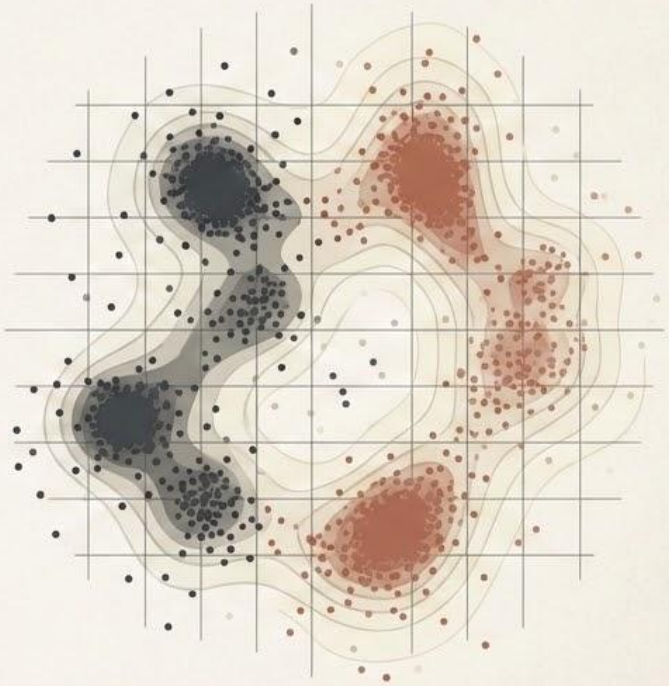
Density-Based Clustering, Grid Methods, and Cluster Evaluation

Lecture Details

- **Duration:** 1 Hour
- **Target Audience:** Data Science/Computer Science students
- **Prerequisites:** Understanding of basic clustering concepts, distance measures, k-means, hierarchical clustering



Moving Beyond Traditional Clustering

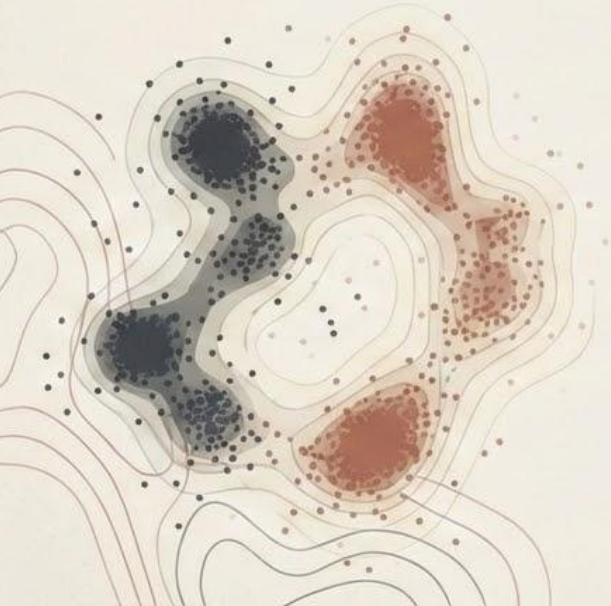
Finding Shape in the Noise

- **The Scenario:** Analyzing satellite imagery to identify urban areas.
- **The Problem**: Real-world clusters aren't always neat, uniform circles. They form arbitrary shapes with highly varying densities.
- **The Limitation**: Traditional methods (like k-means) struggle with non-spherical shapes and heavy noise.
- **The Question**: How do we accurately find clusters of any shape, handle environmental noise, and determine if our results are actually meaningful?

Today's Learning Objectives

Learning Objectives

Mastering Density, Grids, and Evaluation

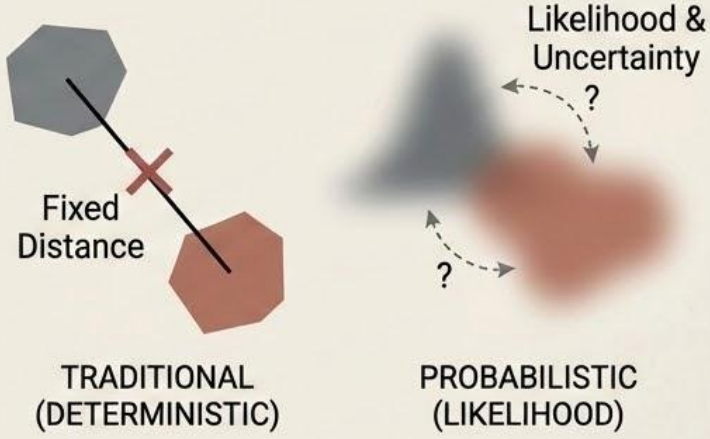


- **Understand** probabilistic hierarchical clustering.
- **Master** density-based clustering algorithms (specifically DBSCAN and DENCLUE).
- **Learn** the mechanics of grid-based clustering methods.
- **Evaluate** clustering quality using both intrinsic and extrinsic measures.
- **Determine** the mathematically optimal number of clusters for a given dataset.

Probabilistic Hierarchical Clustering

Moving Beyond Deterministic Distances

- **The Limitation of Traditional Methods:** Traditional hierarchical clustering relies on rigid, deterministic distance measures. It lacks any notion of uncertainty.
- **The Probabilistic Shift:** Instead of measuring straight-line distance, we use probabilistic models to measure cluster similarity.
- **The Key Concept:** We make merging decisions based on likelihood—how likely it is that two clusters belong to the same underlying distribution—rather than spatial proximity.



Example 8.6. Generative model. Suppose we are given a set of 1-D points $X = \{x_1, \ldots, x_n\}$ for clustering analysis. Let us assume that the data points are generated by a Gaussian distribution,

$$\mathcal{N}(\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}, \tag{8.15}$$

where the parameters are μ (the mean) and σ^2 (the variance).

The probability that a point $x_i \in X$ is then generated by the model is

$$P(x_i|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_i-\mu)^2}{2\sigma^2}}. \tag{8.16}$$

Consequently, the likelihood that the data set X observed is generated by the model is

$$L(\mathcal{N}(\mu, \sigma^2) : X) = P(X|\mu, \sigma^2) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_i-\mu)^2}{2\sigma^2}}. \tag{8.17}$$

The task of learning the generative model is to find the parameters μ and σ^2 such that the likelihood $L(\mathcal{N}(\mu, \sigma^2) : X)$ is maximized, that is, finding

$$\mathcal{N}(\mu_0, \sigma_0^2) = \arg\max\{L(\mathcal{N}(\mu, \sigma^2) : X)\}, \tag{8.18}$$

where $\max\{L(\mathcal{N}(\mu, \sigma^2) : X)\}$ is called the *maximum likelihood*. □

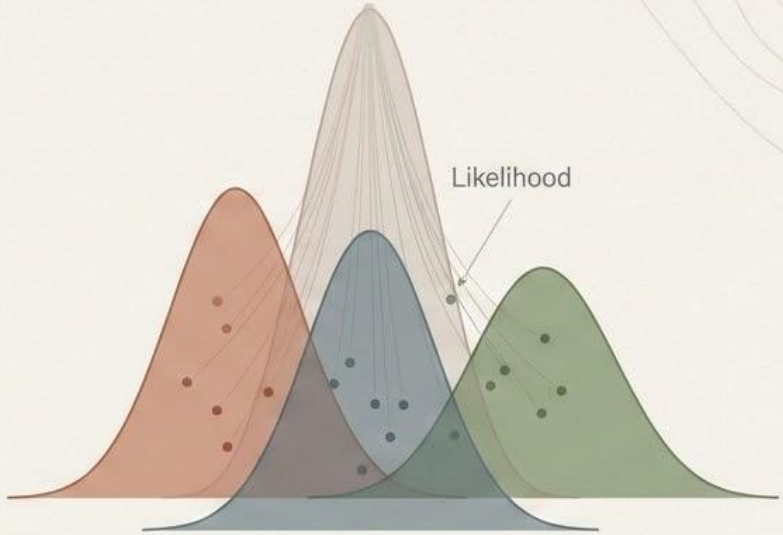
The Generative Model View

Clusters as Probability Distributions

Clusters = Distributions: Think of each cluster not as a physical shape, but as a specific probability distribution (e.g., a Gaussian distribution).

