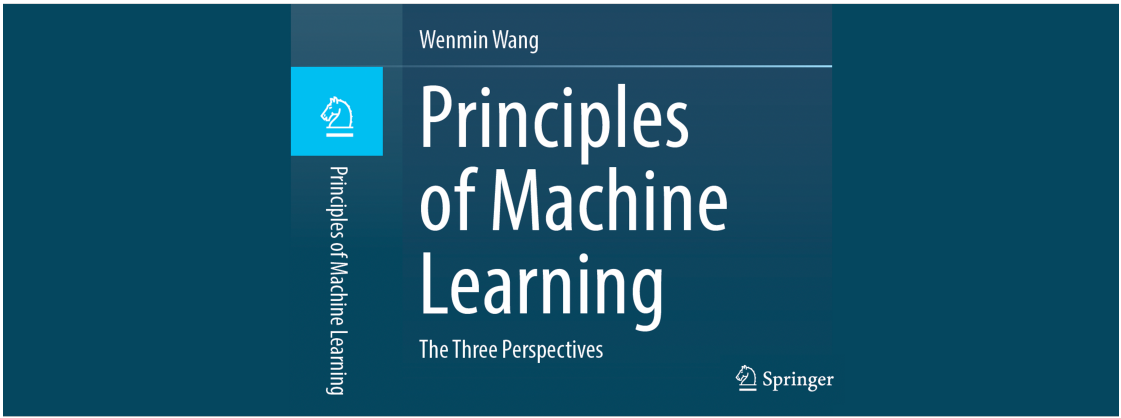



https://link.springer.com/book/10.1007/978-981-97-5333-8

# Part IV Tasks

# 14 Clustering Task

# 14 Clustering Task

# Clustering Problems

Clustering, also known as cluster analysis, is a task belongs to unsupervised learning paradigm.

It is to divide data in such a way that data in the same cluster are more similar to each other than to those in other clusters.

The clustering problems exist in many fields, e.g.,

medical image segmentation, three-dimensional reconstruction, customers or shopping goods grouping, social network analysis and grouping, information scene segmentation, and recommendation systems.

# Definition

**Definition**: (Clustering)

Clustering in machine learning is a task that is based on certain criteria to analyze input data and divide it into several groups called clusters. These clusters are unknown in advance, but there is some correlation between the data within the same cluster.

Clustering has two factors:

- one is to divide the data into clusters, and
- the other is the correlation within the same cluster.

# Clustering vs. Classification

Similarities and differences between them

| Clustering | Classification |
| --- | --- |
| Dividing input data into clusters according to certain criteria | Dividing input data into predefined classes |
| The clusters and their numbers are unknown beforehand | The categories and their numbers are known in advance |
| Without labeled training samples | With labeled training samples |

# 14 Clustering Task

# Formal Description

Let $\mathcal{X} \subseteq \mathbb{R}^m$ be the input space, and $\mathcal{G} \subseteq \mathbb{R}^m$ be the output space. There is a dataset with $n$ independent and identically distributed (i.i.d.) data:

$$D = \{\boldsymbol{x}_i \mid i = 1, \ldots, n\} \subseteq \mathcal{X},$$

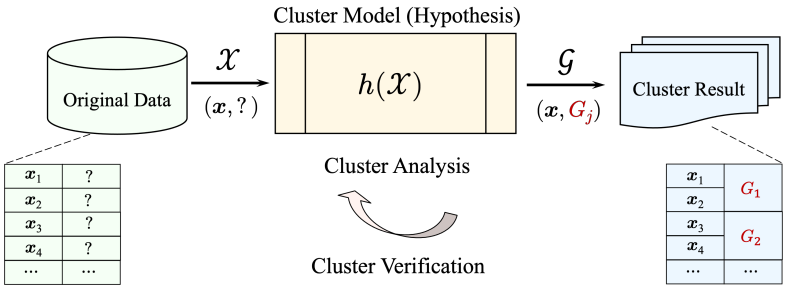
Let $H$ denote a set of hypotheses for clustering, the goal is to find an optimal hypothesis $h \in H$, which divides the dataset $D$ into $k$ clusters based on certain criteria, i.e,

$$h : \mathcal{X} \to \mathcal{G} \ .$$

Using the hypothesis $h(\mathcal{X})$ to cluster the $D$, we get a set of $k$ clusters $G$:

$$G = \{G_j \mid j = 1, \ldots, k\} \subseteq \mathcal{G}, \quad G_j \subseteq D, \text{and } k \leq n.$$

# Illustrated Description




The clustering process includes two important steps:

1) cluster analysis, and
2) result verification.

