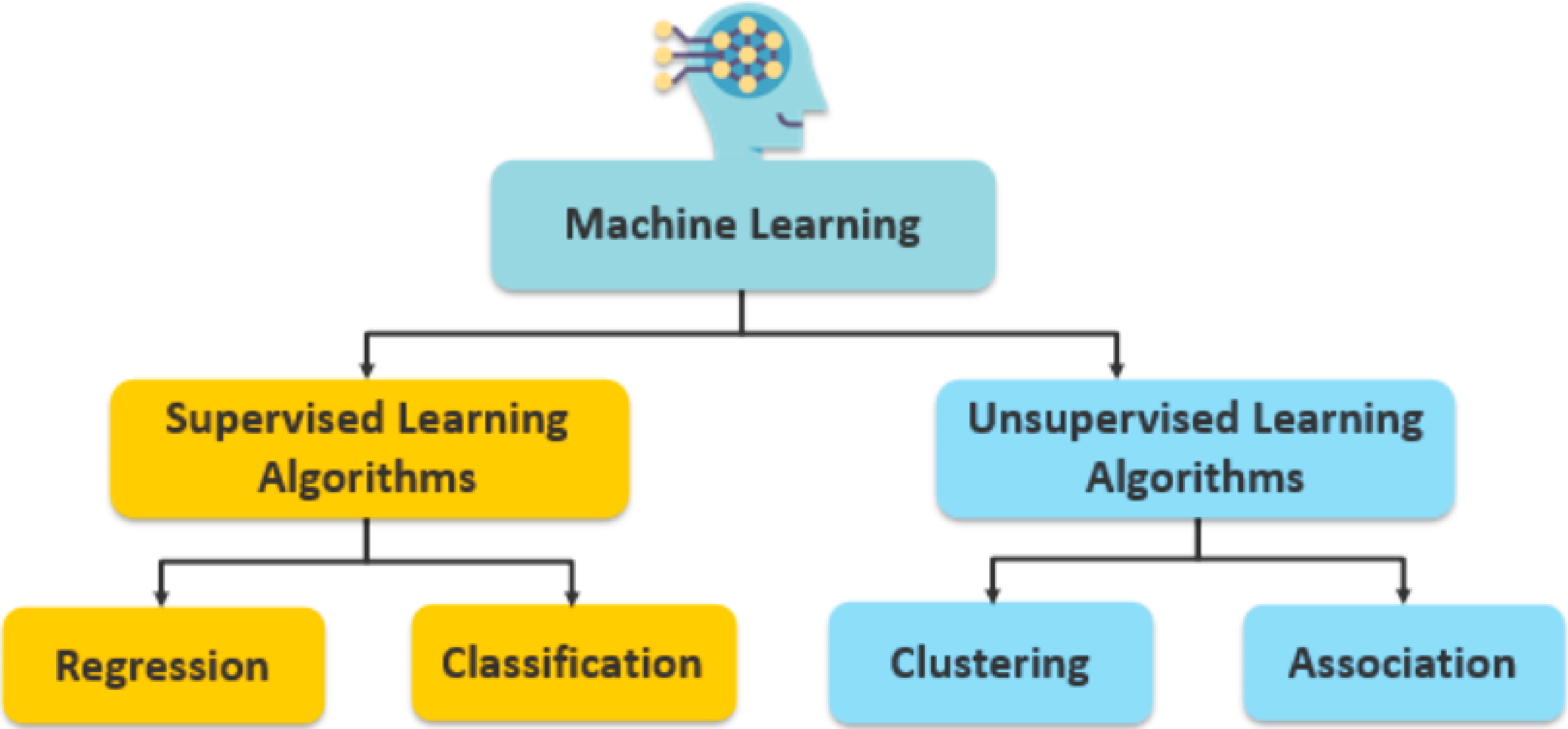
Advanced Clustering Techniques

Hierarchical Clustering and DBSCAN

Recap



Machine Learning
Supervised Learning Algorithms
Unsupervised Learning Algorithms
Regression
Classification
Clustering
Association

	Supervised Learning	Unsupervised learning
Objective	To approximate a function that maps inputs to outputs based out example input-output pairs.	To build a concise representation of the data and generate imaginative content from it.
Accuracy	Highly accurate and reliable.	Less accurate and reliable.
Complexity	Simpler method.	Computationally complex.
Classes	Number of classes is *known.*	Number of classes is *unknown.*
Output	A desired output value (also called the supervisory signal).	No corresponding output values.

Clustering

- What is Cluster Analysis?
- A Categorization of Major Clustering Methods
- Hierarchical Methods
- Density-Based Methods

What is Cluster Analysis?

- Cluster: a collection of data objects
  - Similar to one another within the same cluster
  - Dissimilar to the objects in other clusters
- Cluster analysis
  - Finding similarities between data according to the characteristics found in the data and grouping similar data objects into clusters
- Unsupervised learning: no predefined classes
- Typical applications
  - As a stand-alone tool to get insight into data distribution
  - As a preprocessing step for other algorithms

What is Cluster Analysis?