Data Generation: The core assumption is that the objects within a cluster were mathematically generated from that specific distribution.

Measuring the Fit: We use likelihood to measure exactly how well our proposed model fits the actual data points.



The Clustering Algorithm

Maximizing Likelihood Step-by-Step

Step 1: Initialization

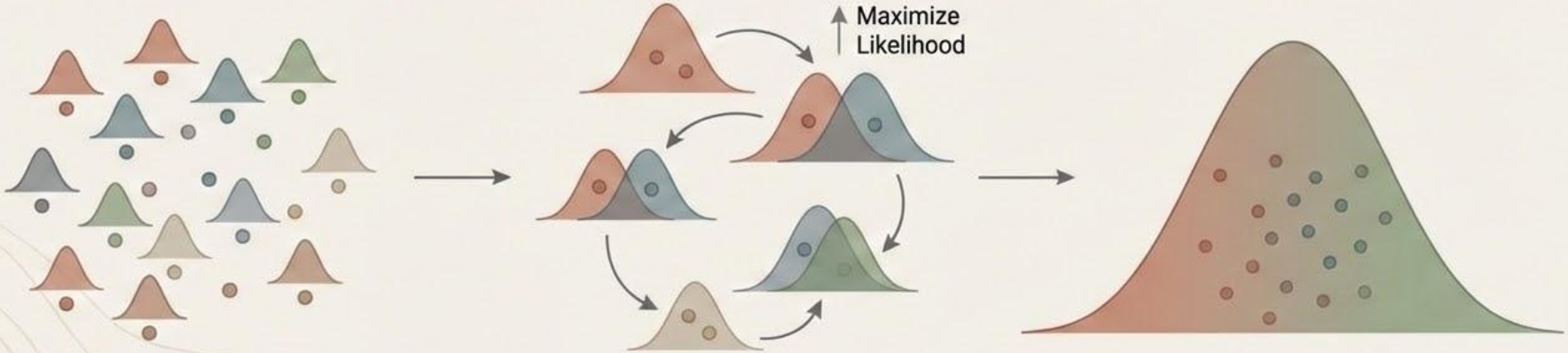
- Start with every single data point acting as its own individual cluster with its own unique distribution.

Step 2: The Merge

Iteratively merge the clusters that maximize the overall likelihood (or minimize the statistical loss) of the model.

Step 3: Termination

Continue this merging process until all points are unified into one single, overarching cluster.

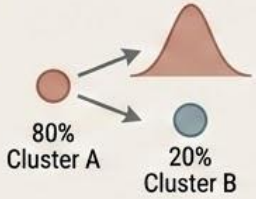


Advantages & Model Connections

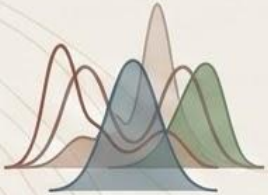
Why Choose a Probabilistic Approach?

Core Advantages

Handles uncertainty naturally.

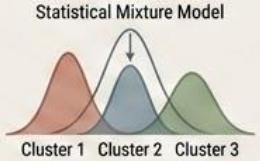


Provides probabilistic “soft” cluster assignments (e.g., 80% chance it belongs to Cluster A, 20% to Cluster B).



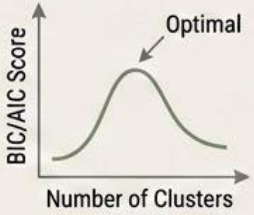
Can easily handle highly complex, overlapping data distributions.

Connection to Model-Based Clustering



Each cluster corresponds directly to a component in a statistical mixture model.

Merging decisions are backed by rigorous statistical tests.



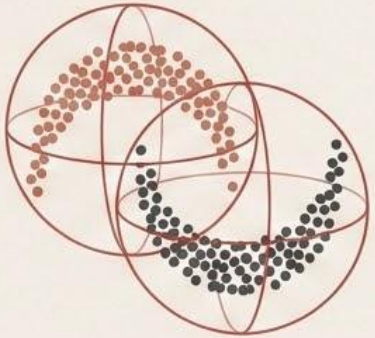
We can mathematically determine the optimal number of clusters using criteria like BIC (Bayesian Information Criterion) or AIC (Akaike Information Criterion).

Density-Based Clustering

Finding Arbitrary Shapes in the Noise

The Problem with Partitioning Methods

Algorithms like k-means require clusters to be roughly spherical or ellipsoidal. They fail completely when clusters have complex, arbitrary shapes.

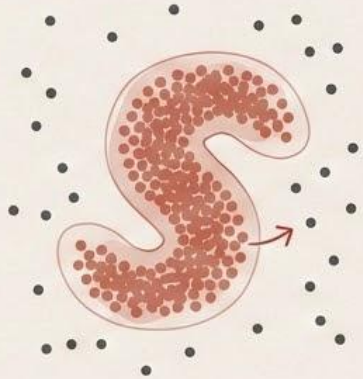


The DBSCAN Core Idea

A cluster is a dense region of points in the data space, separated from other clusters by sparse regions.

The Goal

Locate these high-density regions and grow the clusters iteratively from them.

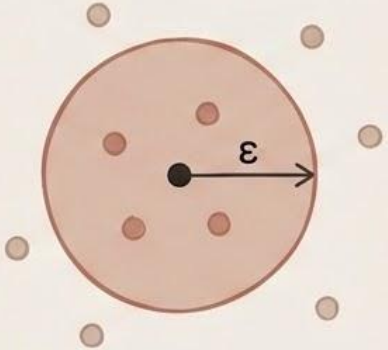


DBSCAN: Key Parameters

Defining What "Dense" Means

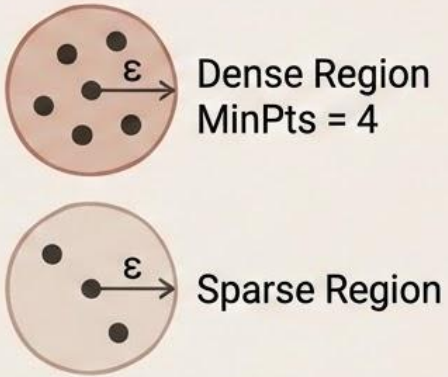
DBSCAN only requires two parameters to operate:

ε (Epsilon): The radius of the neighborhood around a specific point.



(Think: “My immediate surroundings.”)

MinPts: The minimum number of points required within that radius to be considered a dense region.



Sets the threshold for density.