# 14 Clustering Task

# Hard and Soft

Just as classification methods are divided into hard and soft classification, clustering methods can also be divided into hard and soft clustering.

- Hard clustering:

  It determines whether an input data point belongs to a certain cluster, and the result is a definite value.

- Soft clustering:

  It determines the degree to which an input data point belongs to a certain cluster, and the result is expressed as a likelihood or fuzzy value.

# Linear and Nonlinear

Based on the data distribution to be clustered, the clustering problems can be divided into linearly separable and nonlinearly separable clustering.

- Linearly separable clustering:

  Clustering performed on a linearly separable dataset.

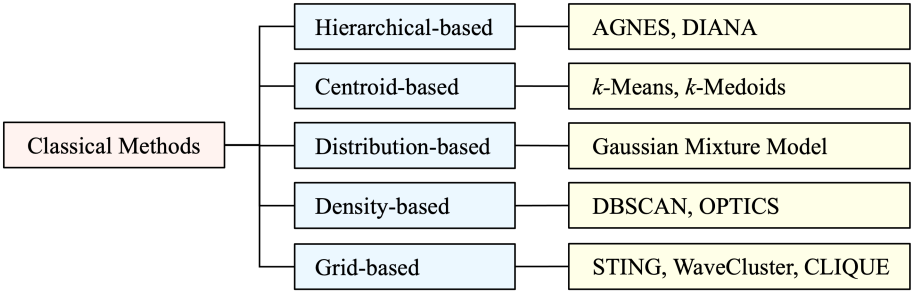
- Nonlinearly separable clustering:

  There are no restrictions similar to linearly separable clustering.

# 14 Clustering Task

# Classical Clustering Methods

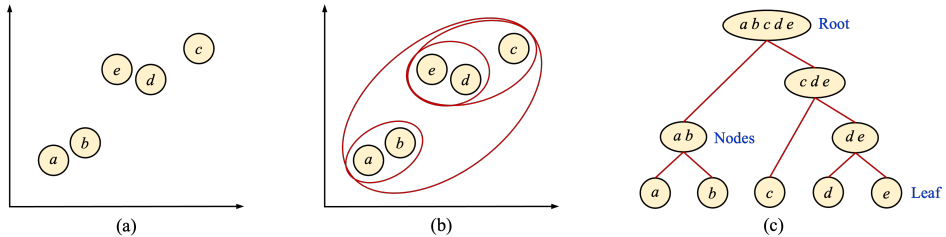
Be also known as traditional clustering methods.




Representative classical clustering methods and their representative algorithms

# Hierarchical-based Clustering

Idea: Data points that are closer to each other have a stronger relationship.




(a) several original data points,
(b) being divided into several clusters according to their distance,
(c) which can also be organized into a dendrogram.

# Hierarchical-based Clustering

Two different ways (bottom-up, and top-down) of hierarchical-based clustering:

- AGNES (AGglomerative NESting):

  Be a bottom-up method.

  1) each data point is regarded as a single-element cluster,
  2) the two closest clusters are merged into a larger cluster,
  3) repeated until all data points become members of a largest cluster.

- DIANA (DIvisive ANAlysis):

  Be a top-down method.

  it is the clustering process from root node to leaf nodes, opposite of bottom-up.

# Hierarchical-based Clustering

Let $G$ and $G'$ denote two different clusters, $\mathrm{dist}(\boldsymbol{x}, \boldsymbol{x}')$ be the distance between $\boldsymbol{x} \in G$ and $\boldsymbol{x}' \in G'$. Hierarchical clustering has three linkage criterions:

1) Single-linkage clustering:

$$\min \left\{ \mathrm{dist}(\boldsymbol{x}, \boldsymbol{x}') \middle| \, \boldsymbol{x} \in G, \boldsymbol{x}' \in G' \right\}.$$

2) Complete-linkage clustering:

$$\max \left\{ \mathrm{dist}(\boldsymbol{x}, \boldsymbol{x}') \middle| \, \boldsymbol{x} \in G, \boldsymbol{x}' \in G' \right\}.$$

3) Average-linkage clustering:

$$\frac{1}{|G| \cdot |G'|} \sum_{\boldsymbol{x} \in G} \sum_{\boldsymbol{x}' \in G'} \mathrm{dist}(\boldsymbol{x}, \boldsymbol{x}').$$

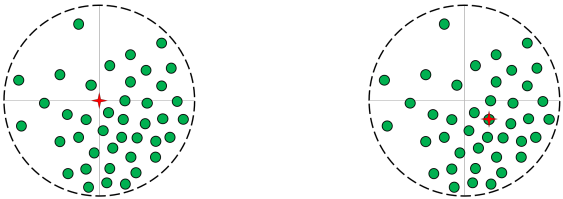
# Centroid-Based Clustering

(a) $k$-means:

Its centroid is the *center* of the cluster, i.e., the geometric center of the cluster, but not necessarily a data point in the cluster.