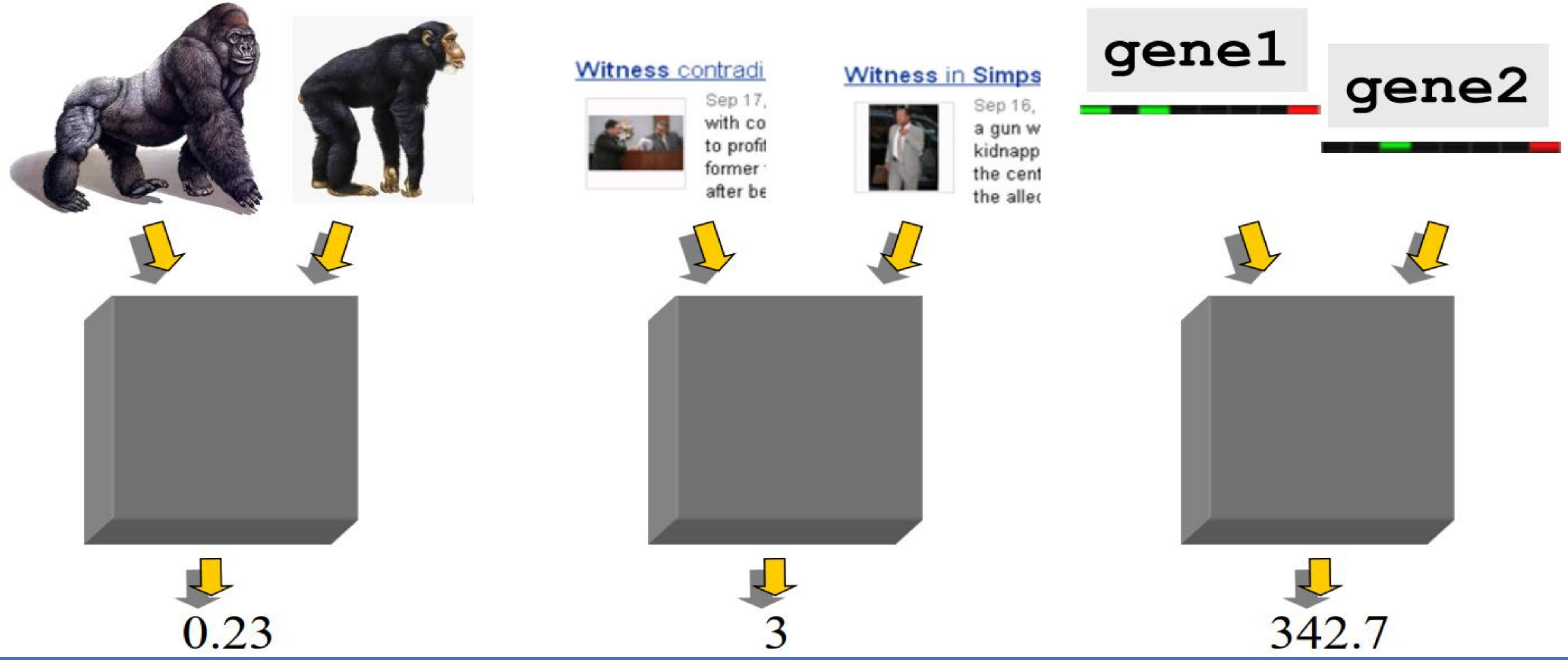
- The quality or state of being similar; likeness; resemblance; as, a similarity of features. **Webster's Dictionary**

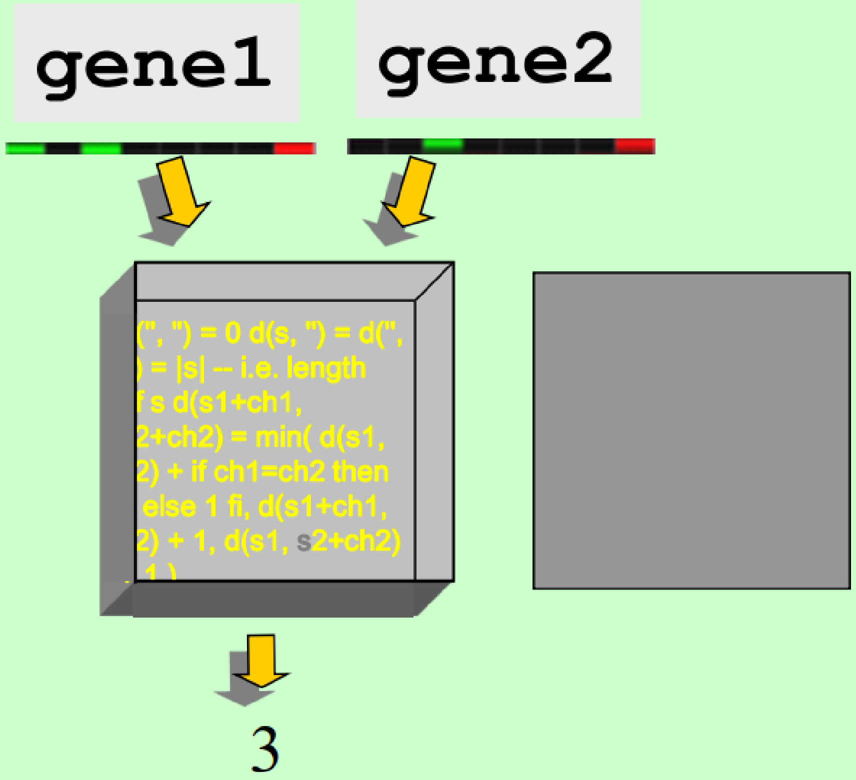
Similarity is hard to define, but…
"*We know it when we see it*"

The real meaning of similarity is a philosophical
question.