Point Classification

Defining Core, Border, and Noise Points

Core Point

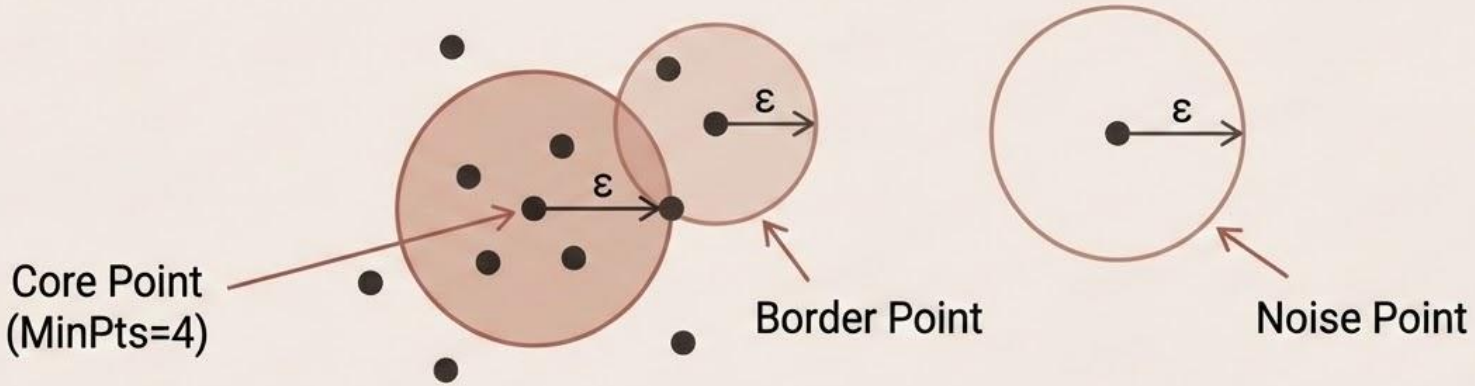
A point that has at least the minimum number of points (MinPts) within its ε-neighborhood. These are the hearts of clusters.

Border Point

A point that falls within the ε-neighborhood of a Core Point but has fewer than MinPts in its own immediate neighborhood.

Noise Point

An isolated point that is neither a Core Point nor a Border Point. These are ignored by the algorithm.

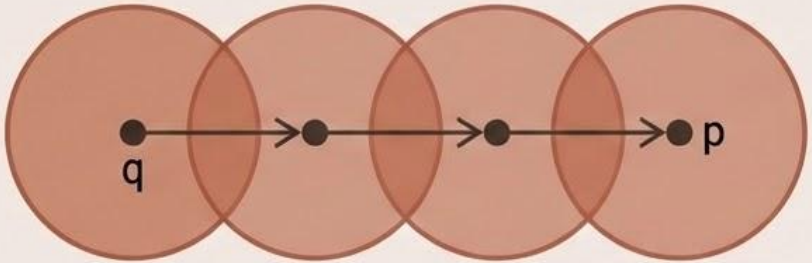


Concepts of Density

Defining Point Connectivity: Reachability vs. Connectivity

Density Reachability

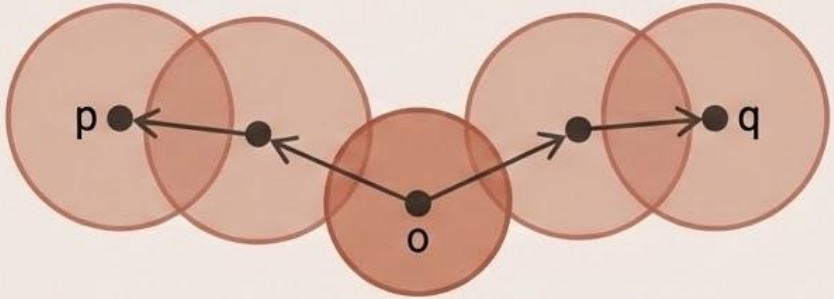
A point p is density-reachable from point q if there is a chain of core points leading from q to p, where each step in the chain is within the ε radius.



Note: This is not symmetric. A border point is reachable from a core point, but the border point can't 'reach' back, because it isn't dense enough to start its own neighborhood chain.

Density Connectivity

Two points p and q are density-connected if there exists an independent point o such that both p and q are density-reachable from that common point o.



This concept defines the actual clusters. If two points are connected via dense chains, they must belong together.

Example 8.7. Density-reachability and density-connectivity. Consider Fig. 8.17 for a given ϵ represented by the radius of the circles, and, say, let $MinPts = 3$.

Of the labeled objects, $\boldsymbol{p}$, $\boldsymbol{m}$, $\boldsymbol{o}$, $\boldsymbol{q}$, and $\boldsymbol{t}$ are core objects, since each of the ϵ-neighborhoods (dashed circles in the figure) of them contains at least three objects. Objects $\boldsymbol{p}$ and $\boldsymbol{o}$ are ϵ-reachable, so are $\boldsymbol{o}$ and $\boldsymbol{q}$. Thus $\boldsymbol{p}$ and $\boldsymbol{q}$ are density-connected.

It can be verified that the core objects $\boldsymbol{p}$, $\boldsymbol{m}$, $\boldsymbol{o}$, $\boldsymbol{q}$, and $\boldsymbol{t}$ form a cluster, since each two among them are density-connected and no other core objects can be added into this group so that the pairwise density-connectivity is maintained.

Object $\boldsymbol{s}$ is not a core object, since the ϵ-neighborhood of s contains only two objects. However, $\boldsymbol{s}$ is in the ϵ-neighborhood of core object $\boldsymbol{p}$, thus $\boldsymbol{s}$ is a border object.

Objects $\boldsymbol{u}$ and $\boldsymbol{v}$ are not core objects, and they do not belong to the ϵ-neighborhood of any core objects. Thus they are outliers. □

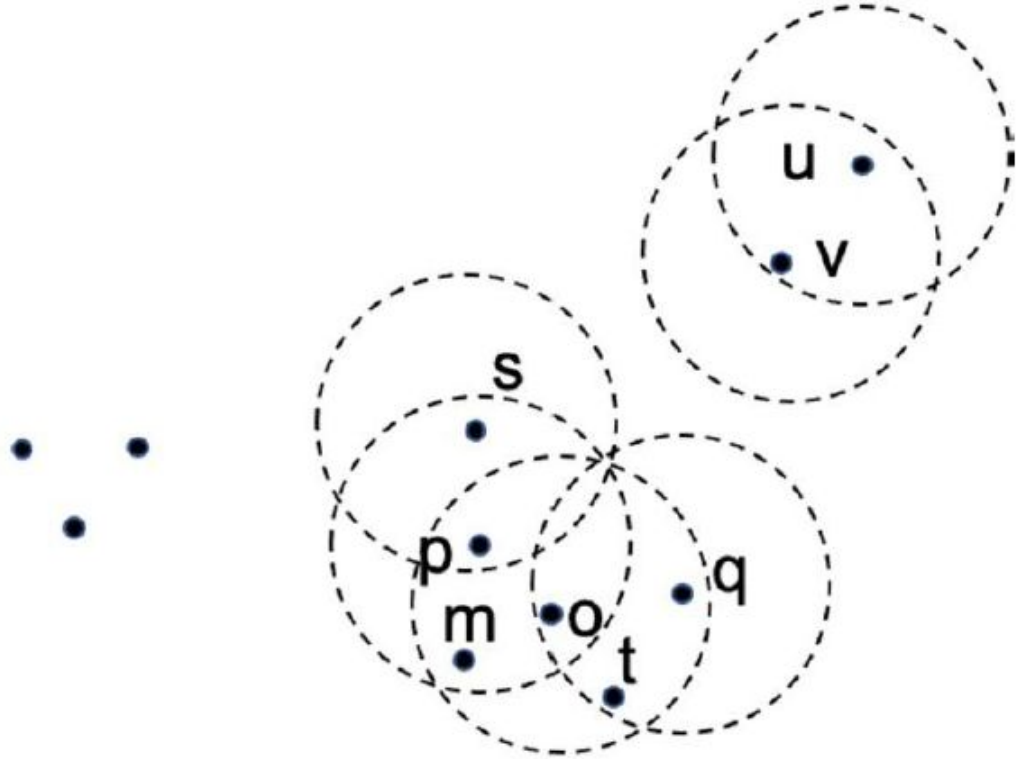


FIGURE 8.17

Density-reachability and density-connectivity in DBSCAN.