(b) $k$-medoids:

Its centroid is the *medoid* of the cluster, and is a data point in the cluster, but not necessarily the geometric center of the cluster.



# Distribution-Based Clustering

Be the clustering methods based on the probability distribution of data.

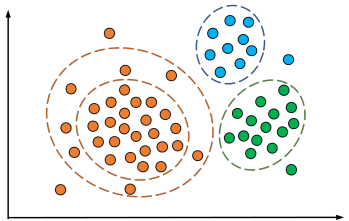
Typical method is Gaussian mixture model (GMM) using expectation maximization (EM) algorithm.

In GMM, a fixed number of Gaussian distributions are used.

They are randomly initialized, and parameters are optimized iteratively to better partition.

Multiple iterations may produce different results, and will eventually converge to a local optimum.

The data in the same cluster all belong to the same distribution.

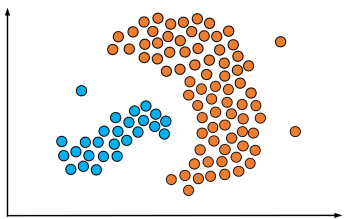


# Density-based Clustering

It accords to the density of the areas where the data is located, and divides each connected high-density data area into corresponding clusters.

Typical methods are DBSCAN (density-based spatial clustering of applications with noise) and OPTICS (ordering points to identify the clustering structure).

- DBSCAN: the minimum number of data points in high-density areas is defined by density threshold, and the maximum radius of high-density areas is determined by distance.
- OPTICS: a variant of DBSCAN, which improves the handling of different density clusters.



# Grid-based Clustering

It divides multidimensional data space into a finite number of cells to form a grid structure, and then forms clusters based on the cells in grid structure.

Each cluster corresponds to a different area, and the data points in it are denser than their surrounding environment.

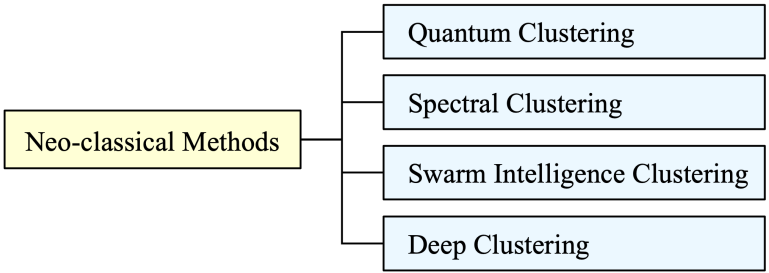
Representative grid-based clustering algorithms include:

- STING (Statistical Information Grid): a clustering algorithm based on statistical information grid.
- WaveCluster: uses wavelet transform for multi-resolution clustering.
- CLIQUE (Clustering in Ques): for high-dimensional data space clustering based on grid and density.

# 14 Clustering Task

# About Neo-classical Methods

Neo-classical clustering methods are also known as modern clustering ones.




Different from classical clustering methods, they use quantum theory, spectral graph theory, metaheuristics, and deep learning to implement clustering.

# Quantum Clustering

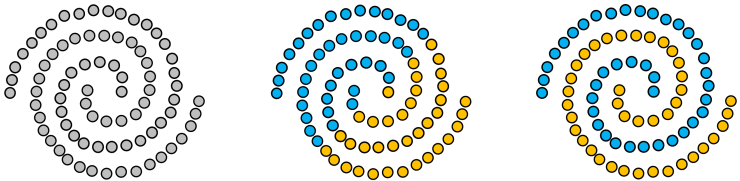
It uses quantum mechanics concepts and mathematical tools, to study the distribution of data in scale space based on the distribution of particles in the energy field.

- Original Quantum Clustering: using a multidimensional Gaussian distribution to represent data, with its width (standard deviation) $\sigma$, centered at each data point, then adding these Gaussian coefficients together to create a single distribution for entire dataset.
- Dynamic Quantum Clustering: replacing original gradient descent with a quantum evolution method, using time-dependent Schrödinger equation to calculate the evolution of each wave function in a given quantum potential over time, establishing a trajectory for each data.