Defining Distance Measures

Definition: Let O_1 and O_2 be two objects from the universe of possible objects. The distance (dissimilarity) between O_1 and O_2 is a real number denoted by $D(O_1,O_2)$





Inside these black boxes:
some function on two variables
(might be simple or very
complex)

A few examples:

- Euclidian distance

$$d(x,y)=\sqrt{\sum_i (x_i-y_i)^2}$$

- Similarity rather than distance
- Can determine similar trends

- Correlation coefficient

$$s(x,y)=\frac{\sum_i (x_i-\mu_x)(y_i-\mu_y)}{\sigma_x\sigma_y}$$

General Applications of Clustering

- Pattern Recognition
- Spatial Data Analysis
  - Create thematic maps in GIS by clustering feature spaces
  - Detect spatial clusters or for other spatial mining tasks
- Image Processing
- Economic Science (especially market research)
- WWW
  - Document classification
  - Cluster Weblog data to discover groups of similar access patterns

Examples of Clustering Applications

- Marketing: Help marketers discover distinct groups in their customer bases, and then use this knowledge to develop targeted marketing programs
- Land use: Identification of areas of similar land use in an earth observation database
- Insurance: Identifying groups of motor insurance policy holders with a high average claim cost
- City-planning: Identifying groups of houses according to their house type, value, and geographical location
- Earth-quake studies: Observed earth quake epicenters should be clustered along

What Is Good Clustering?

- A good clustering method will produce high quality clusters with
  - high intra-class similarity
  - low inter-class similarity
- The quality of a clustering result depends on both the similarity measure used by the method and its implementation
- The quality of a clustering method is also measured by its ability to discover some or all of the hidden patterns

Measure the Quality of Clustering

- Dissimilarity/Similarity metric: Similarity is expressed in terms of a distance function, typically metric: $d(i, j)$
- There is a separate "quality" function that measures the "goodness" of a cluster.
- The definitions of distance functions are usually very different for interval-scaled, boolean, categorical, ordinal ratio, and vector variables.
- Weights should be associated with different variables based on applications and data semantics.
- It is hard to define "similar enough" or "good enough"
    - the answer is typically highly subjective.

Desirable Properties of a Clustering Algorithm

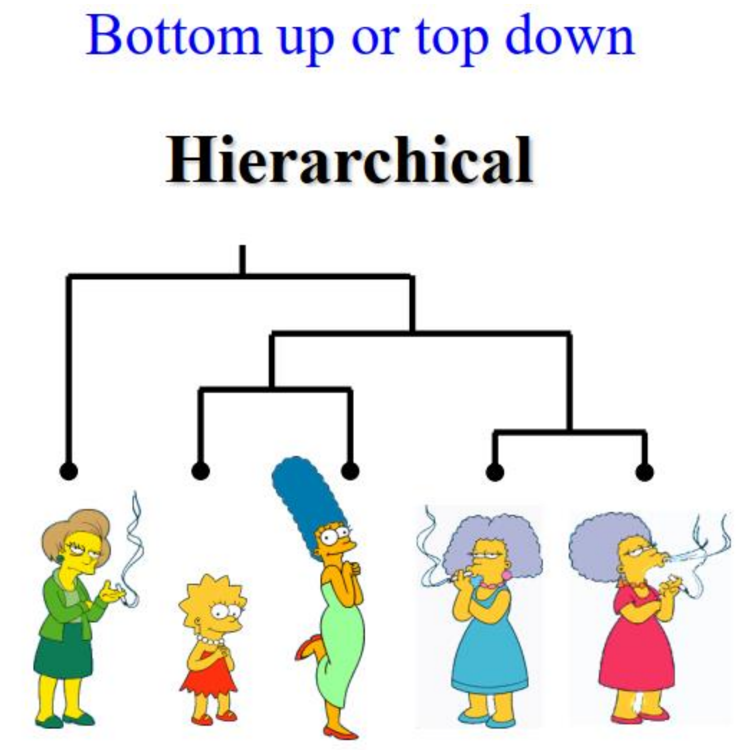
- Scalability (in terms of both time and space)
- Ability to deal with different data types
- Minimal requirements for domain knowledge to determine input parameters
- Interpretability and usability

Optional

- Incorporation of user-specified constraints

Types of Clustering Algorithm

- **Partitional algorithms:** Construct various partitions and then evaluate them by some criterion
- **Hierarchical algorithms:** Create a hierarchical decomposition of the set of objects using some criterion (focus of this class)



Hierarchical Clustering

- A hierarchical clustering method works by grouping objects into a tree of clusters.
- It does not require the number of clusters to be predefined.

Types:

- Agglomerative (Bottom-Up): Starts with each data point as its own cluster, and pairs of clusters are merged as you move up the hierarchy.
- Divisive (Top-Down): Starts with all data points in one cluster and splits them recursively until all points are individual clusters.

Key Concepts in Hierarchical Clustering

Distance Metrics:

- Euclidean Distance: The straight-line distance between two points in Euclidean space.
- Manhattan Distance: The sum of the absolute differences of the coordinates of two points.
- Cosine Similarity: Measures the cosine of the angle between two vectors (used in text mining).
- Jaccard Similarity: Used for comparing the similarity of sample sets.