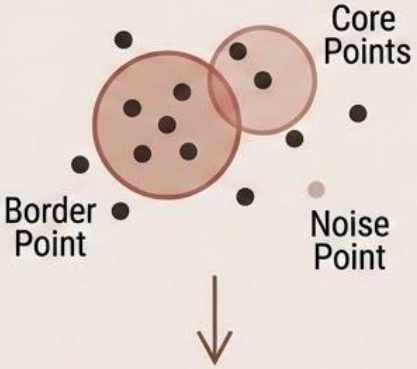
The DBSCAN Algorithm

High-Level Procedure

The process is executed systematically:

Label:

Label every single point in the dataset as either Core, Border, or Noise.



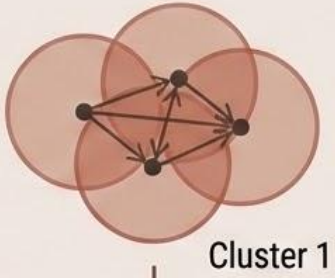
Clean:

Eliminate all Noise points from the system.

Process Core:

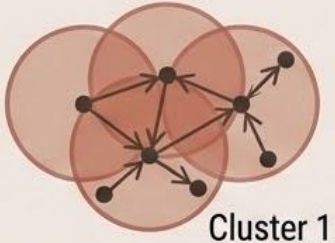
For each Core point that has not yet been processed:

- Assign it to a new, unique cluster.
- Find all density-reachable core points and add them to the same cluster.



Process Border:

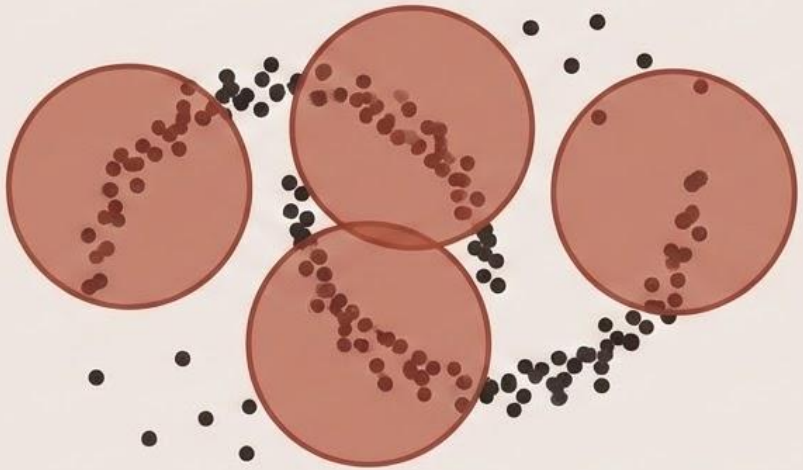
Finally, assign all remaining Border points to the cluster of their associated Core point.



DBSCAN Visual Example

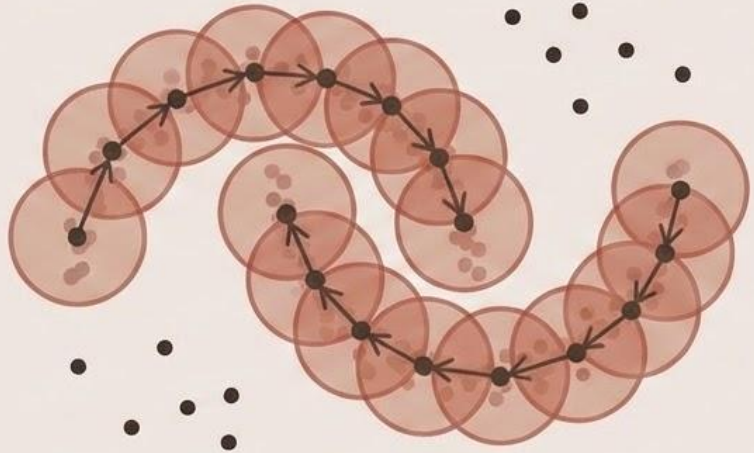
Finding Arbitrary Shapes

k-means (Fails)



k-means: Assumes spherical clusters, fails to capture non-convex shapes and noise.

DBSCAN (Succeeds)

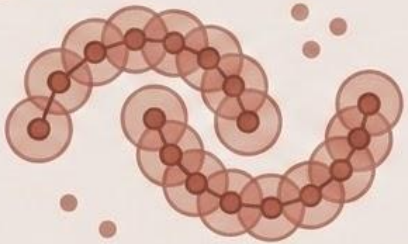


DBSCAN: Identifies arbitrary shapes based on density, effectively handling noise.

Evaluating DBSCAN

When to Use It (And When Not To)

Advantages



- Can successfully find clusters of arbitrary shape.
- Handles noise naturally, stripping out isolated points that are not mathematically part of a dense cluster.
- Requires only two primary input parameters.
- It is Deterministic: It always produces the same clusters for the same data and parameters on every run.

Disadvantages

- Highly sensitive to the choice of ε and MinPts parameters.
- Struggles when the data has varying densities across the space.
- Cannot handle very high-dimensional data well due to the “curse of dimensionality” impacting density calculations.

Selecting Parameters (ε and MinPts)

Using the k-Distance Graph

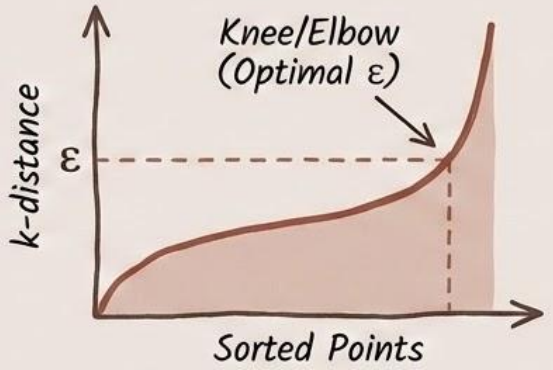
Since parameters are so critical, how do we select them responsibly?

MinPts:

- A simple default rule is that for 2D data, MinPts is typically set to 4 or 5.

ε (Epsilon):

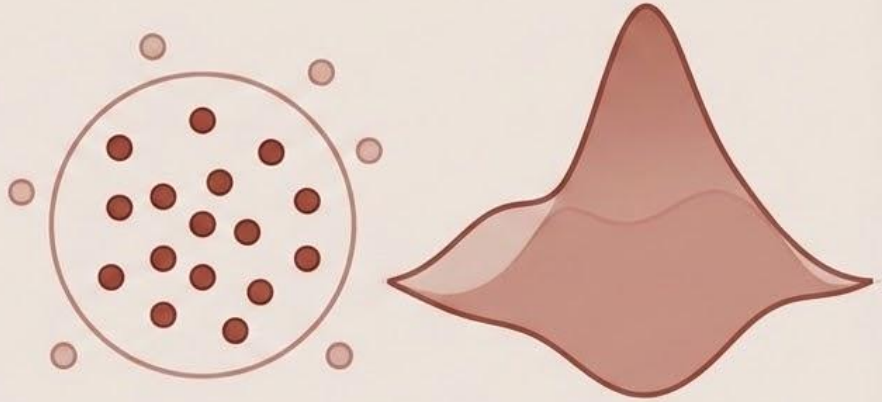
- The optimal value is found by analyzing the sorted ‘k-distance graph’ and looking for the point of maximum curvature—the ‘knee’ or ‘elbow’—which indicates the optimal dense radius.