# Spectral Clustering

It represents the dataset as a spectral graph, where nodes represent data and edges represent the similarity between data.

The similarity matrix is extracted from the spectral graph, and eigenvectors in the matrix are mapped to low-dimensional space for clustering.




(a) raw non-convex dataset, (b) result of $k$-means clustering, and (c) result of spectral clustering.

# Swarm Intelligence Clustering

It is the clustering using swarm intelligence.

Swarm intelligence refers to a series of AI algorithms inspired by biological collective behavior, which can provide sufficiently good solutions for combinatorial optimization problems and so on.

Clustering is essentially regarded as a kind of combinatorial optimization problem. Typical swarm intelligence clustering methods include:

1) Clustering based on ant colony optimization (ACO).
2) Clustering based on particle swarm optimization (PSO).
3) Clustering based on shuffled frog-leaping algorithm (SFLA).
4) Clustering based on artificial bee colony (ABC).

# Deep Clustering

Deep clustering, also known as deep learning-based clustering, is the use of deep neural networks for clustering.

Classical clustering methods often struggle when dealing with large, implicit patterns, high-dimensional, and complex datasets.

Deep clustering effectively solves the problems of these types of datasets.

Representative deep clustering can be divided into following categories:

1) Multistage deep clustering
2) Iterative deep clustering
3) Generative deep clustering
4) Simultaneous deep clustering

# 14 Clustering Task

# $k$-Means

The $k$-means is the most commonly used clustering algorithm.

It is aiming to divide $n$ input data into $k$ clusters, where each data point belongs to the cluster with nearest mean.

Given $n$ input data $D = \{\boldsymbol{x}_i \mid i = 1, \ldots, n\}$, where $\boldsymbol{x}_i \in \mathbb{R}^m$, the purpose of $k$-means algorithm is to divide the $D$ into $k$ clusters $G = \{G_j \mid j = 1, \ldots, k\}$.

Let $\mu_j$ be the mean of all data in the $j$th cluster, also known as the $j$th cluster center, then its objective function $f$ is:

$$f = \arg\min_{G} \sum_{j=1}^{k} \sum_{\boldsymbol{x}_i \in G_j} \|\boldsymbol{x}_i - \mu_j\|^2.$$

# $k$-Means

The algorithm proceeds by alternating between two steps:

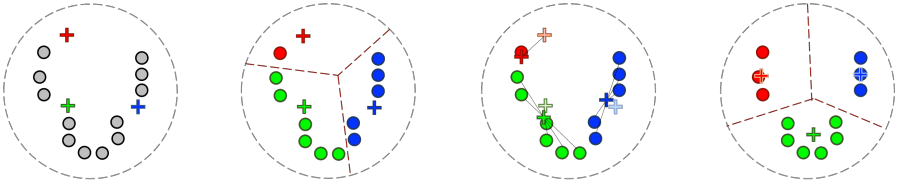
- Assignment: Each input data is assigned to a cluster such that the squared Euclidean distance of that cluster is minimized. That is: $\forall j'$, $1 \leq j' \leq k$, then

$$G_j^{(t)} = \left\{ \boldsymbol{x}_i \,\middle|\, \|\boldsymbol{x}_i - \mu_j^{(t)}\|^2 \leq \|\boldsymbol{x}_i - \mu_{j'}^{(t)}\|^2 \right\}.$$

- Update: Recalculate the mean of the input data assigned to each cluster to obtain the new cluster center. That is:

$$\mu_j^{(t+1)} = \frac{1}{\left|G_j^{(t)}\right|} \sum_{\boldsymbol{x}_i \in G_j^{(t)}} \boldsymbol{x}_i.$$

# $k$-Means




(a) Initial means, $k = 3$, are randomly generated within the data domain.

(b) The clusters are created by associating every data point with nearest mean. The partitions represent *Voronoi diagram* generated by the means.

(c) The center of each of the $k$ clusters becomes new mean.

(d) Repeat the steps (b) and (c), until convergence has been reached.

# Gaussian Mixture Clustering

Gaussian mixture model (GMM) for clustering assumes that all data points are generated by a mixture of finite Gaussians with unknown parameters.

Using expectation-maximization (EM) algorithm, its parameters are determined by maximum likelihood.