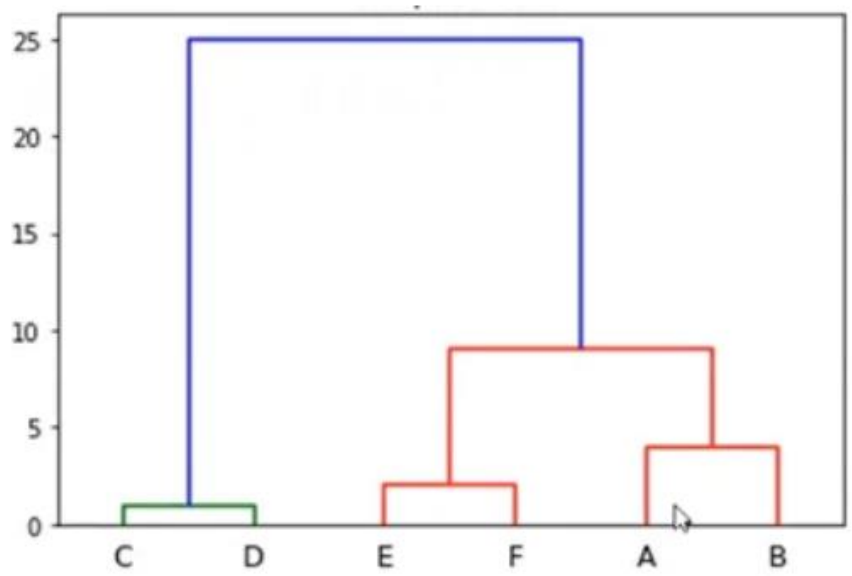
Linkage Methods:

- Single Linkage: Measures the shortest distance between points in different clusters.
- Complete Linkage: Measures the farthest distance between points in different clusters.
- Average Linkage: Considers the average distance between all points in the clusters.
- Ward's Method: Minimizes the total within-cluster variance, leading to compact clusters.

Understanding the Dendrogram

Dendrogram Overview:

- A tree-like diagram used to represent the results of hierarchical clustering.
- Shows how clusters are merged or divided at each level, and the height at which clusters are merged indicates their distance.



Interpreting the Dendrogram:

- The x-axis represents the data points, and the y-axis represents the distance at which clusters are merged.
- Cutting the dendrogram at a particular height determines the number of clusters.
- A lower cut will result in fewer clusters, while a higher cut will produce more clusters.

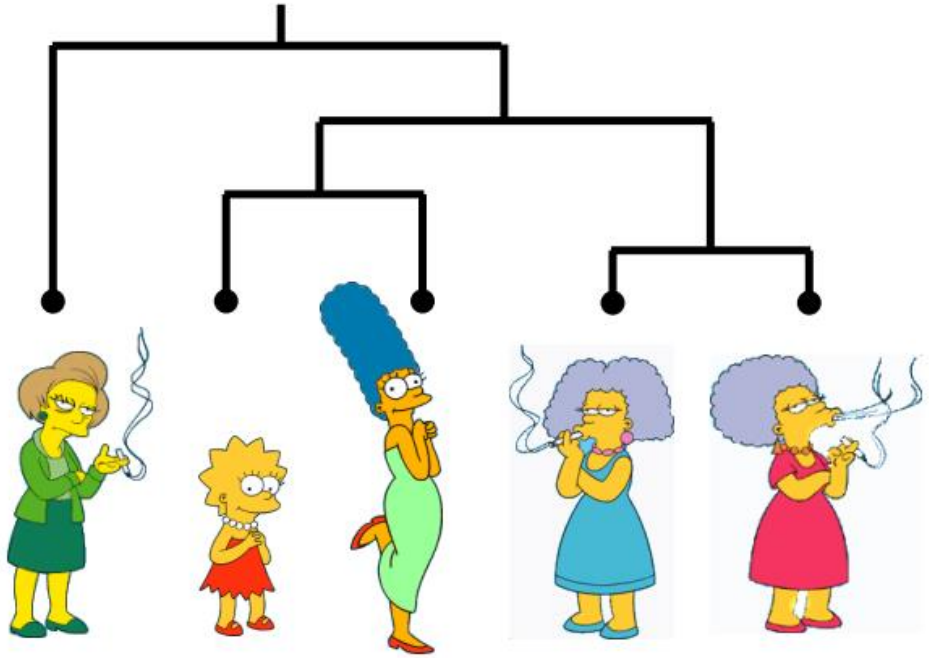
Hierarchical Clustering: Agglomerative

- This bottom-up strategy starts by placing each object in its own cluster and then merges these atomic clusters into larger and larger clusters, until all of the objects are in a single cluster or until certain termination conditions are satisfied.

- **Method:**
- Start with partition Pn, where each object forms its own cluster.
- Merge the two closest clusters, obtaining Pn-1.
- Repeat merge until only one cluster is left or termination condition is satisfied.

Hierarchical Clustering: Agglomerative

- This bottom-up strategy starts by placing each object in its own cluster and then merges these atomic clusters into larger and larger clusters, until all of the objects are in a single cluster or until certain termination conditions are satisfied.



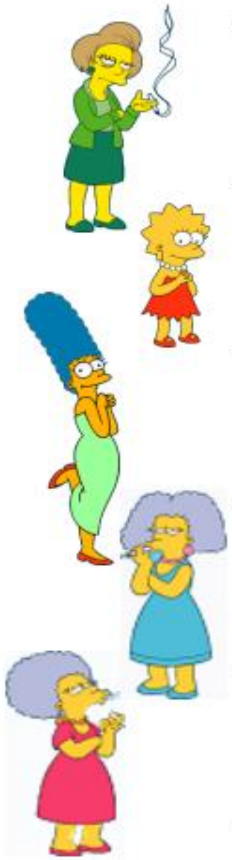
Bottom-Up (agglomerative): Starting with each item in its own cluster, find the best pair to merge into a new cluster. Repeat until all clusters are fused together.

