Clustering

-Slides compiled by Sanghamitra De

Basics

- **Clustering** methods simply try to group similar **patterns** into clusters whose members are more similar to each other (according to some distance measure) than to members of other clusters.
- There is no a priori knowledge of **patterns** that belong to certain groups, or even how many groups are appropriate.
- **Pattern recognition** is the process of recognizing patterns by using machine learning algorithm. Classification is used in supervised learning.

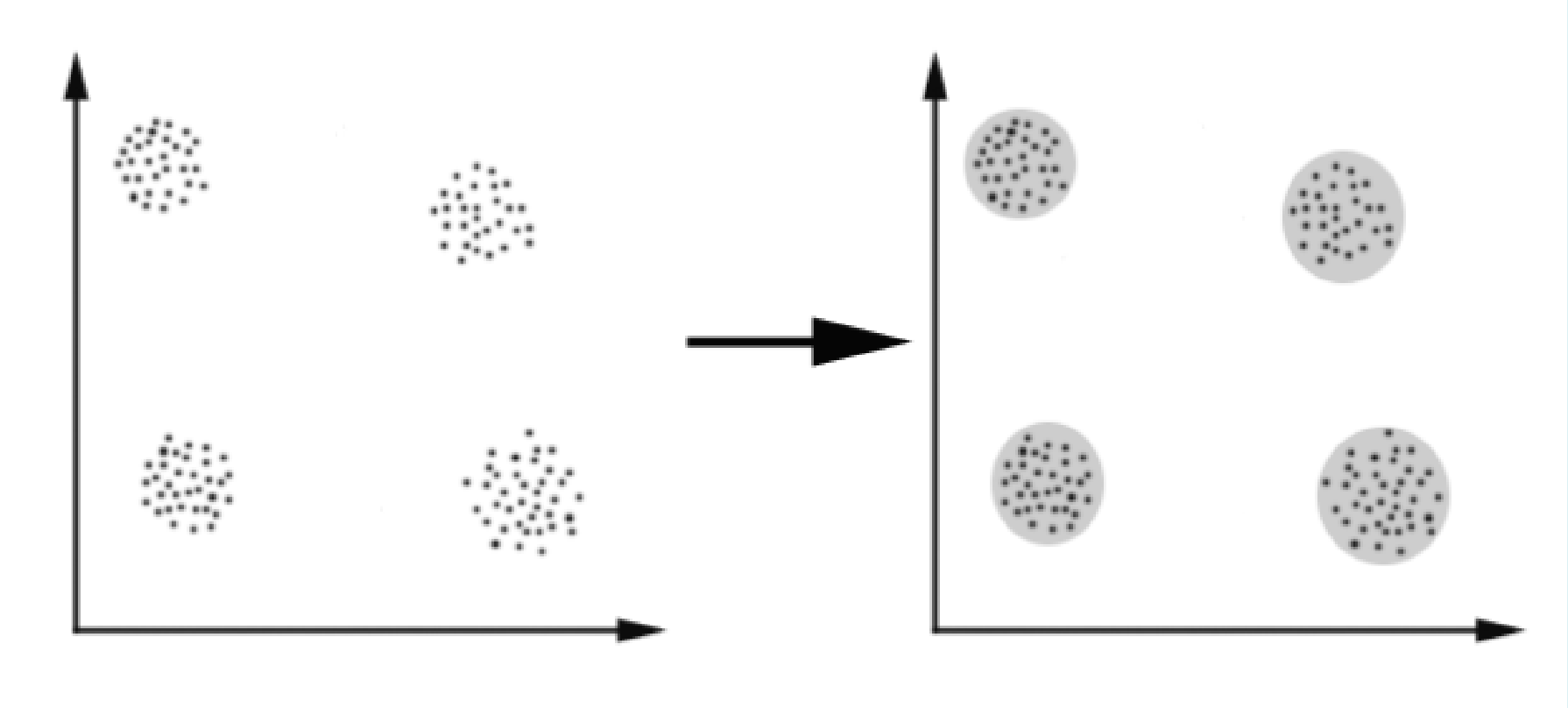
Basics

- While problems in Pattern Recognition and Machine Learning can be of various types, they can be broadly classified into three categories:
  - ***Supervised Learning:*** *The system is presented with example inputs and their desired outputs, given by a "teacher", and the goal is to learn a general rule that maps inputs to outputs.*
  - ***Unsupervised Learning:*** *No labels are given to the learning algorithm, leaving it on its own to find structure in its input. Unsupervised learning can be a goal in itself (discovering hidden patterns in data) or a means towards an end (feature learning).*
  - ***Reinforcement Learning:*** *A system interacts with a dynamic environment in which it must perform a certain goal (such as driving a vehicle or playing a game against an opponent). The system is provided feedback in terms of rewards and punishments as it navigates its problem space.*

Clustering

- **Clustering** can be considered the most important *unsupervised learning* problem; so, as every other problem of this kind, it deals with finding a structure in a collection of unlabeled data.
- A loose definition of clustering could be “the process of organizing objects into groups whose members are similar in some way”.
- A cluster is therefore a collection of objects which are “similar” between them and are “dissimilar” to the objects belonging to other clusters.

Clustering



Clustering Algorithms

- Clustering algorithms may be classified as listed below:
  - ✓ Exclusive Clustering
  - ✓ Overlapping Clustering
  - ✓ Hierarchical Clustering
  - ✓ Probabilistic Clustering

K-Means Clustering

- K-means is one of the simplest unsupervised learning algorithms that solve the well known clustering problem.
- The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed a priori.
- The main idea is to define k centers, one for each cluster.