DENCLUE: Density Distribution Functions

A Continuous Approach to Density

- The Limitation of DBSCAN: While DBSCAN is powerful, it enforces a strict ‘hard boundary’ between core points and non-core points, and it lacks a probabilistic interpretation.
- The DENCLUE Core Idea: Instead of drawing hard radii, DENCLUE uses Kernel Density Estimation (KDE).
- The Result: It creates a smooth, continuous density function across the entire data space, where clusters are simply the peaks (local maxima) of that topographical density map.



DBSCAN:
Hard Boundary

DENCLUE:
Continuous Density Peak

The Mathematics of Density

Kernel Density Estimation (KDE)

Calculating the Continuous Function

The overall density function for a given point x is calculated by summing the influence of all other points in the dataset:

$$f(x) = \sum_{i=1}^{n} K(x - x_i)$$

- $f(x)$: The overall estimated density at point x.
- n: The total number of data points.
- K: The kernel function, which defines how much influence a point x_i exerts on point x.

Note: The kernel function K is typically a Gaussian (normal) distribution.

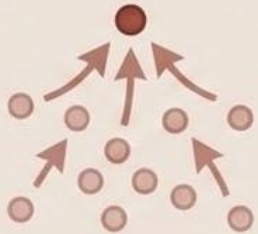
The Mechanics of Attraction

How Points Find Their Clusters

DENCLUE defines clusters using the concept of **gravitational flow**:

Density Attractor

A local maximum (peak) of the overall density function. Think of this as the very top of a mountain.



Density-Attracted Points

Data points that mathematically “flow” upward al gradient toward a specific Density Attractor.

Clusters

A set of all points that are successfully attracted to the exact same Density Attractor.

The DENCLUE Algorithm

The Algorithm in Action: Step-by-Step Execution

1. Compute

Calculate the continuous density function for all points in the dataset.

2. Identify

Locate the Density Attractors (the local maxima) by using a mathematical gradient ascent (climbing to the highest point).

3. Assign

Determine the flow of each data point and assign it to the specific attractor it flows toward.

4. Merge

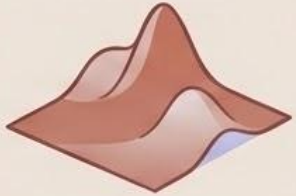
If two Density Attractors are separated by a shallow density "valley," merge them into a single, complex cluster.

The Advantages of DENCLUE

Why Use a Continuous Estimate?

Shape Flexibility:
Effectively handles clusters of completely arbitrary shapes.

Noise Resistance:
Highly robust to background noise and outliers (they don't generate strong attractors).

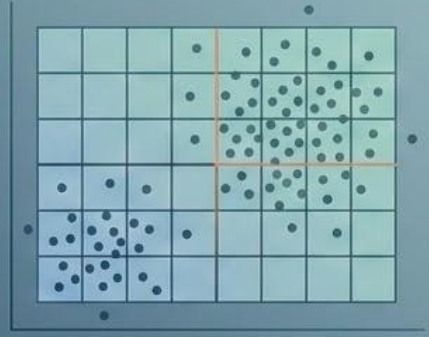


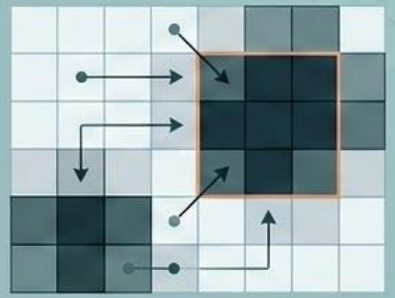
Smooth Topography:
Provides a highly useful continuous density estimate rather than a binary "in or out" classification.

Efficiency:
Can utilize mathematical approximations of the kernel function to execute efficiently even on larger datasets.

Grid-Based Clustering

Quantizing Space for Speed and Scale





The Core Idea

Instead of evaluating every single data point individually, we divide the entire data space into a finite number of geometric grid cells.

The Shift in Focus

We shift our clustering logic from points to cells. We calculate the density of each cell and form clusters based on neighboring dense cells.

STING: Statistical Information Grid

A Hierarchical Approach to Grids

Structure

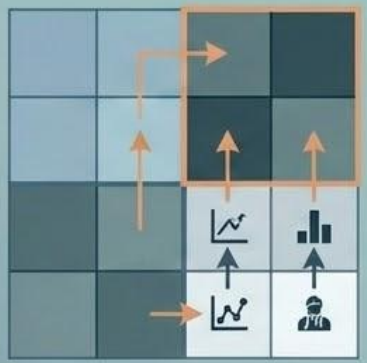
- Organizes the data space into a hierarchical grid structure (like a quadtree).

Granularity

- Features multiple levels of resolution. A high-level cell is divided into several smaller, lower-level cells.

Precomputation is Key

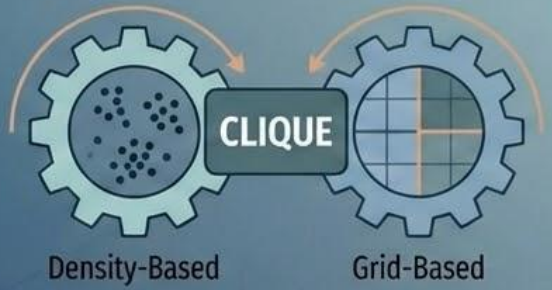
- The system precomputes and stores statistical parameters (like mean, variance, count) for each cell at the lowest level, rolling them up to higher levels.



CLIQUE: Tackling High Dimensions

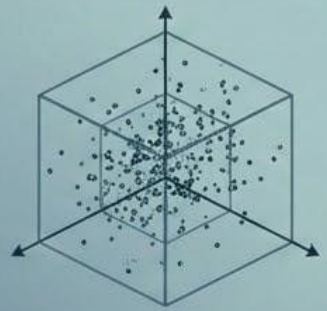
Combining Density and Grids

The Hybrid Approach



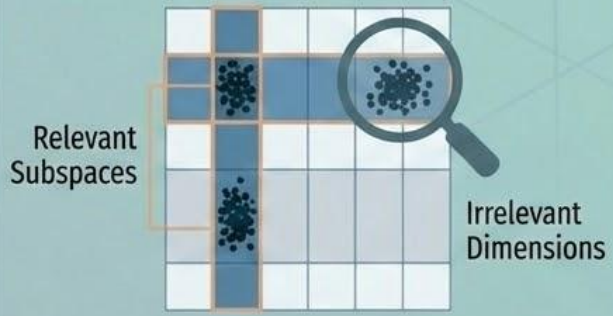
CLIQUE seamlessly integrates density-based clustering with a grid-based framework.

The High-Dimensional Solution



Designed specifically to find clusters in very high-dimensional data, where traditional methods fail.