Hierarchical Clustering: Agglomerative

We begin with a distance matrix which contains the distances between every pair of objects in our database.

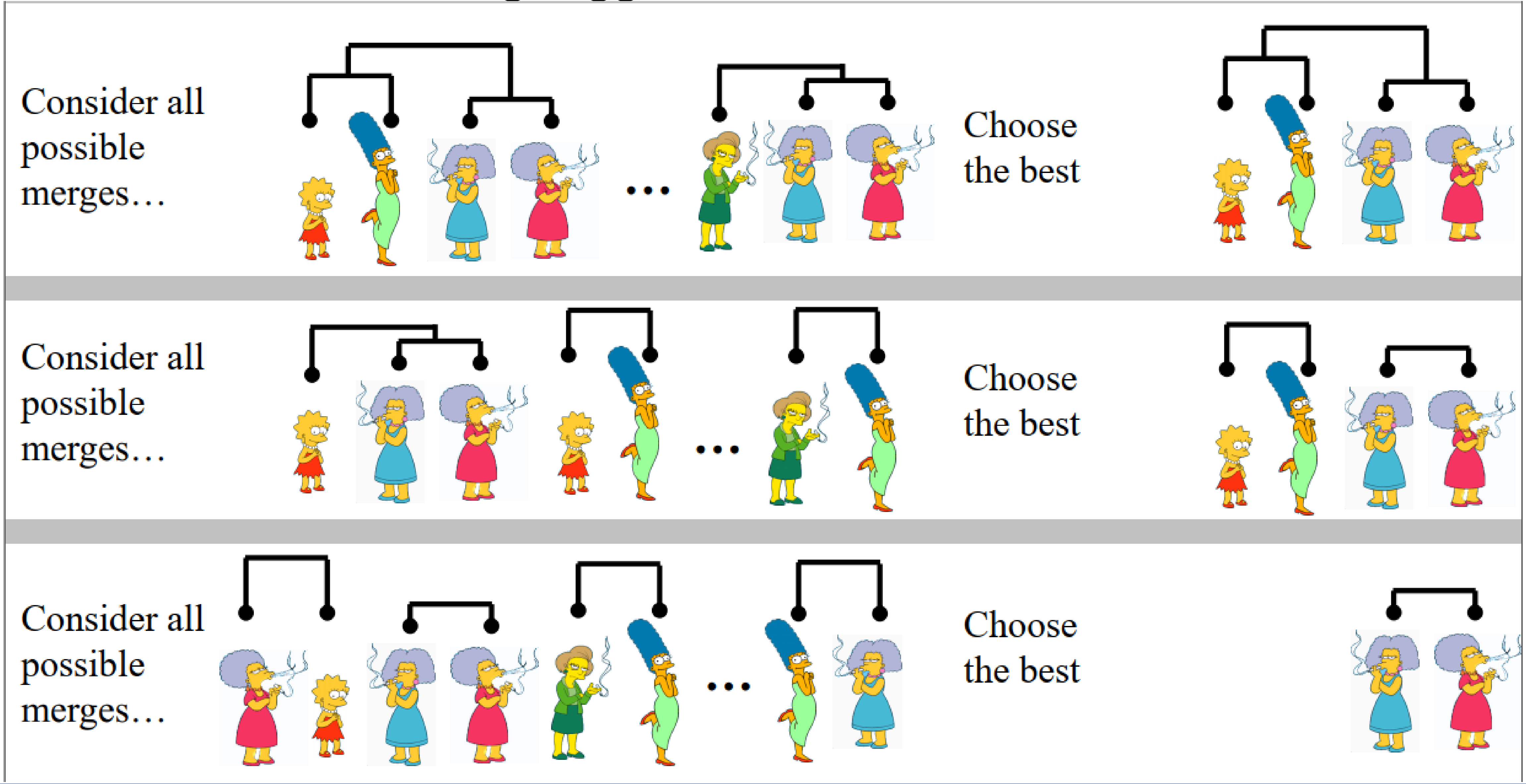
$D(\;,\;) = 8$

$D(\;,\;) = 1$



0	8	8	7	7
	0	2	4	4
		0	3	3
			0	1
				0

Hierarchical Clustering: Agglomerative

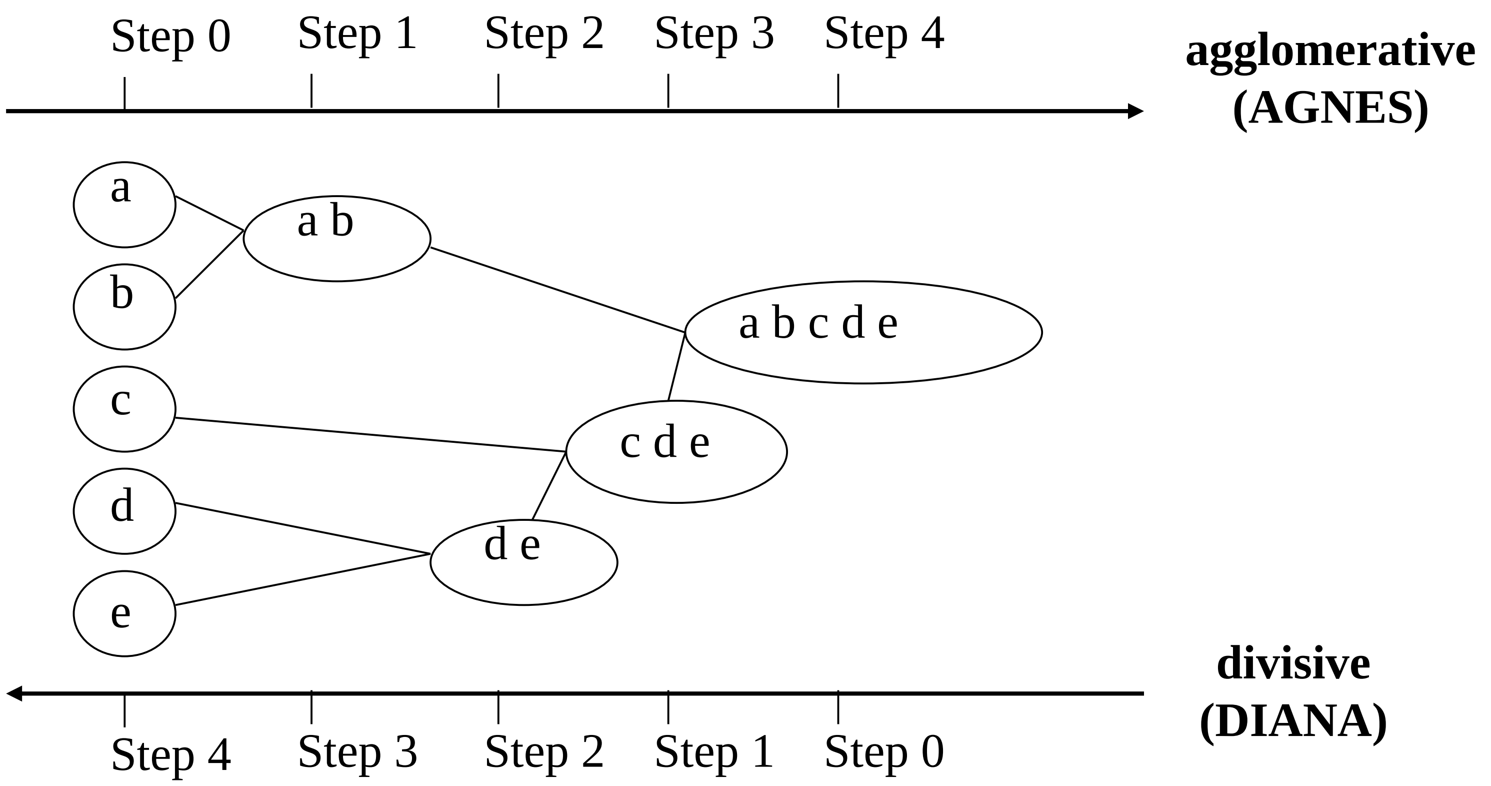


Hierarchical Clustering: Divisive (DIANA)

- This top-down strategy does the reverse of agglomerative hierarchical clustering by starting with all objects in one cluster. It subdivides the clusters into smaller and smaller pieces, until each object form a cluster on its own or until it satisfies certain termination conditions, such as a desired number of cluster or the diameter of each cluster is within a certain threshold.
- **Method**:
- Start with P_1.
- Split the collection into two clusters that are as homogenous (and as different from each other) as possible.
- Apply splitting procedure recursively to the clusters.