- These centers should be placed carefully because different location causes different result.
- So, the better choice is to place them as much as possible far away from each other.
- The next step is to take each point belonging to a given data set and associate it to the nearest center.

K-Means Clustering

- When no point is pending, the first step is completed and an early group stage is done.
- At this point we need to re-calculate k new centroids as barycenter of the clusters resulting from the previous step.
- After we have these k new centroids, a new binding has to be done between the same data set points and the nearest new center.
- A loop has been generated.
- As a result of this loop we may notice that the k centers change their location step by step until no more changes are done or in other words centers do not move any more.
- Finally, this algorithm aims at minimizing an objective function know as ***squared error function*** given by:

$$J(V) = \sum_{i=1}^{c} \sum_{j=1}^{c_i} \left(\| x_i - v_j \| \right)^2$$

where,

'$||x_i - v_j||$' is the Euclidean distance between x_i and v_j.

'c_i' is the number of data points in i^{th} cluster.

'c' is the number of cluster centers.

Algorithm 1 (K-means clustering)

1 begin initialize $n, c, \mu_1, \mu_2, \ldots, \mu_c$
2 do classify n samples according to nearest μ_i
3 recompute μ_i
4 until no change in μ_i
5 return $\mu_1, \mu_2, \ldots, \mu_c$
6 end

- Here ***c*** represents the number of clusters, ***n*** is the number of samples and ***μ*** stands for mean
- The computational complexity of the algorithm is O(*ndc*T) where d is the number of features and T is the number of iterations.
- In practice, the number of iterations is generally much less than the number of samples.

Data description & Clustering

- Let us reconsider the original problem of learning something of use from a set of unlabeled samples.
- Viewed geometrically, these samples may form clouds of points in a d-dimensional space.
- Suppose these points came from a single normal distribution.
- Then the sample mean and the sample covariance matrix would give a compact description of the data or the samples.
- The sample mean locates the center of gravity of the cloud; it can be thought of as the single point ***m*** that best represents all of the data in the sense of minimizing the sum of squared distances from m to the samples.
- The sample covariance matrix describes the amount the data scatters along various directions.

