CS430/910: Foundations of Data Analytics

Clustering | Dr Greg Watson

Part A: Introduction to Clustering

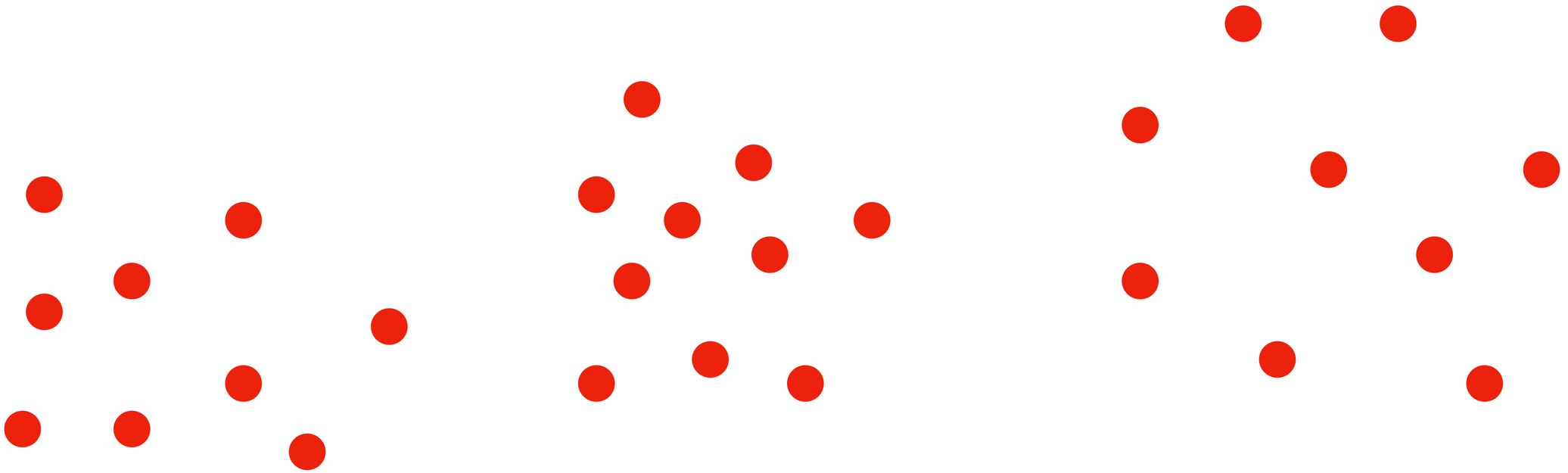
Objectives

- To understand the concept of clustering and clustering methods
- To see different clustering methods and objectives:
  - k-**means**
  - k-**medoids**
  - **Hierarchical Agglomerative Clustering**
  - **DBSCAN**
- To use tools (Weka) to cluster data and study the results