Hierarchical Clustering

- Example: A data-set has five objects {a,b,c,d,e}
- AGNES (Agglomerative Nesting)
- DIANA (Divisive Analysis)



Strengths and Limitations Hierarchical Clustering

Strengths of hierarchical clustering:

- No need to specify the number of clusters upfront
- Produces a hierarchical structure that can be useful for understanding data relationships
- Results are easily visualizable
- Works well for many data types

Limitations of hierarchical clustering:

- Can be slow on large datasets
- Sensitive to noise and outliers

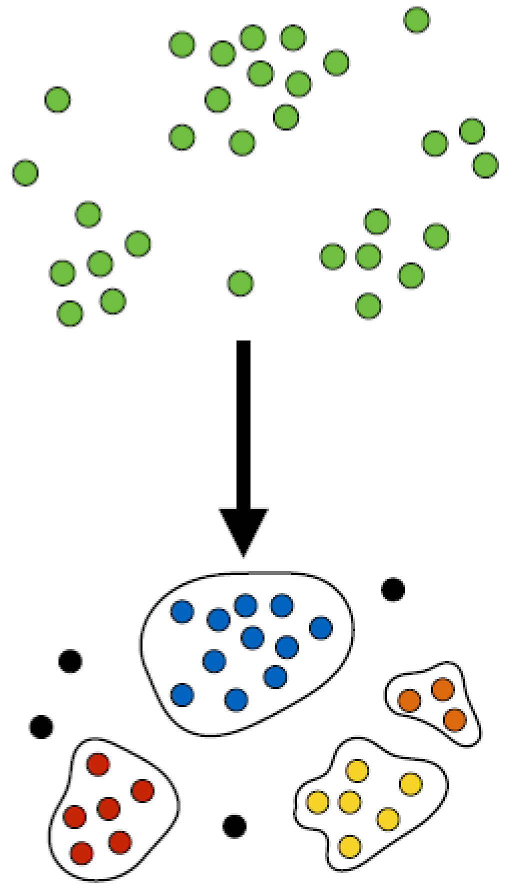
Density-Based Clustering Methods

- Clustering based on density (local cluster criterion), such as density-connected points
- Major features:
  - Discover clusters of arbitrary shape
  - Handle noise
  - One scan
  - Need density parameters as termination condition
- Density-Based algorithms:
  - DBSCAN
  - OPTICS
  - DENCLUE
  - CLIQUE