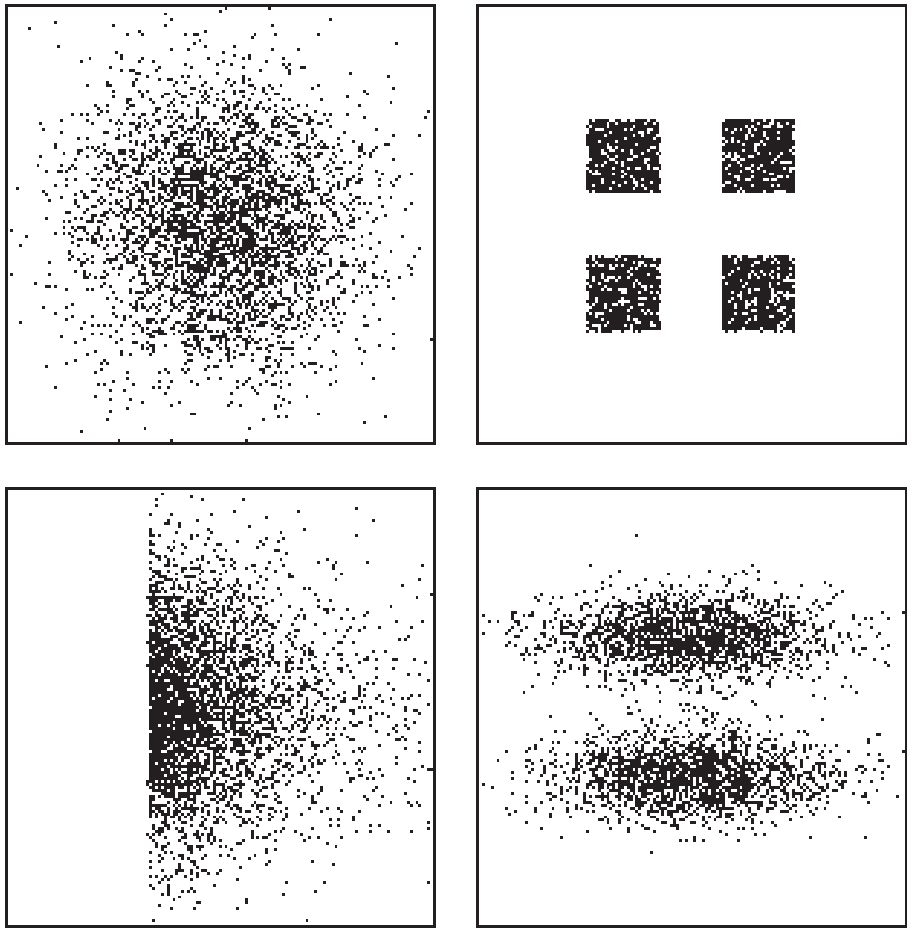


Figure 10.5: These four data sets have identical statistics up to second-order, i.e., the same mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$. In such cases it is important to include in the model more parameters to represent the structure more completely.

Data description & Clustering

- If it is assumed that the samples come from a mixture of c normal distributions, then a greater variety of situations could be approximated.
- This corresponds to assuming that the samples fall in ***hyperellipsoidally*** shaped clouds of various sizes and orientations.
- If the number of component densities is sufficiently high, any density function as a mixture model can be approximated in this way, and, parameters of the mixture can be used to describe the data.
- In situations where we have relatively little prior knowledge about the nature of the data, the assumption of particular parametric forms may lead to poor or meaningless results.

Data description & Clustering

- To find subclasses, a more direct alternative is to use a clustering procedure.
- Clustering procedures yield a data description in terms of clusters or groups of data points that possess strong internal similarities.
- Formal clustering procedures use a criterion function, such as the sum of the squared distances from the cluster centers, and seek the grouping that extremizes the criterion function.

Similarity Measures

- In what sense it can be said that the samples in one cluster are more like one another than like samples in other clusters?

OR

- How should one measure the similarity between samples?

NOTE: *If distance is a good measure of dissimilarity, then one would expect the distance between samples in the same cluster to be significantly less than the distance between samples in different clusters.*

Similarity measures

➢**Euclidean distance:**

Clusters defined by Euclidean distance will be invariant to translations or rotations in feature space — rigid-body motions of the data points. However, they will not be invariant to linear transformations in general, or to other transformations that distort the distance relationships.