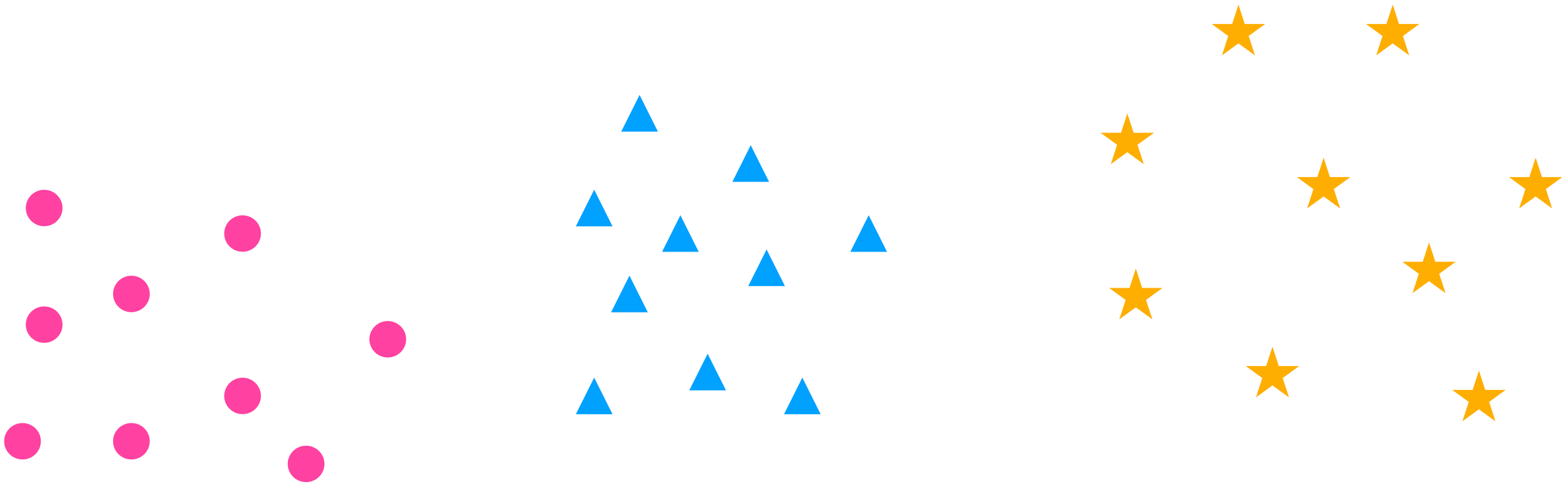
Clustering

- **Classification** learns classes based on **labeled** training data (supervised learning)
- **Clustering** aims to find classes **without labeled** examples
  - An “**unsupervised**” learning method
  - Place similar items in same group, different items in different groups

What is Clustering?

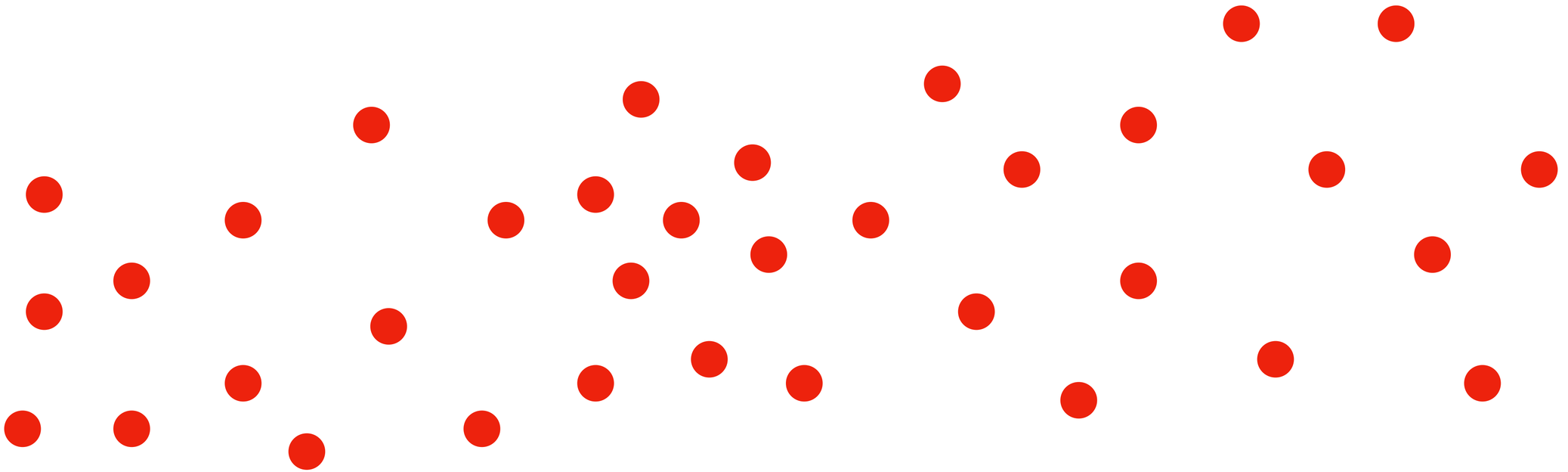


What is Clustering?