GMM superimposed by $k$ Gaussian distributions is defined as:

$$p(\boldsymbol{x}|\theta) = \sum_{j=1}^{k} \pi_j \mathcal{N}\left(\boldsymbol{x}|\mu_j, \sigma_j\right),$$

where $\mathcal{N}\left(\boldsymbol{x}|\mu_j, \sigma_j\right)$ is called a component of the mixture model, and $\pi_j$ is referred to as mixture coefficients, satisfying $0 \leq \pi_j \leq 1$ and $\sum_{j=1}^{k} \pi_j = 1$.

# Gaussian Mixture Clustering

Given input data $D = \{\boldsymbol{x}_i \mid i = 1, \ldots, n\}$, and $k$ clusters $G = \{G_j \mid j = 1, \ldots, k\}$.

Introduce a latent variable $z_i \in \{1, \ldots, k\}$, used to specify $\boldsymbol{x}_i$ which cluster $G_j$ it belongs to, the prior probability of $z_i$ is $p(z_i = j|\theta) = \pi_j$.

Calculate the posterior probability of $\boldsymbol{x}_i$ belonging to $j$th cluster $G_j$:

$$p(z_i = j|\boldsymbol{x}_i, \theta) = \frac{p(z_i = j|\theta)\, p(\boldsymbol{x}_i|z_i = j, \theta)}{\sum_{j'=1}^{k} p(z_i = j'|\theta) p(\boldsymbol{x}_i|z_i = j', \theta)}.$$

Let $1 - \max_j p(z_i = j|\boldsymbol{x}_i, \theta)$, use mean average precision (mAP):

$$z_i^* = \arg\max_j p(z_i = j|\boldsymbol{x}_i, \theta) = \arg\max_j (\log p(\boldsymbol{x}_i|z_i = j, \theta) + \log p(z_i = j|\theta)).$$

# DBSCAN

DBSCAN is the most representative density-based clustering algorithm.

Given a dataset $D$, and two kinds of data points $q, p \in D$.

Two parameters:

- Density threshold $\mathrm{minPts}$: the minimum number of data points in a high-density area;
- Distance threshold $\varepsilon$: the radius of high-density area.

Three types of data points:

- Core point: if $\exists\, q \in D$, and $|N_\varepsilon(q)| \geq \mathrm{minPts}$.
- Neighbor: if $\exists\, p \in D$, and $N_\varepsilon(p) : \{p \mid \mathrm{dist}(p, q) \leq \varepsilon\}$;
- Outlier: if the data point is not core point or neighbor.

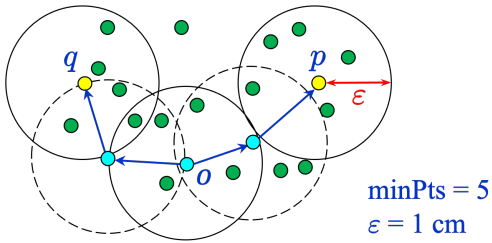
# DBSCAN

Density-reachable:

- A data point $p$ is density-reachable from the data point $q$, if there is a chain of data points $p_1, \ldots, p_n$, such that $p_1 \to q$, $p_n \to p$, and $p_{i+1}$ can be directly density-reachable from $p_i$.

Density-connected:

- A data point $p$ is density-connected to the data point $q$, if there exists a data point $o$, such that data points $p$ and $q$ can be density-reachable from $o$.



# Density Peak Clustering

One of the main problems of clustering is how to find the centroid of a cluster.

Density peak clustering is a novel method for quickly finding cluster centers.

Two basic ideas of this method: a cluster center is characterized by

- a higher *density* than their neighbors.
- a larger *distance* from points with higher densities.

The features of this method:

- the number of clusters arises intuitively;
- outliers (cluster centers) are automatically spotted;
- clusters can be recognized regardless of their shape and the dimensionality of the space.

# Density Peak Clustering

it defines two variables $\rho_i$ and $\delta_i$ correspond to the above two basic ideas.