Similarity measures

➢**Minkowski distance:**

Minkowski distance is a distance/similarity measurement between two points in the normed vector space (N dimensional real space) and is a generalization of the Euclidean distance and the Manhattan distance.

Similarity measures

➢**Similarity Function:**

One can introduce a nonmetric similarity function s(x, x') to compare two vectors x and x'. Conventionally, this is a symmetric function whose value is large when x and x' are somehow "similar."

For example, when the angle between two vectors is a meaningful measure of their similarity, then the normalized inner product may be an appropriate similarity function:

$$s(\mathbf{x}, \mathbf{x}') = \frac{\mathbf{x}^t \mathbf{x}'}{\|\mathbf{x}\| \, \|\mathbf{x}'\|}$$

This measure (a cosine of angle between x and x') is invariant to rotation and dilation but not invariant to translation and general linear transformation.

Similarity measures

➢**Tanimoto distance:**

When features are binary valued (0 or 1) similarity function has a non-geometrical interpretation in terms of shared features or shared attributes.

So, s(x, x') becomes the ratio of the number of shared attributes to the number possessed by x or x'.

$$s(\mathbf{x}, \mathbf{x}') = \frac{\mathbf{x}^t \mathbf{x}'}{\mathbf{x}^t \mathbf{x} + \mathbf{x}'^t \mathbf{x}' - \mathbf{x}^t \mathbf{x}'},$$

This measure (sometimes known as the Tanimoto coefficient or Tanimoto distance) is frequently encountered in the fields of information retrieval and biological taxonomy.

Criterion Functions for Clustering

- Suppose that we have a set D of n samples $\{x_1, \ldots, x_n\}$ that we want to partition into exactly c disjoint subsets $D_1, \ldots, D_c$.
- Each subset is to represent a cluster, with samples in the same cluster being somehow more similar than samples in different clusters.
- One way to make this into a well-defined problem is to define a ***criterion function*** that measures the clustering quality of any partition of the data.
- Then the problem is one of finding the partition that extremizes the criterion function.

Sum-of-Squared-Error Criterion

- This criterion function used for clustering is the sum-of-squared-error criterion.
- Let n_i be the number of samples in D_i and let m_i be the mean of those samples,

$$\mathbf{m}_i = \frac{1}{n_i} \sum_{\mathbf{x} \in \mathcal{D}_i} \mathbf{x}.$$

- Then the sum-of-squared errors is defined by:

$$J_e = \sum_{i=1}^{c} \sum_{\mathbf{x} \in \mathcal{D}_i} \|\mathbf{x} - \mathbf{m}_i\|^2.$$

Sum-of-Squared-Error Criterion

- So, for a given cluster D_i, the mean vector $\mathbf{m_i}$ is the best representative of the samples in D_i in the sense that it minimizes the sum of the squared lengths of the "error" vectors $\mathbf{x} - \mathbf{m_i}$ in D_i.
- Thus, J_e measures the total squared error incurred in representing the n samples $\mathbf{x_1}, \ldots, \mathbf{x_n}$ by the c cluster centers $\mathbf{m_1}, \ldots, \mathbf{m_c}$.
- The value of J_e depends on how the samples are grouped into clusters and the number of clusters; the optimal partitioning is defined as one that minimizes J_e
- Clusterings of this type are often called ***minimum variance partitions***.

Figure 10.9: When two natural groupings have very different numbers of points, the clusters minimizing a sum-squared-error criterion (Eq. 49) may not reveal the true underlying structure. Here the criterion is smaller for the two clusters at the bottom than at the more natural clustering at the top.