Clustering

It’s Not Always Easy…



Clustering

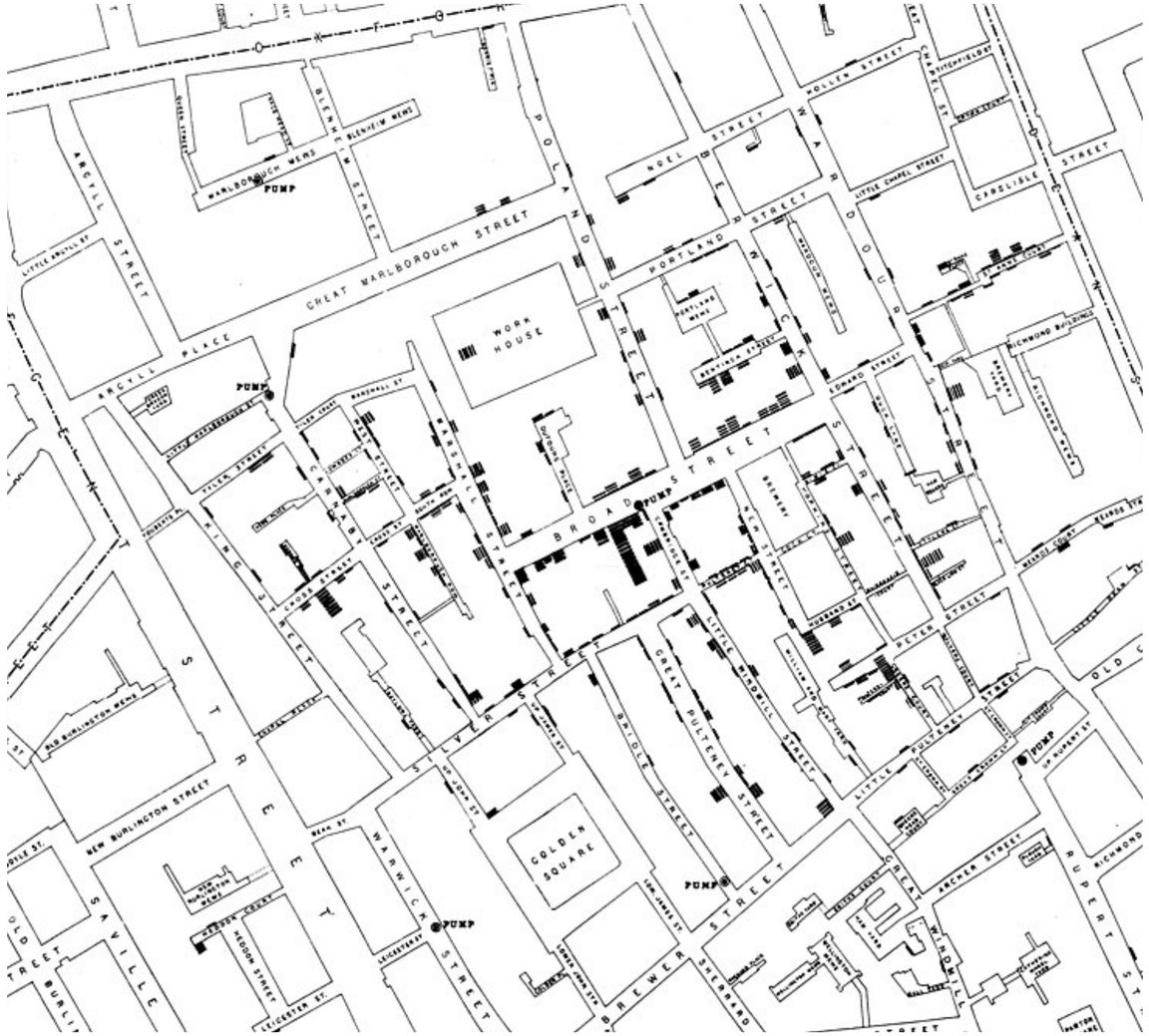
- Many different uses for clustering:
  - **Understand** data better: find groups in data
  - **Identify possible classes** that can be predicted by classification
  - Extract **representative examples** from each cluster
  - **Data Reduction**: only work with a few examples per cluster
  - As a **preprocessing** step: identify outliers
  - To **split up** data: process each cluster on a different machine
  - To **speed up similarity search**: only look within certain clusters