The $\rho_i$ denotes the local density of $i$th data point:

$$\rho_i = \sum_j \mathbb{I}\left(d_{ij} - d_c\right),$$

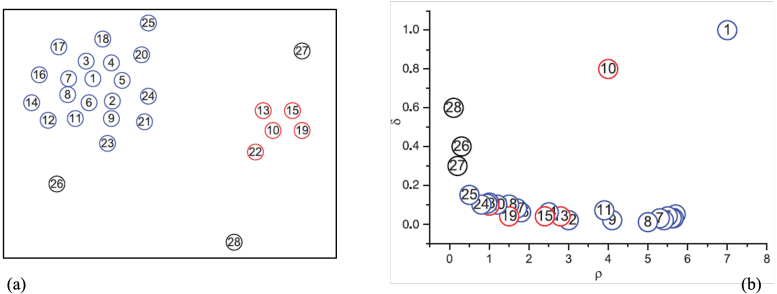
where $d_{ij}$ is distance between $i$th and $j$th data points, $d_c$ is cut-off distance, and $\mathbb{I}(\omega)$ is indicator function: equal 1 if $\omega < 0$ and 0 otherwise.

The $\delta_i$ is minimum distance between $i$th and $j$th data points: $\delta_i = \min_{j:\rho_j > \rho_i} (d_{ij})$.

For the point with highest density, it conventionally takes: $\delta_i = \max_j (d_{ij})$.

# Density Peak Clustering

*Image source*: https://www.science.org/doi/10.1126/science.1242072




(a) The 28 data points are ranked in order of decreasing density.
(b) Decision graph for the 28 data points.
Different colors correspond to different clusters.

# 14 Clustering Task

# About Evaluation Metrics for Clustering

As a task of unsupervised learning, there are two types of evaluation metrics:

- Extrinsic evaluation measures: given manually labeled clustering benchmarks, and use them to compare with the results of clustering.
- Intrinsic evaluation measures: without any manually labeled benchmarks, directly evaluate the clustering algorithm and its clustering results.

| | Extrinsic Evaluation Measures | Intrinsic Evaluation Measures |
|---|---|---|
| Evaluation Metrics | • Adjusted Rand Index<br>• Adjusted Mutual Information<br>• Fowlkes-Mallows Index<br>• V-Measure | • Calinski-Harabasz Index<br>• Davies-Bouldin Index<br>• Silhouette Coefficient |

# Adjusted Rand Index

The adjusted Rand index (ARI) is a function based on the Rand index (RI).

Given a dataset $D = \{\boldsymbol{x}_i \mid i = 1, \ldots, n\}$, labeled cluster set and predicted cluster set $U = \{U_k \mid k = 1, \ldots, r\}$ and $V = \{V_l \mid l = 1, \ldots, s\}$. Let

$$a = \left|\hat{D}\right|, \text{ and } \hat{D} = \{\langle \boldsymbol{x}_i, \boldsymbol{x}_{i'} \rangle \mid \boldsymbol{x}_i, \boldsymbol{x}_{i'} \in U_k, \boldsymbol{x}_i, \boldsymbol{x}_{i'} \in V_l\},$$

$$b = \left|\hat{D}\right|, \text{ and } \hat{D} = \{\langle \boldsymbol{x}_i, \boldsymbol{x}_{i'} \rangle \mid \boldsymbol{x}_i \in U_k, \boldsymbol{x}_{i'} \in U_{k'}, \boldsymbol{x}_i \in V_l, \boldsymbol{x}_{i'} \in V_{l'}\},$$

where $\langle \boldsymbol{x}_i, \boldsymbol{x}_{i'} \rangle$ is a pair of data, and $i \neq i'$. The RI is defined as:

$$\mathrm{RI} = \frac{a+b}{\binom{n}{2}}, \quad \text{where } \binom{n}{2} = C_2^n = \frac{n(n-1)}{2}.$$

# Adjusted Rand Index

The range of RI is $[0, 1]$, where 0 indicates that the predicted cluster set $V$ is completely different from the labeled cluster set $U$ (i.e., $V \neq U$), and 1 indicates that $V$ is completely identical to $U$ (e.g., $V = U$).

The adjusted Rand index (ARI) is defined as:

$$\mathrm{ARI} = \frac{\mathrm{RI} - \mathbb{E}[\mathrm{RI}]}{\max(\mathrm{RI}) - \mathbb{E}[\mathrm{RI}]},$$

where $\mathbb{E}[\mathrm{RI}]$ is expected value of RI, and $\max(\mathrm{RI})$ is maximum value of RI.

The range of ARI is $[-1, 1]$, the larger the value, the better the similarity between the predicted cluster set $V$ and the labeled cluster set $U$.