Related Minimum Variance Criteria

- By some simple algebraic manipulation one can eliminate the mean vectors from the expression for J_e and obtain the equivalent expression

$$J_e = \frac{1}{2} \sum_{i=1}^{c} n_i \bar{s}_i ,$$

- where

$$\bar{s}_i = \frac{1}{n^2} \sum_{\mathbf{x} \in \mathcal{D}_i} \sum_{\mathbf{x}' \in \mathcal{D}_i} \|\mathbf{x} - \mathbf{x}'\|^2 .$$

Related Minimum Variance Criteria

- 2nd equation in last slide leads to the interpretation that $\bar{s}_i$ as the average squared distance between points in the i^{th} cluster, and emphasizes the fact that the sum-of-squared-error criterion uses Euclidean distance as the measure of similarity.
- It also suggests an obvious way of obtaining other criterion functions. For example, one can replace $\bar{s}_i$ by the average, the median, or perhaps the maximum distance between points in a cluster.
- More generally, one can introduce an appropriate similarity function s(x, x') and replace $\bar{s}_i$ by functions such as

$$\bar{s}_i = \frac{1}{n_i^2} \sum_{\mathbf{x} \in \mathcal{D}_i} \sum_{\mathbf{x}' \in \mathcal{D}_i} s(\mathbf{x}, \mathbf{x}')$$

or

$$\bar{s}_i = \min_{\mathbf{x}, \mathbf{x}' \in \mathcal{D}_i} s(\mathbf{x}, \mathbf{x}').$$

Scattering Criteria

- Another interesting class of criterion functions can be derived from the scatter matrices used in multiple discriminant analysis.
- The total scatter matrix is the sum of the within-cluster scatter matrix and the between-cluster scatter matrix:

$$\mathbf{S}_T = \mathbf{S}_W + \mathbf{S}_B.$$

- Note that the total scatter matrix does not depend on how the set of samples is partitioned into clusters; it depends only on the total set of samples.
- In talking about the amount of within-cluster or between-cluster scatter, we need a scalar measure of the "size" of a scatter matrix.
- The two measures that we shall consider are the *trace* and the *determinant*.

Table 10.1: Mean vectors and scatter matrices used in clustering criteria.

	Depend on cluster center?		
	Yes	No	
Mean vector for the ith cluster		×	$\mathbf{m}_i = \frac{1}{n_i} \sum_{\mathbf{x} \in \mathcal{D}_i} \mathbf{x}$ (54)
Total mean vector		×	$\mathbf{m} = \frac{1}{n} \sum_{\mathcal{D}} \mathbf{x} = \frac{1}{n} \sum_{i=1}^{c} n_i \mathbf{m}_i$ (55)
Scatter matrix for the ith cluster	×		$\mathbf{S}_i = \sum_{\mathbf{x} \in \mathcal{D}_i} (\mathbf{x} - \mathbf{m}_i)(\mathbf{x} - \mathbf{m}_i)^t$ (56)
Within-cluster scatter matrix	×		$\mathbf{S}_W = \sum_{i=1}^{c} \mathbf{S}_i$ (57)
Between-cluster scatter matrix	×		$\mathbf{S}_B = \sum_{i=1}^{c} n_i (\mathbf{m}_i - \mathbf{m})(\mathbf{m}_i - \mathbf{m})^t$ (58)
Total scatter matrix		×	$\mathbf{S}_T = \sum_{\mathbf{x} \in \mathcal{D}} (\mathbf{x} - \mathbf{m})(\mathbf{x} - \mathbf{m})^t$ (59)

Scattering Criteria: The Trace Criterion

- The simplest scalar measure of a scatter matrix is its trace — the sum of its diagonal elements.
- The trace measures the square of the scattering radius, since it is proportional to the sum of the variances in the coordinate directions.
- Thus, an obvious criterion function to minimize is the trace of S_W.
- In fact, this criterion is nothing more or less than the sum-of-squared-error criterion, since the definitions of scatter matrices (Eqs. 56 & 57) yield

$$tr\ \mathbf{S}_W = \sum_{i=1}^{c} tr\ \mathbf{S}_i = \sum_{i=1}^{c} \sum_{\mathbf{x}\in\mathcal{D}_i} \|\mathbf{x} - \mathbf{m}_i\|^2 = J_e.$$