Applications

- **Biology/Genetics**: identify similar entities (organisms/genomes)
- **Information Retrieval**: find clusters of similar documents
- **Marketing**: identify customers with similar behaviour
- **Urban Planning**: organise regions into similar land-use
- **Climate**: find patterns of weather behaviour
- **Sociology**: find groups of people with similar views

An Early Application

- John Snow plotted the location of cholera cases on a map during an outbreak in the summer of 1854
- His hypothesis was that the disease was carried in water, so he plotted location of cases and water pumps, identifying the source
- Clusters easy to identify in two dimensions, more points and higher dimension?



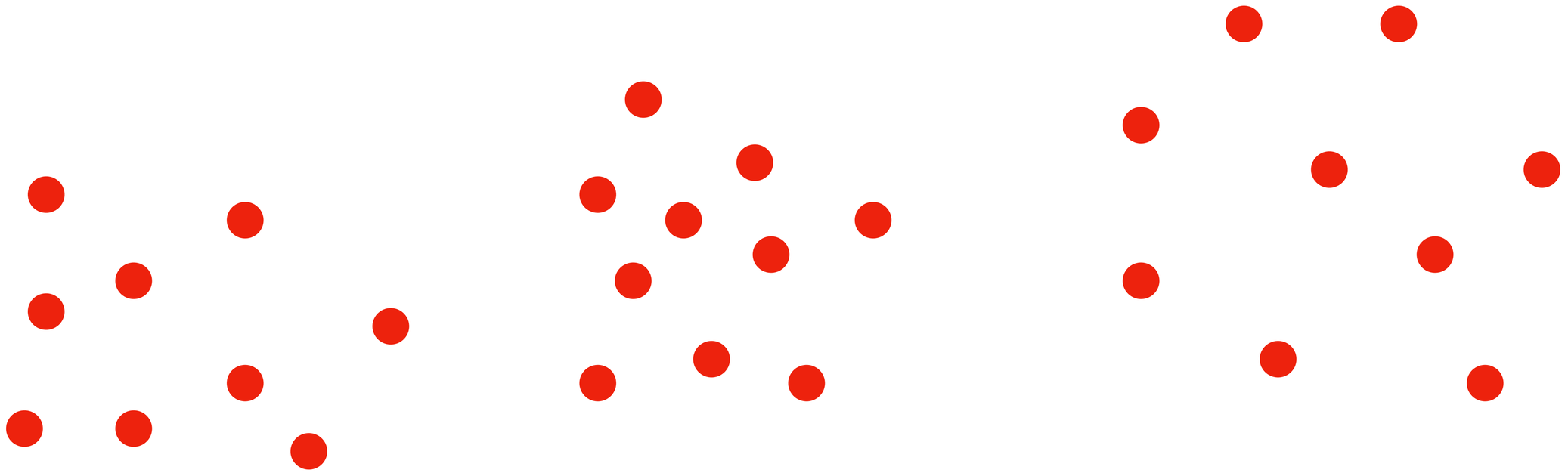
Snow, J., 1856. On the mode of communication of cholera. *Edinburgh medical journal*, *1*(7), p.668.
Ph.ucla.edu. n.d. *High resolution maps of John Snow*. [online] Available at: <https://www.ph.ucla.edu/epi/snow/highressnowmap.html> [Accessed 30 July 2021].