Concepts of Density based clustering

- Relies on a density-based notion of clusters
- Discovers clusters of arbitrary shape in spatial databases with noise

- **Basic Idea**
  - Group together points in high-density
  - Mark as outliers points that lie alone in low-density regions



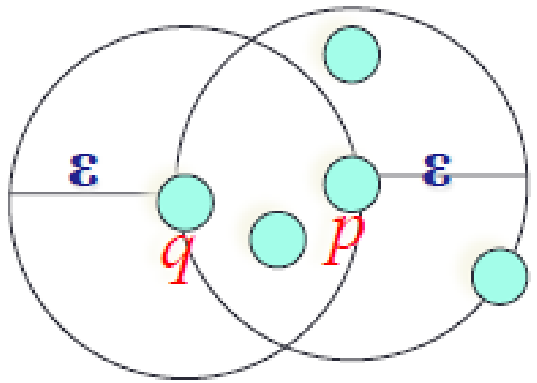
Density-Reachability

- **Neighborhood**: It is determined by a distance function (e.g., Manhattan Distance, Euclidean Distance) for two points p and q, denoted by dist(p,q).

- **Eps-neighborhood**: The Eps-neighborhood of a point p is defined by:

$$\{q \in D \,|\, \mathrm{dist}(p,q) \leqslant \mathrm{Eps}\}.$$

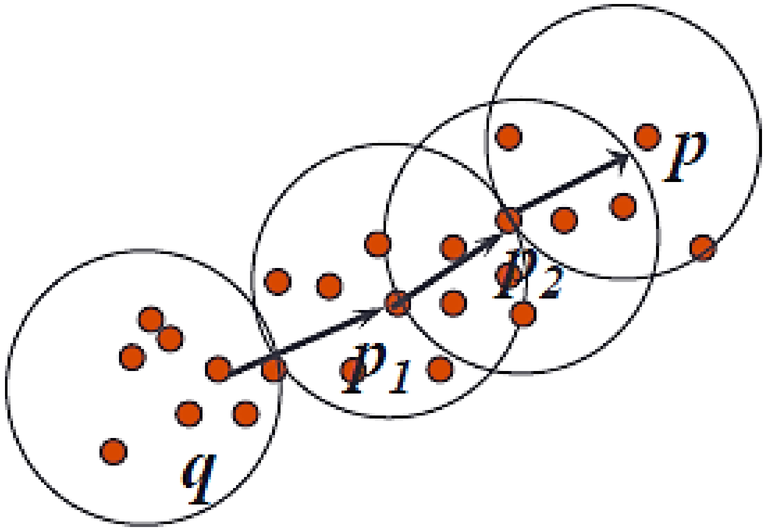
- **Directly density-reachable**: An object q is directly density-reachable from the object p if q is within the Eps-neighborhood of p, and p is a core object



Minpts = 4

Density-Reachability

- **Density-reachable** : A point p is density-reachable from the object q with respect to Eps and MinPts if there is a chain of objects $p_1, \ldots, p_n$, $p_1 = q$ and $p_n = p$ such that p_{i+1} is directly density-reachable from p_i with respect to Eps and MinPts

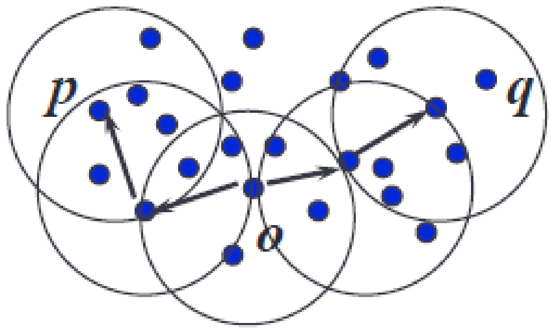


MinPts = 7

- p1 is directly density-reachable from q
- p2 is directly density-reachable from p1
- p is directly density-reachable from p2
- There is a chain from q to p (q→p1→p2→q)

Density-Connectivity

- **Density-connected**. A pair of points p and q are density-connected if they are commonly density-reachable from a point o.