Scattering Criteria: The Trace Criterion

- Since $tr[\mathbf{S}_T] = tr[\mathbf{S}_W] + tr[\mathbf{S}_B]$ and $tr[\mathbf{S}_T]$ is independent of how the samples are partitioned, we see that no new results are obtained by trying to maximize $tr[\mathbf{S}_B]$. However, in seeking to minimize the within-cluster criterion $J_e = tr[\mathbf{S}_W]$ we are maximizing the between-cluster criterion:

$$trS_B = \sum_{i=1}^{c} n_i \|\mathbf{m}_i - \mathbf{m}\|^2.$$

Scattering Criteria: The Determinant Criterion

- The determinant of the scatter matrix has been used to obtain a scalar measure of scatter.
- The determinant measures the square of the scattering volume, since it is proportional to the product of the variances in the directions of the principal axes.
- Since $\mathbf{S}_B$ will be singular if the number of clusters is less than or equal to the dimensionality, $|\mathbf{S}_B|$ is obviously a poor choice for a criterion function.
- However, if we assume that $\mathbf{S}_W$ is nonsingular, we are led to consider the determinant criterion function

$$J_d = |\mathbf{S}_W| = \left| \sum_{i=1}^{c} \mathbf{S}_i \right|$$

- The partition that minimizes $\boldsymbol{J}_d$ is often similar to the one that minimizes $\boldsymbol{J}_e$, but the two need not be the same.

Scattering Criteria: The Invariant Criterion

- The eigenvalues $\lambda_1, \ldots, \lambda_d$ of $\mathbf{S}^{-1}_{\mathrm{W}}\,\mathbf{S}_{\mathrm{B}}$ are invariant under nonsingular linear transformations of the data.
- These eigenvalues are the basic linear invariants of the scatter matrices.
- Their numerical values measure the ratio of between-cluster to within-cluster scatter in the direction of the eigenvectors, and partitions that yield large values are usually desirable.
- Since the trace of a matrix is the sum of its eigenvalues, one might elect to maximize the criterion function

$$tr\mathbf{S}_W^{-1}\mathbf{S}_B = \sum_{i=1}^{d} \lambda_i.$$

- Since such criterion functions are invariant to linear transformations, the same is true of the partitions that extremize them.

Hierarchical Clustering

- Hierarchical clustering permeates classification activities in the sciences.
- Let us consider a sequence of partitions of the n samples into c clusters.
- The first of these is a partition into n clusters, each cluster containing exactly one sample.
- The next is a partition into n−1 clusters, the next a partition into n−2, and so on until the nth, in which all the samples form one cluster.
- We shall say that we are at level k in the sequence when $c = n - k + 1$.
- So, level one corresponds to n clusters and level n to one cluster.
- Given any two samples x and x^- at some level they will be grouped together in the same cluster.
- If the sequence has the property that whenever two samples are in the same cluster at level k they remain together at all higher levels, then the sequence is said to be a ***hierarchical clustering.***

Hierarchical Clustering

- The most natural representation of hierarchical clustering is a corresponding tree, called a ***dendrogram***, which shows how the samples are grouped.
- Figure next shows a dendrogram for a simple problem involving eight samples.
- Level 1 shows the eight samples as singleton clusters. At level 2, samples $\mathbf{x_6}$ and $\mathbf{x_7}$ have been grouped to form a cluster, and they stay together at all subsequent levels.
- If it is possible to measure the similarity between clusters, then the dendrogram is usually drawn to scale to show the similarity between the clusters that are grouped.
- In the given figure, for example, the similarity between the two groups of samples that are merged at level 5 has a value of roughly 60.