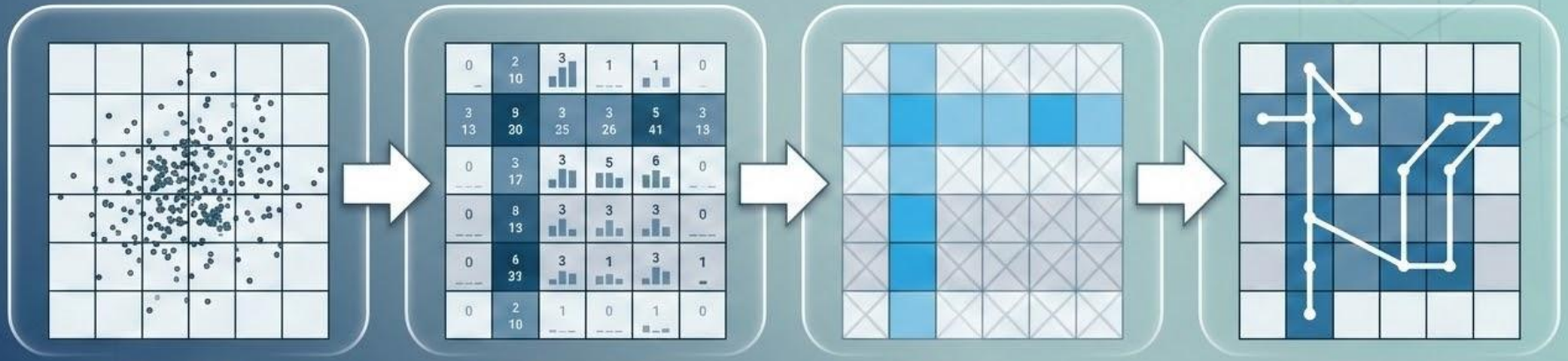
Subspace Discovery



It automatically identifies the specific subspaces (combinations of specific dimensions) where dense clusters actually exist, ignoring irrelevant dimensions.

Grid-Based Algorithm Flow

Step-by-Step Execution



1. Partition

Divide the entire data space into a finite number of grid cells.

2. Compute

Calculate the density (point count or statistical weight) for every single cell.

3. Filter

Remove all cells that fall below the predefined density threshold (eliminating the noise).

4. Cluster

Form the final clusters by connecting adjacent, contiguous dense cells.

Evaluating Grid-Based Methods

The Trade-Offs of the Grid

Advantages

- **Incredibly Fast:** Processing time is typically $O(n)$, dependent only on the number of cells, not the number of points.
- **Distribution Independent:** Does not assume data follows a specific statistical distribution.
- **Handles Dimensions:** Algorithms like CLIQUE naturally handle high-dimensional spaces by isolating subspaces.

Disadvantages

- **Loss of Precision:** Hard boundaries can split clusters or combine distinct groups that happen to share a cell boundary.
- **Curse of Dimensionality:** As dimensions increase, the number of required grid cells grows exponentially, severely impacting performance.

Example 8.8. Clustering requires nonuniform distribution of data. Fig. 8.23 shows a data set that is uniformly distributed in 2-D data space. Although a clustering algorithm may still artificially partition

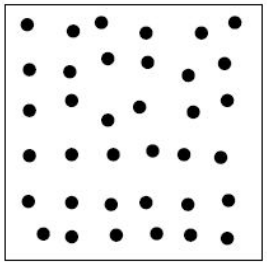


FIGURE 8.23

A data set that is uniformly distributed in the data space.

the points into groups, the groups will unlikely mean anything significant to the application due to the uniform distribution of the data. □

Assessing Clustering Tendency

Are There Actually Clusters Here?

The Fundamental Question:

Before we even apply an algorithm like DBSCAN or k-means, we must ask: Does this dataset actually contain any natural clusters at all?

The Risk of Blind Application:

Clustering algorithms will almost always group data if you force them to—even if the data is completely random and uniform.

The Solution:

We must evaluate the data's "clustering tendency" before processing it to avoid finding meaningless, artificial patterns.

Visual Assessment of Cluster Tendency (VAT)

Seeing the Patterns

The Concept

VAT is a visual technique used to determine if clustering is viable.

The Interpretation

Dark, distinct blocks along the diagonal of the image strongly indicate the presence of potential clusters.

The Process

- Compute a complete dissimilarity matrix (the distances between all pairs of objects).
- Reorder the matrix so that similar objects are grouped close to one another.
- Visualize the reordered matrix as an image (typically using grayscale).

The Hopkins Statistic

Mathematically Proving Clusterability

The Formula

$$H = \frac{\sum_{i=1}^{m} u_i}{\sum_{i=1}^{m} u_i + \sum_{i=1}^{m} w_i}$$

If we need a rigorous mathematical proof rather than a visual check, we use the Hopkins Statistic.

The Variables

- u_i: The distances from artificially generated random points to their nearest neighbor in the actual dataset.
- w_i: The distances from actual data points to their own nearest neighbor in the dataset.

Example 8.9. Hopkins statistic. Consider a 1-D data set $D = \{0.9, 1, 1.3, 1.4, 1.5, 1.8, 2, 2.1, 4.1, 7, 7.4, 7.5, 7.7, 7.8, 7.9, 8.1\}$ in the data space $[0, 10]$. We draw a sample of four points from D without replacement, say, 1.3, 1.8, 7.5, and 7.9. We also draw a sample of four points uniformly from the data space $[0, 10]$, say, 1.9, 4, 6, 8. Then, the Hopkins statistic can be calculated as

$$H = \frac{|1.9-2| + |4-4.1| + |6-7| + |8-8.1|}{(|1.9-2| + |4-4.1| + |6-7| + |8-8.1|) + (|1.3-1.4| + |1.8-2| + |7.5-7.4| + |7.9-7.8|)}$$
$$= \frac{1.3}{1.3+0.5} = \frac{1.3}{1.8} = 0.72.$$

Since the Hopkins statistic is substantially larger than 0.5 and is close to 1, the data set D has a strong clustering tendency. Indeed, there are two clusters, one around 1.5 and the other one around 7.8. □

Reading the Result (H)

What Does the Number Mean?

Once calculated, the Hopkins Statistic (H) provides a clear threshold for action:

Value of H	Interpretation	Action
$H \approx 0.5$	The data is uniformly distributed (random). Real distances and artificial distances are roughly the same.	❌ Stop. Do not apply clustering algorithms; results will be meaningless.
$H \to 1$	The data is highly clustered. The real data points are much closer to each other than the random artificial points are.	✅ Proceed. The data contains distinct, natural groupings ready for analysis.

The ‘K’ Problem

Determining the Number of Clusters

The Challenge

Algorithms like k-means require you to input the number of clusters (K) beforehand. But in **unsupervised learning**, we don’t know the ground truth.

The Goal

We need objective, mathematical methods to evaluate our data and determine the truly optimal number of clusters.