Overview

- Clustering has an intuitive appeal
  - People often talk “informally” about “clusters”: “cancer clusters”, “disease clusters”, “crime clusters”, …
- Will define what is meant by clustering, formalise the goals of clustering, and give algorithms for clustering data
  - Which **attributes** will be used for clustering, what is an appropriate **distance** between objects to use, what **objective** should the clustering optimise, **how many** clusters should we aim to find, how to **evaluate** the results?

Clustering

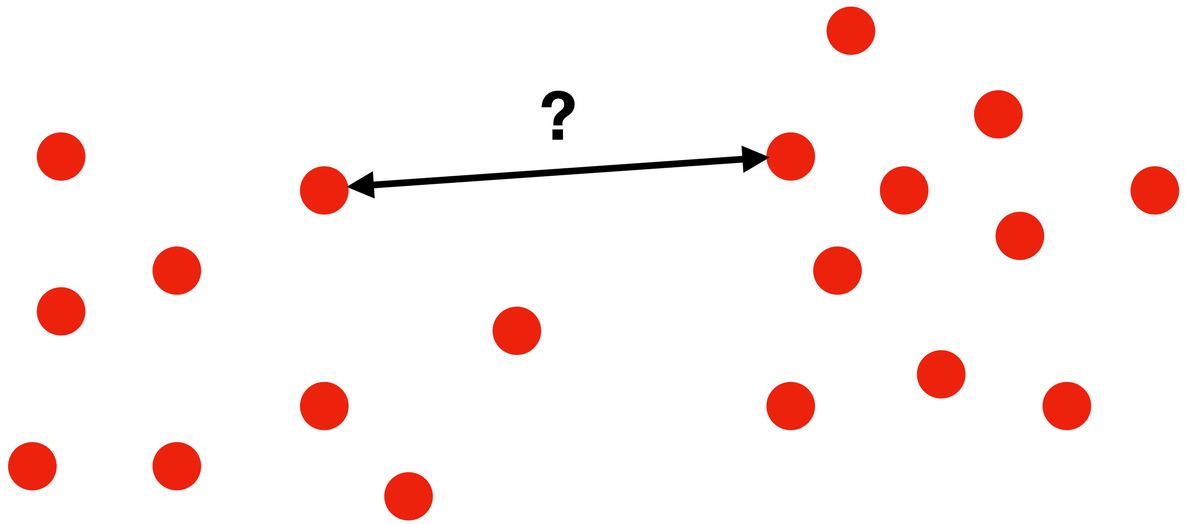
Objectives

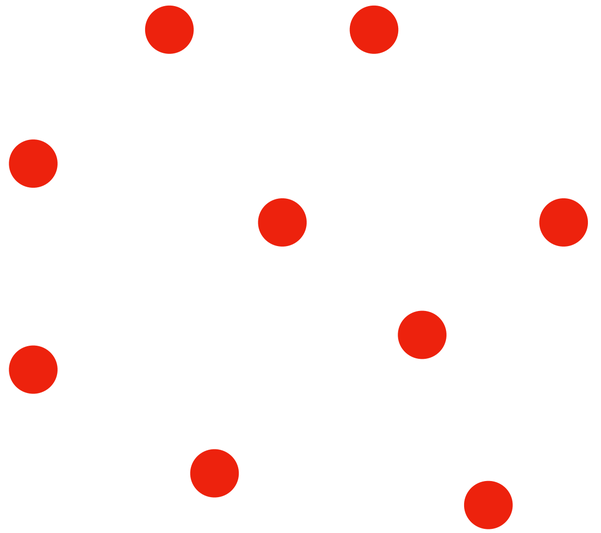


Informally, there are items, and we want to **group** them into **clusters.**

Clustering

Objectives