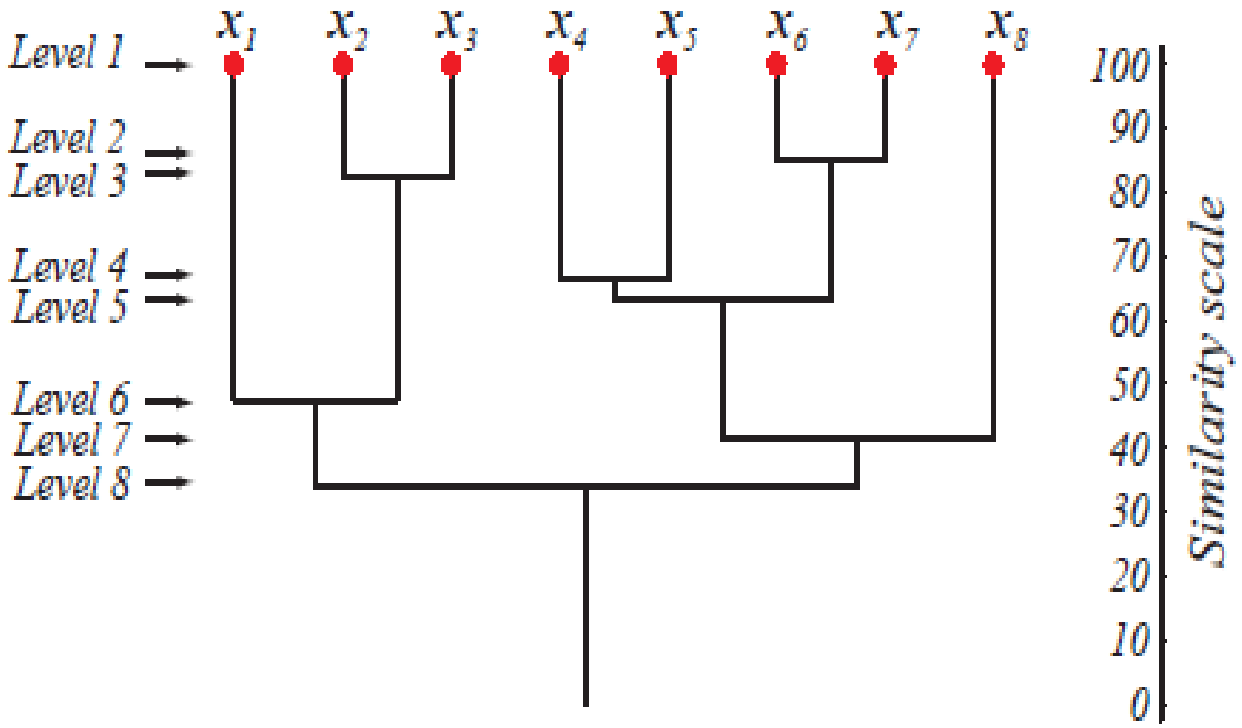


Figure 10.10: A dendrogram can represent the results of hierarchical clustering algorithms. The vertical axis shows a generalized measure of similarity among clusters. Here, at level 1 all eight points lie in singleton clusters; each point in a cluster is highly similar to itself, of course. Points $\mathbf{x}_6$ and $\mathbf{x}_7$ happen to be the most similar, and are merged at level 2, and so forth.

Hierarchical Clustering

- Another representation for hierarchical clustering is based on sets, in which each level of cluster may contain sets that are subclusters, as shown in next figure.
- Yet another, textual, representation uses brackets, such as: {{x1, {x2, x3}}, {{x4, x5}, {x6, x7}}, x8}}.
- While such representations may reveal the hierarchical structure of the data, they do not naturally represent the similarities quantitatively. For this reason dendrograms are generally preferred.