# Adjusted Mutual Information

Given a dataset with $n$ data, labeled cluster set $U = \{U_k \mid k = 1, \ldots, r\}$, and predicted cluster set $V = \{V_l \mid l = 1, \ldots, s\}$. Then the entropy of $U$ and $V$:

$$\mathrm{H}(U) = -\sum_{k=1}^{r} P(k) \log\left(P(k)\right), \quad \text{where} \quad P(k) = |U_k|/n$$

$$\mathrm{H}(V) = -\sum_{l=1}^{s} P'(l) \log\left(P'(l)\right), \quad \text{where} \quad P(l) = |V_l|/n.$$

The mutual information (MI) between $U$ and $V$ can be calculated by:

$$\mathrm{MI}(U, V) = \sum_{k=1}^{r} \sum_{l=1}^{s} P(k, l) \log\left(\frac{P(k, l)}{P(k)P'(l)}\right), \quad \text{where} \quad P(k, l) = |U_k \cap V_l|/n.$$

# Adjusted Mutual Information

$\mathrm{MI}(\cdot)$ is a non-negative value, with entropy $\mathrm{H}(U)$ and $\mathrm{H}(V)$ as upper bound.

The normalized mutual information (NMI) is defined by entropy and MI:

$$\mathrm{NMI} = \frac{\mathrm{MI}(U, V)}{\mathrm{mean}(\mathrm{H}(U), \mathrm{H}(V))}.$$

The range of NMI is $[0, 1]$. The adjusted mutual information (AMI) is defined by.

$$\mathrm{AMI} = \frac{\mathrm{MI} - \mathbb{E}[\mathrm{MI}]}{\mathrm{mean}(\mathrm{H}(U), \mathrm{H}(V)) - \mathbb{E}[\mathrm{MI}]}.$$

The AMI is equal to 1 if $V$ and $U$ are the same, and AMI is 0 if the MI between $V$ and $U$ is equal to the expected value.

# Fowlkes-Mallows Index

Fowlkes-Mallows index (FMI) is used to measure similarity of two clusterings.

Given $n$ data, and two clustering trees $C$ and $C'$. For each value of $k$, the following matrix can be generated, $\mathrm{M} = (m_{i,j})$, where $i, j = 1, \ldots, k$, and $m_{i,j}$ is the number of data points in $i$th cluster in $C$ and $j$th cluster in $C'$.

Therefore, the FMI for a specific value $k$ is defined as follows:

$$\mathrm{FMI}_k = \frac{T_k}{\sqrt{P_k \cdot Q_k}}, \quad \text{where}$$

$$T_k = \sum_{i=1}^{k}\sum_{j=1}^{k} m_{i,j}^2 - n, \quad P_k = \sum_{i=1}^{k}\left(\sum_{j=1}^{k} m_{i,j}\right)^2 - n, \quad Q_k = \sum_{j=1}^{k}\left(\sum_{i=1}^{k} m_{i,j}\right)^2 - n.$$

# Fowlkes-Mallows Index

In summary, by calculating the $\mathrm{FMI}_k$ for different values of $k$, we can judge the similarity of two hierarchical clustering algorithms.

The $\mathrm{FMI}_k$ can also be generalized to measure the similarity of two clusterings with different numbers of clusters or non-hierarchical clusterings.

For another FMI, let $U$, $V$ be labeled cluster set, predicted cluster set, and

- TP as the number of data point pairs that appear in same cluster in $U$ and $V$,
- FP as the number of data point pairs that appear in same cluster in $U$ but not $V$,
- FN as the number of data point pairs that appear in same cluster of $V$ but not $U$,

Then, we have

$$\mathrm{FMI} = \frac{\mathrm{TP}}{\sqrt{(\mathrm{TP}+\mathrm{FP})\,(\mathrm{TP}+\mathrm{FN})}}.$$

# V-Measure

V-measure is based on two metrics of cluster assignments:

- Homogeneity: each cluster contains the members of one category only.
- Completeness: same category members are assigned to same cluster.

Given a dataset with $n$ data, labeled cluster set $U = \{U_k | \ k = 1, \ldots, r\}$, predicted cluster set $V = \{V_l | \ l = 1, \ldots, s\}$. And let $\mathrm{M} = (m_{k,l})$ be a confusion matrix generated by $U$ and $V$.

