\[ current\_cluster\_label \leftarrow 1 \]

\[ \text{for all core points do} \]

\[ \text{if the core point has no cluster label then} \]

\[ current\_cluster\_label \leftarrow current\_cluster\_label + 1 \]

Label the current core point with cluster label \( current\_cluster\_label \)

\[ \text{end if} \]

\[ \text{for all points in the } Eps\text{-neighborhood, except } i^{th} \text{ the point itself do} \]

\[ \text{if the point does not have a cluster label then} \]

Label the point with cluster label \( current\_cluster\_label \)

\[ \text{end if} \]

\[ \text{end for} \]

\[ \text{end for} \]

The Eps-neighborhood of a point \( q \): 
\[ N_{Eps}(q): \{ p \text{ belongs to } D \mid dist(p, q) \leq Eps \} \]
DBSCAN: Core, Border and Noise Points

Original Points

Point types: core, border and noise

Eps = 10, MinPts = 4
When DBSCAN Works Well

- Resistant to Noise
- Can handle clusters of different shapes and sizes
When DBSCAN Does NOT Work Well

- Varying densities
- High-dimensional data

Original Points

(MinPts=4, Eps=9.75).

(MinPts=4, Eps=9.92)
Cluster Validity

- For supervised classification we have a variety of measures to evaluate how good our model is
  - Accuracy, precision, recall

- For cluster analysis, the analogous question is how to evaluate the “goodness” of the resulting clusters?
Cluster Validity

- For supervised classification we have a variety of measures to evaluate how good our model is
  - Accuracy, precision, recall

- For cluster analysis, the analogous question is how to evaluate the “goodness” of the resulting clusters?

- But “clusters are in the eye of the beholder”!

- Then why do we want to evaluate them?
  - To avoid finding patterns in noise
  - To compare clustering algorithms
  - To compare two sets of clusters
  - To compare two clusters
Clusters found in Random Data

Random Points

DBSCAN

K-means

Complete Link
Different Aspects of Cluster Validation

1. Determining the **clustering tendency** of a set of data, i.e., distinguishing whether non-random structure actually exists in the data.

2. Comparing the results of a cluster analysis to externally known results, e.g., to externally given class labels.

3. Evaluating how well the results of a cluster analysis fit the data *without* reference to external information.
   - Use only the data

4. Comparing the results of two different sets of cluster analyses to determine which is better.

5. Determining the ‘correct’ number of clusters.

For 2, 3, and 4, we can further distinguish whether we want to evaluate the entire clustering or just individual clusters.
Order the similarity matrix with respect to cluster labels and inspect visually.
Using Similarity Matrix for Cluster Validation

- Clusters in random data are not so crisp

**DBSCAN**
Intrinsic Measures of Clustering quality

\[
\text{centroid} = C_m = \frac{\sum_{i=1}^{N} (t_{mi})}{N}
\]

\[
\text{radius} = R_m = \sqrt{\frac{\sum_{i=1}^{N} (t_{mi} - C_m)^2}{N}}
\]

\[
\text{diameter} = D_m = \sqrt{\frac{\sum_{i=1}^{N} \sum_{j=1}^{N} (t_{mi} - t_{mj})^2}{(N)(N-1)}}
\]
A proximity graph based approach can also be used for cohesion and separation.

- Cluster cohesion is the sum of the weight of all links within a cluster.
- Cluster separation is the sum of the weights between nodes in the cluster and nodes outside the cluster.
Silhouette Coefficient

- Silhouette Coefficient combine ideas of both cohesion and separation, but for individual points, as well as clusters and clusterings.
- For an individual point, \( i \):
  - Calculate \( a \) = average distance of \( i \) to the points in its cluster.
  - Calculate \( b = \min \) (average distance of \( i \) to points in another cluster).
  - The silhouette coefficient for a point is then given by
    
    \[
    s = 1 - \frac{a}{b} \quad \text{if} \quad a < b, \quad \text{(or} \quad s = \frac{b}{a} - 1 \quad \text{if} \quad a \geq b, \quad \text{not the usual case)}
    \]
  - Typically between 0 and 1.
  - The closer to 1 the better.

- Can calculate the Average Silhouette width for a cluster or a clustering.
Other Measures of Cluster Validity

- **Entropy/Gini**
  - If there is a class label — you can use the entropy/gini of the class label — similar to what we did for classification *(Check problem III in sample midterm)*

  - If there is no class label — one can compute the entropy w.r.t each attribute (dimension) and sum up or weighted average to compute the disorder within a cluster

- **Classification Error**
  - If there is a class label one can compute this in a similar manner
Extensions: Clustering Large Databases

- Most clustering algorithms assume a large data structure which is memory resident.

- Clustering may be performed first on a sample of the database then applied to the entire database.

- Algorithms
  - BIRCH
  - DBSCAN (we have already covered this)
  - CURE
Backup slides
**MST: Divisive Hierarchical Clustering**

- **Build MST (Minimum Spanning Tree)**
  - Start with a tree that consists of any point
  - In successive steps, look for the closest pair of points \((p, q)\) such that one point \((p)\) is in the current tree but the other \((q)\) is not
  - Add \(q\) to the tree and put an edge between \(p\) and \(q\)
MST: Divisive Hierarchical Clustering

- Use MST for constructing hierarchy of clusters

**Algorithm 7.5** MST Divisive Hierarchical Clustering Algorithm

1. Compute a minimum spanning tree for the proximity graph.
2. repeat
3. Create a new cluster by breaking the link corresponding to the largest distance (smallest similarity).
4. until Only singleton clusters remain