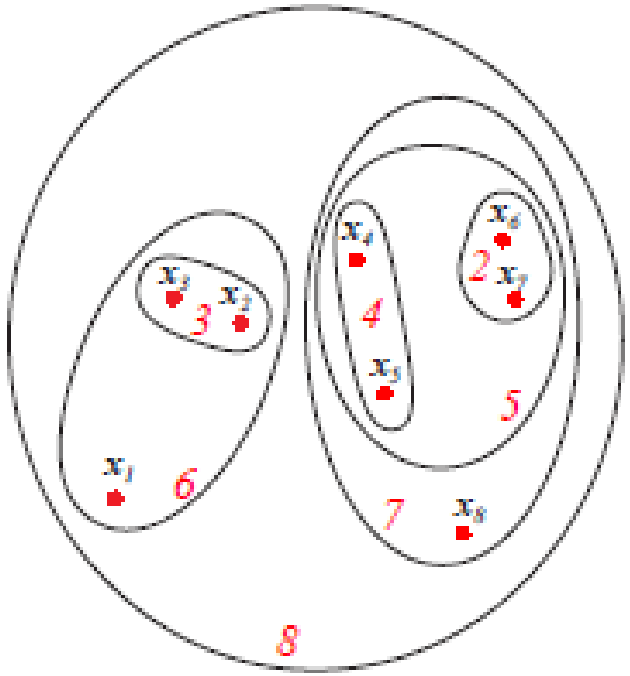


Figure 10.11: A set or Venn diagram representation of two-dimensional data (which was used in the dendrogram of Fig. 10.10) reveals the hierarchical structure but not the quantitative distances between clusters. The levels are numbered in red.

Hierarchical Clustering

- Due to their conceptual simplicity, hierarchical clustering procedures are among the best-known of unsupervised methods.
- The procedures themselves can be divided according to two distinct approaches — agglomerative and divisive.
- Agglomerative (bottom-up, clumping) procedures start with n singleton clusters and form the sequence by successively merging clusters.
- Divisive (top-down, splitting) procedures start with all of the samples in one cluster and form the sequence by successively splitting clusters.
- The computation needed to go from one level to another is usually simpler for the agglomerative procedures.
- However, when there are many samples and one is interested in only a small number of clusters, this computation will have to be repeated many times.

Agglomerative Hierarchical Clustering

- In the procedure below, c is the desired number of final clusters.

Algorithm 4 (Agglomerative hierarchical clustering)

1 begin initialize $c, \hat{c} \leftarrow n, \mathcal{D}_i \leftarrow \{\mathbf{x}_i\}, i = 1, \ldots, n$
2 do $\hat{c} \leftarrow \hat{c} - 1$
3 Find nearest clusters, say, $\mathcal{D}_i$ and $\mathcal{D}_j$
4 Merge $\mathcal{D}_i$ and $\mathcal{D}_j$
5 until $c = \hat{c}$
6 return c clusters
7 end

Agglomerative Hierarchical Clustering

- This procedure terminates when the specified number of clusters has been obtained and returns the clusters, described as set of points (rather than as mean or representative vectors).
- If we continue until c = 1 we can produce a dendrogram like that in Fig. 10.10.
- At any level the “distance” between nearest clusters can provide the dissimilarity value for that level.

$$d_{min}(\mathcal{D}_i, \mathcal{D}_j) = \min_{\substack{\mathbf{x} \in \mathcal{D}_i \\ \mathbf{x}' \in \mathcal{D}_j}} \|\mathbf{x} - \mathbf{x}'\| \tag{74}$$

$$d_{max}(\mathcal{D}_i, \mathcal{D}_j) = \max_{\substack{\mathbf{x} \in \mathcal{D}_i \\ \mathbf{x}' \in \mathcal{D}_j}} \|\mathbf{x} - \mathbf{x}'\| \tag{75}$$

$$d_{avg}(\mathcal{D}_i, \mathcal{D}_j) = \frac{1}{n_i n_j} \sum_{\mathbf{x} \in \mathcal{D}_i} \sum_{\mathbf{x}' \in \mathcal{D}_j} \|\mathbf{x} - \mathbf{x}'\| \tag{76}$$

$$d_{mean}(\mathcal{D}_i, \mathcal{D}_j) = \|\mathbf{m}_i - \mathbf{m}_j\|. \tag{77}$$

Agglomerative Hierarchical Clustering

- For agglomerative clustering algorithm the space complexity is $O(n^2)$.
- The full time complexity is thus $O(cn^2d^2)$, and in typical conditions $n>>c$ where n patterns and d-dimensional space is considered along with c clusters required.

Resources

- https://towardsdatascience.com/unsupervised-learning-and-data-clustering-eeecb78b422a
- https://sites.google.com/site/dataclusteringalgorithms/k-means-clustering-algorithm
- Pattern Classification-Duda, Hart & Stork.

End