So, the formal definitions of homogeneity (h) and completeness (c) are:

$$h = 1 - \frac{\mathrm{H}(U|V)}{\mathrm{H}(U)}, \qquad c = 1 - \frac{\mathrm{H}(V|U)}{\mathrm{H}(V)}.$$

# V-Measure

$$\mathrm{H}(U|V) = -\sum_{k=1}^{r}\sum_{l=1}^{s}\frac{m_{k,l}}{n}\log\left(\frac{m_{k,l}}{m_{\cdot,l}}\right), \quad \mathrm{H}(V|U) = -\sum_{l=1}^{s}\sum_{k=1}^{r}\frac{m_{k,l}}{n}\log\left(\frac{m_{k,l}}{m_{k,\cdot}}\right).$$

$$\mathrm{H}(U) = -\sum_{k=1}^{r}\frac{m_{k,\cdot}}{n}\log\left(\frac{m_{k,\cdot}}{n}\right), \quad \mathrm{H}(V) = -\sum_{l=1}^{s}\frac{m_{\cdot,l}}{n}\log\left(\frac{m_{\cdot,l}}{n}\right),$$

$$m_{\cdot,l} = \sum_{k=1}^{r} m_{k,l}, \quad m_{k,\cdot} = \sum_{l=1}^{s} m_{k,l}.$$

Therefore, V-measure is defined as the harmonic mean of $h$ and $c$, i.e.,

$$\mathrm{V}_{\text{measure}} = 2 \cdot \frac{h \cdot c}{h + c}.$$

# Calinski-Harabasz Index

Calinski-Harabasz index (CHI) is a measure of cohesion and separation.

Given a dataset $D = \{\boldsymbol{x}_i \mid i = 1, \ldots, n\}$, and assign $n$ data points to $k$ clusters,

$$\text{CHI} = \left[\frac{B_k}{(k-1)}\right] / \left[\frac{W_k}{(n-k)}\right].$$

$$B_k = \sum_{j=1}^{k} n_k \|c_k - c\|^2, \quad W_k = \sum_{j=1}^{k} \sum_{i=1}^{n_k} \|\boldsymbol{x}_i - c_k\|^2,$$

where $n_k$ and $c_k$ are the number of data points and the centroid in the $k$th cluster respectively, and $c$ is the centroid of all data points in the dataset.

The higher the CHI, the better the clustering model.

# Davies-Bouldin Index

Davies-Bouldin index (DBI) evaluates the clustering model by average similarity between clusters.

Given $N$ clusters, let $G_j$ denote $j$th cluster, $\boldsymbol{x}_i$ be a data point in $G_j$, and $r_j$ be the centroid of $G_j$, then the average distance $S_j$ between each data point in $G_j$ and its centroid is:

$$S_j = \left( \frac{1}{|G_j|} \sum_{i=1}^{|G_j|} \|\boldsymbol{x}_i - r_j\|_p^q \right)^{1/q}.$$

where if $q = 1$, then $S_j$ is the average Euclidean distance, while the value of $p$ is usually set to 2, making it a Euclidean distance function.

# Davies-Bouldin Index

The deviation rate between clusters $G_j$ and $G_{j'}$ is as follows:

$$M_{j,j'} = \|G_j - G_{j'}\|_p, \qquad R_{j,j'} = \frac{S_j + S_{j'}}{M_{j,j'}},$$

where $p$ is set to 2, and $R_{j,j'}$ is the quality of clustering scheme, must consider:

- Divergence between clusters $j$ and $j'$ should be as large as possible.
- Cohesion of the cluster $j$ should be as low as possible.

$$\mathrm{DBI} = \frac{1}{N}\sum_{j=1}^{N} \max_{j \neq j'} R_{j,j'}.$$

The lower the DBI, the better the clustering model.

# Silhouette Coefficient

Silhouette coefficient (SC) is used for each data point to measure the cohesion with its cluster, and the separation rate from neighboring cluster.

Let $\boldsymbol{x}_i$ and $\boldsymbol{x}_{i'}$ denote data points, and $G_{\text{own}}$ and and $G_{\text{next}}$ be owner and neighbor cluster. For $\boldsymbol{x}_i, \boldsymbol{x}_{i'} \in G_{\text{own}}$:

$$a(i) = \frac{1}{|G_{\text{own}}| - 1} \sum_{\boldsymbol{x}_{i'} \in G_{\text{own}}} d(\boldsymbol{x}_i, \boldsymbol{x}_{i'}), \quad b(i) = \min_{G_{\text{own}} \neq G_{\text{next}}} \frac{1}{|G_{\text{next}}|} \sum_{\boldsymbol{x}_{i'} \in G_{\text{next}}} d(\boldsymbol{x}_i, \boldsymbol{x}_{i'}).$$

So, the SC of data point $\boldsymbol{x}_i$ is defined as below, its range of $\text{SC}(i)$ is $[-1, +1]$:

$$\text{SC}(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}.$$

# Thank You