The Three Methods:

- The Elbow Method
- The Silhouette Method
- The Gap Statistic

The Elbow Method

Finding the Point of Diminishing Returns

The Metric:

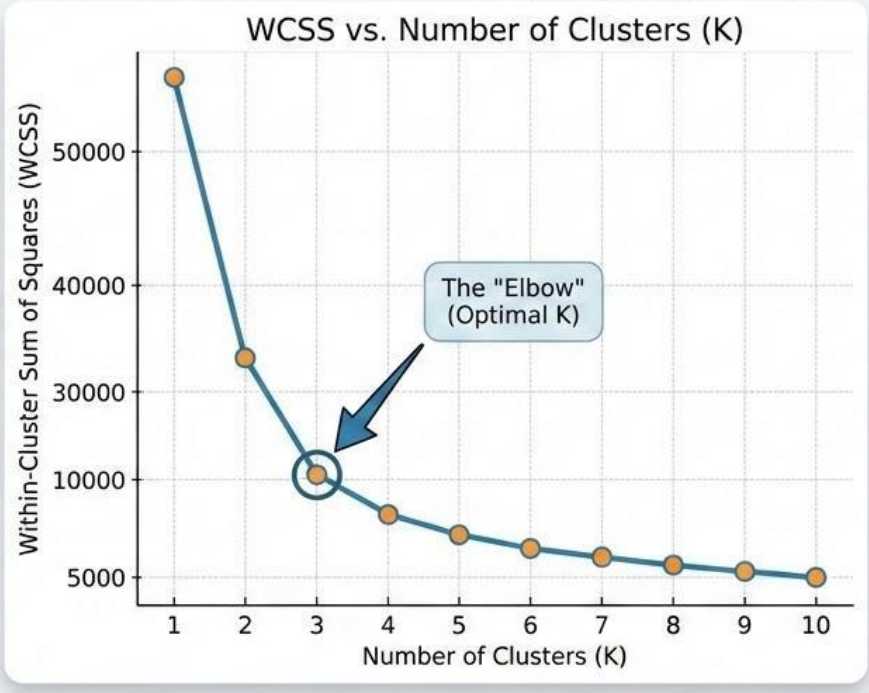
We compute the Within-Cluster Sum of Squares (WCSS). This measures how tightly packed the data points are within their assigned clusters.

The Process:

1. Run the clustering algorithm for increasing values of K (e.g., 1 through 10).
2. Plot the resulting WCSS against the number of clusters (K).

The Interpretation:

Look for the "elbow" in the graph—the specific point where the rate of improvement sharply diminishes.



The Silhouette Method

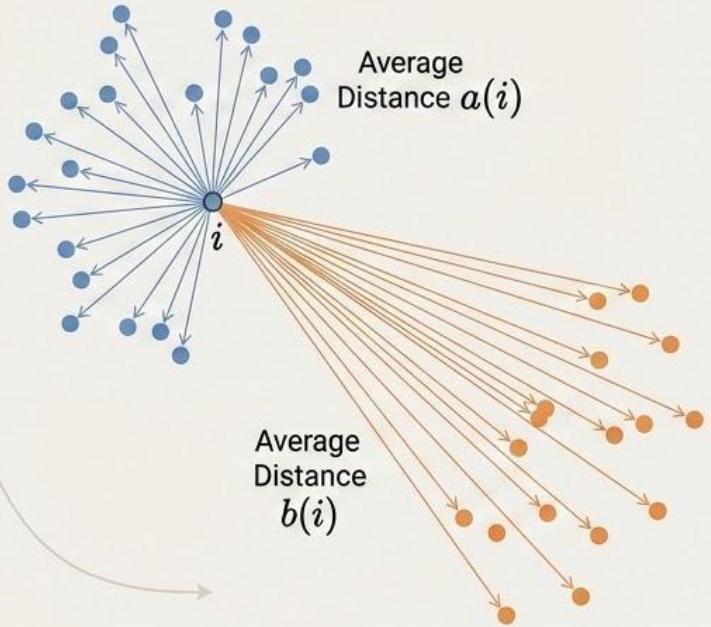
Balancing Cohesion and Separation

The Concept

Evaluates how similar an object is to its own cluster compared to other clusters.

The Formula

$$s(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))}$$



The Variables

$a(i)$ (Cohesion)

The average distance from point i to all other points within its same cluster. (Smaller is better)

$b(i)$ (Separation)

The average distance from point i to all points in the nearest other cluster. (Larger is better)

Reading the Silhouette

What Does $s(i)$ Tell Us?

Near 1

Well-Clustered.

The point is tightly grouped within its cluster and far away from neighboring clusters.

System Health

Excellent. The chosen K fits this data well.

Near 0

Ambiguous.

The point is situated right on the border between two overlapping clusters.

System Health

Warning. Clusters may be poorly defined.

Near −1

Misclassified.

The point is mathematically closer to a neighboring cluster than to its own assigned cluster.

System Health

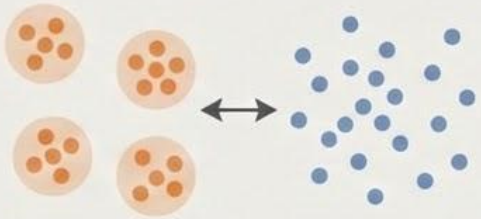
Failure. The point has been placed in the wrong group.

The Gap Statistic

Comparing Against Randomness

The Core Idea

Compares the actual within-cluster variation to what we would expect if the data had absolutely no clusters (a random uniform distribution).



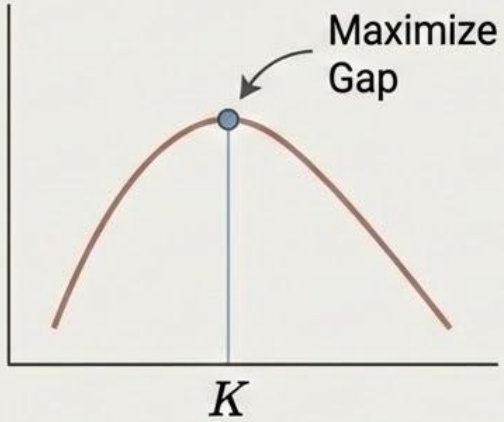
The Calculation

- Generate a random "null" reference dataset.
- Compare the log of the variation:

$$Gap(K) = \text{expected\_log}(W_K) - \text{actual\_log}(W_K)$$

The Conclusion

Choose the value of K that maximizes the gap.

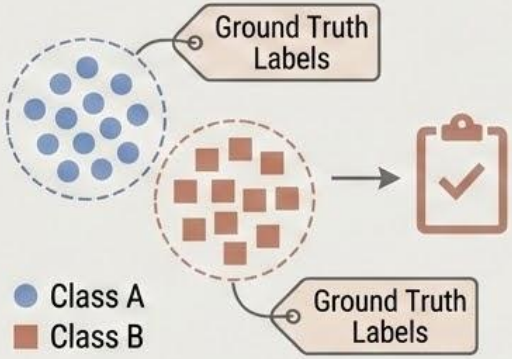


Measuring Clustering Quality: Extrinsic Methods

Evaluating with Ground Truth Data

When to Use

Extrinsic methods are utilized strictly when ground truth labels are already available for your dataset.



Metric 1: Purity

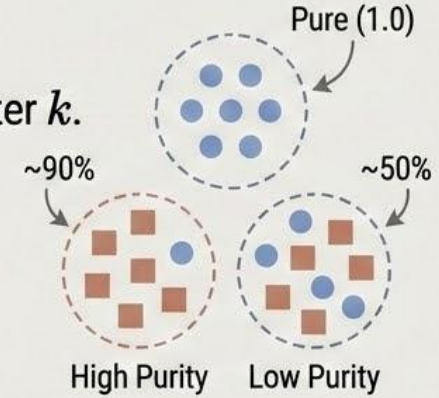
Measures the extent to which clusters contain a single class.

$$Purity = \frac{1}{n} \sum_{k} \max_{j} |C_k \cap T_j|$$

The Variables:

- n: Total number of data points.
- C_k: The set of points in assigned cluster k.
- T_j: The set of points in true class j.

Interpretation: Higher is better. A score of 1.0 indicates perfect clustering where every cluster contains only points from a single true class.



Example 8.10. Purity. Consider the set of objects $D = \{a, b, c, d, e, f, g, h, i, j, k\}$. The clustering ground truth and two clusterings $\mathcal{C}_1$ and $\mathcal{C}_2$ output by two methods are shown in Table 8.1.