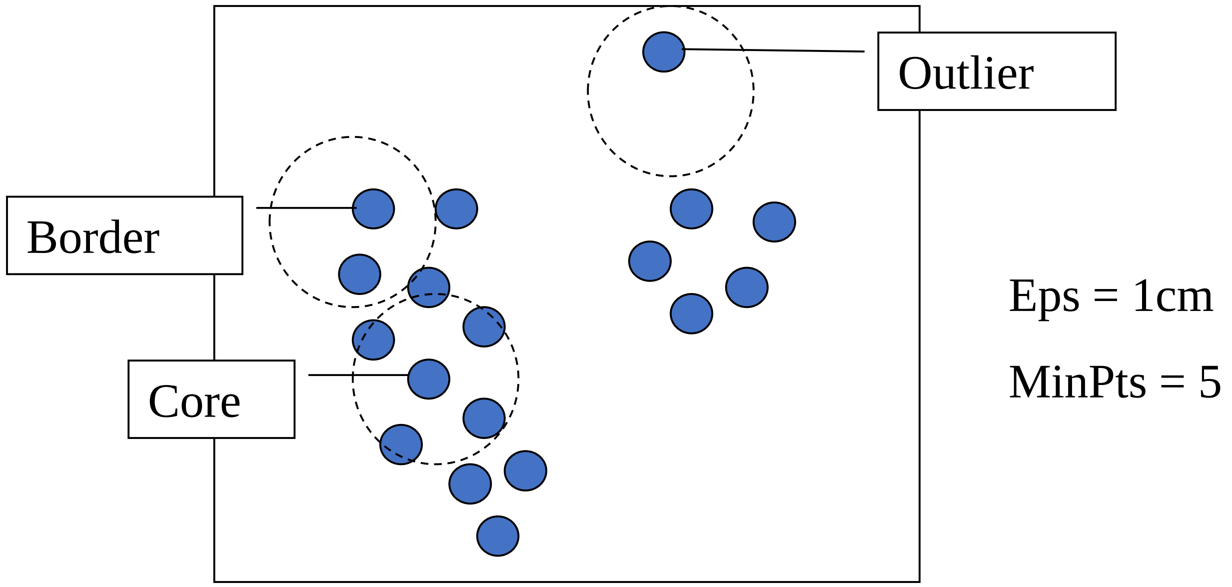


MinPts = 7

- **Density-based cluster**: A cluster C is a subset of D satisfying two criteria
  - Maximality
    - ∀ p, q if $p \in$ **C** and if q is density-reachable from p, then also $q \in$ **C**
  - Connectivity
    - ∀ p, $q \in$ **C**, p and q are density-connected

DBSCAN: Density Based Spatial Clustering of Applications with Noise

- Relies on a *density-based* notion of cluster: A *cluster* is defined as a maximal set of density-connected points
- Discovers clusters of arbitrary shape in spatial databases with noise



DBSCAN Parameters

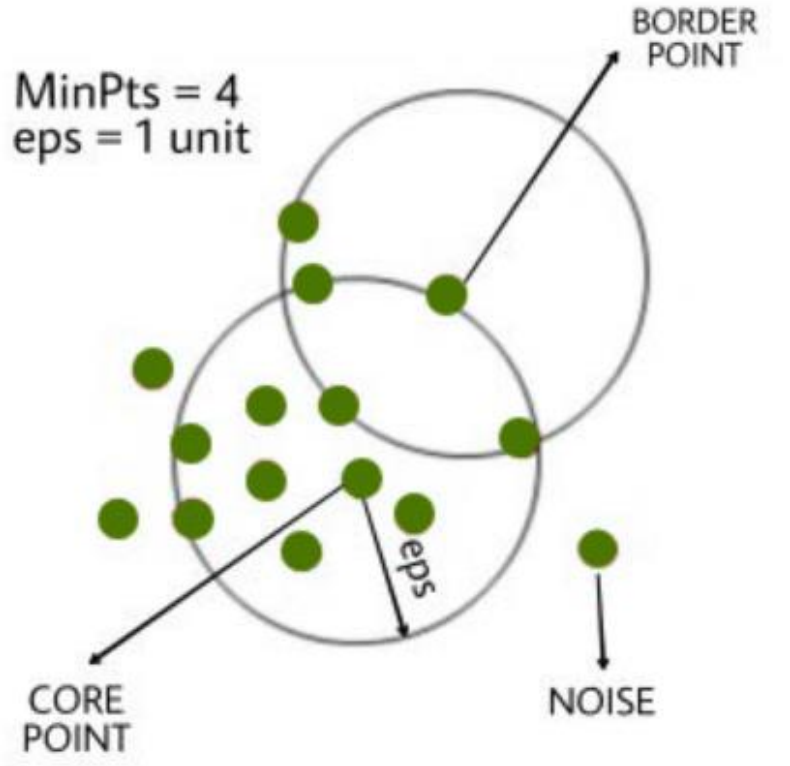
Epsilon (ε):

- The maximum distance between two points to be considered as neighbours.
- Too large a value results in one large cluster; too small may result in too many small clusters.

MinPts:

- Minimum number of points required to form a cluster.

- **Core Points, Border Points, and Noise**:
- Core Points: Points with at least MinPts points within eps distance.
- Border Points: Points that are within eps of a core point but don't have enough neighbors to be core points.
- Noise Points: Points that are neither core nor border points.



DBSCAN: The Algorithm

- Arbitrarily select a point ***p***
- Retrieve all points density-reachable from ***p*** wrt ***Eps*** and ***MinPts***.
- If ***p*** is a core point, a cluster is formed.
- If ***p*** is a border point, no points are density-reachable from ***p*** and DBSCAN visits the next point of the database.
- Continue the process until all of the points have been processed.

Comparing K-Means, DBSCAN, and Hierarchical Clustering

- **K-Means**:
  - Best for large datasets, simple and fast.
  - Assumes clusters are spherical and equally sized.
- **DBSCAN**:
  - Great for datasets with irregular shapes and noise.
- **Hierarchical**:
  - Produces a dendrogram that helps visualize clusters at different levels.
  - Suitable for small to medium-sized datasets but can be computationally expensive.