The purity of clustering $\mathcal{C}_1$ is calculated by $\frac{1}{11} \times (4+2+4+1) = \frac{11}{11} = 1$ and that of clustering $\mathcal{C}_2$ is $\frac{1}{11}(2+3+1) = \frac{6}{11}$. In terms of purity, $\mathcal{C}_1$ is better than $\mathcal{C}_2$. Please note that, although $\mathcal{C}_1$ has purity 1, it splits G_1 in the ground truth into two clusters, C_1 and C_2. □

There are some other matching based methods further refine the measurement of matching quality, such as maximum matching and using F-measure.

Table 8.1 A set of objects, the clustering ground truth, and two clusterings.

Object	a	b	c	d	e	f	g	h	i	j	k
Ground truth $\mathcal{G}$	G_1	G_1	G_1	G_1	G_1	G_1	G_2	G_2	G_2	G_2	G_3
Clustering $\mathcal{C}_1$	C_1	C_1	C_1	C_1	C_2	C_2	C_3	C_3	C_3	C_3	C_4
Clustering $\mathcal{C}_2$	C_1	C_1	C_2	C_2	C_2	C_3	C_1	C_2	C_2	C_1	C_3

The Rand Index (RI)

Measuring Pairwise Agreements

The Concept

- The Rand Index views clustering as a series of pairwise decisions. It calculates the percentage of decisions the algorithm got right.

The Formula

$$RI = \frac{TP + TN}{TP + FP + TN + FN}$$

The Decision Matrix

TP (True Positive)

Points belong to the same class and were placed in the same cluster.

TN (True Negative)

Points belong to different classes and were placed in different clusters.

FP (False Positive)

Points belong to different classes but were grouped in the same cluster.

FN (False Negative)

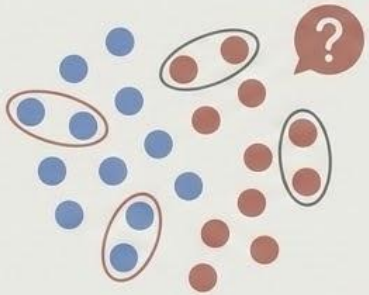
Points belong to the same class but were split into different clusters.

Adjusted Rand Index (ARI)

Correcting for Random Chance

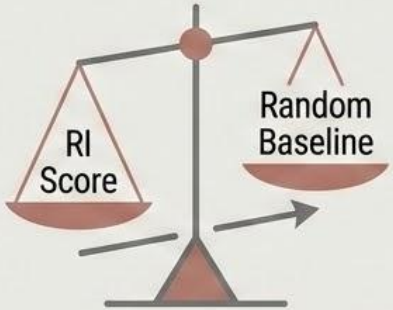
The Limitation of RI

The standard Rand Index does not account for the fact that a completely random clustering algorithm would still get some pairs right by pure chance.



The Solution

The Adjusted Rand Index (ARI) mathematically corrects the score by establishing a baseline for random assignments.



Interpreting the Score (Range: -1 to 1)

Negative Score

The clustering is somehow performing worse than random chance.

0.0

The clustering is exactly what you would expect from completely random guessing.

1.0

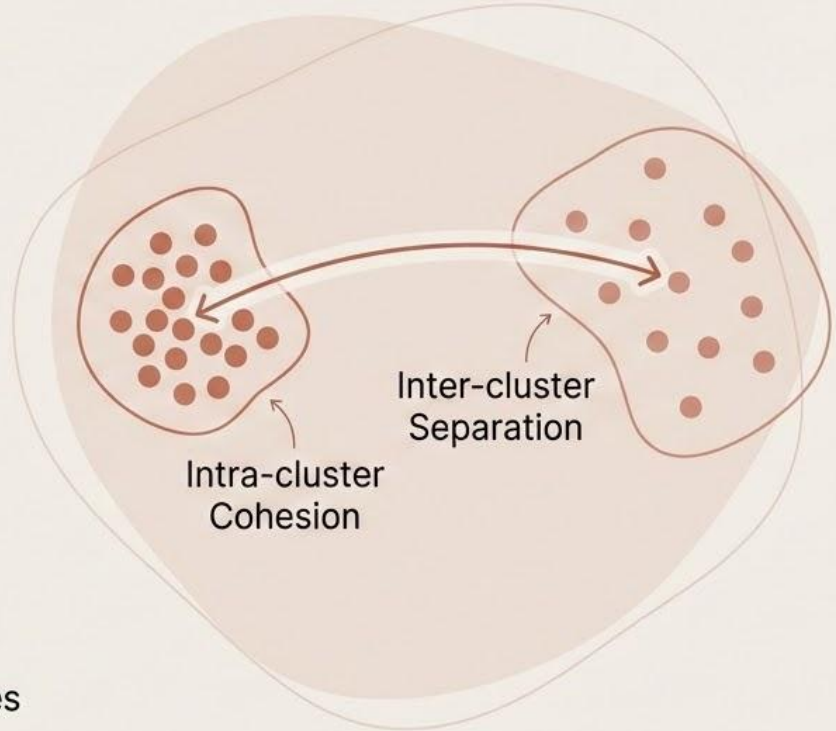
Perfect agreement with the ground truth.

Measuring Clustering Quality: Intrinsic Methods

Assessing Internal Structure

When to Use: Intrinsic methods are essential when no ground truth labels are available (which is the reality for most unsupervised learning tasks).

The Goal: Evaluate the clustering based purely on the data itself—specifically looking for high intra-cluster cohesion and high inter-cluster separation.



Metric 1: The Silhouette Coefficient

- Measures how similar a point is to its own assigned cluster compared to neighboring clusters.
- Calculated as an average over all points in the dataset.
- Interpretation: Higher is better. (Approaching 1 indicates perfectly dense and separated clusters).

Davies-Bouldin Index (DBI)

Measuring Worst-Case Separation

The Concept

Evaluates the average "similarity" between each cluster and its most similar neighboring cluster.

$$DB = \frac{1}{k} \sum_{i=1}^{k} \max_{j \neq i} \left(\frac{\sigma_i + \sigma_j}{d(c_i, c_j)} \right)$$

The Variables

- σ_i: The average distance of all points within cluster i to their centroid (spread).
- $d(c_i, c_j)$: The actual distance between the centroids of cluster *i* and cluster *j*.

Interpretation

Lower is better.
A lower score indicates that the clusters are compact (low σ) and well-separated (high d).

Calinski-Harabasz Index

The Variance Ratio Criterion

The Concept

Compares the variance (dispersion) between all clusters to the variance within the clusters themselves.

$$CH = \frac{B/(k-1)}{W/(n-k)}$$

The Variables

- B: Between-cluster dispersion (how far the cluster centers are from the overall dataset center).
- W: Within-cluster dispersion (how scattered the points are within their own clusters).
- k: Number of clusters; n: Total number of points.

Interpretation

Higher is better.
A high score means clusters are spread far apart relative to how tightly packed they are internally.

Dunn Index

Maximizing the Minimum Distance

The Concept

Identifies the "weakest link" in your clustering geometry by looking at the absolute closest clusters and the absolute widest cluster.

$$D = \frac{\min_{i \neq j} d(C_i, C_j)}{\max_k \operatorname{diam}(C_k)}$$

The Variables

- $d(C_i, C_j)$: The distance between the two closest separate clusters (Inter-cluster distance).
- $\operatorname{diam}(C_k)$: The size of the absolutely largest/widest single cluster (Intra-cluster diameter).

Interpretation

Higher is better.
To get a high score, you must maximize inter-cluster distance and minimize intra-cluster distance.