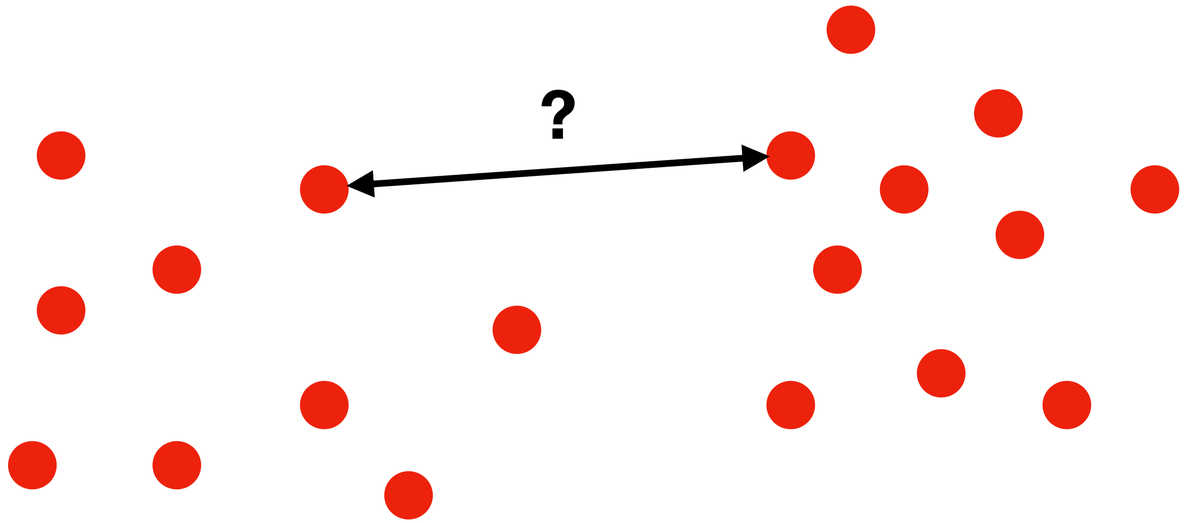


But how do we measure the distance between points?

Informally, there are items, and we want to **group** them into **clusters.**

Clustering

Objectives



But how do we measure the distance between points?

It's obvious,… right? Maybe in 2D… or is it?

We could use the Euclidean distance.

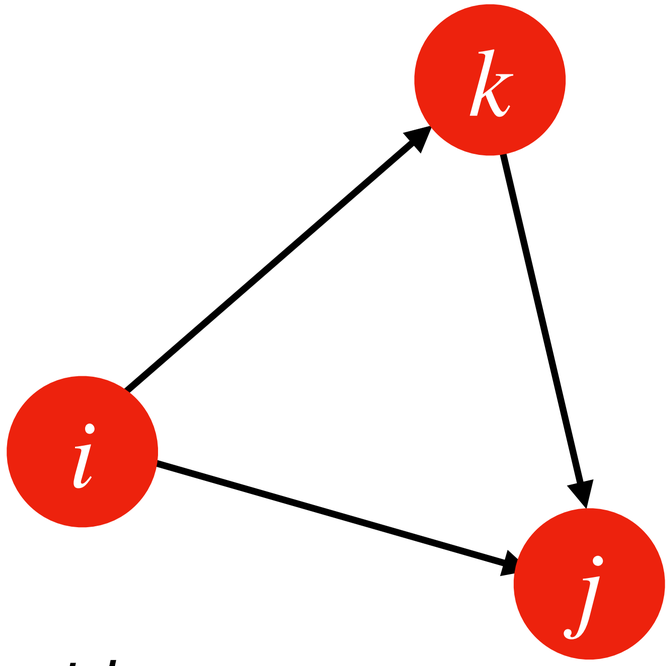
But what about data that is a mix of different types (e.g. time, text, boolean values). Should we weight different attributes?

Some clustering algorithms assume a certain distance function, with many assuming Euclidean distance.

Distances Refresher

- The most common distance measures satisfy the following mathematical properties:

  1. **Non-negativity**: $d(i,j) \geq 0$
  2. **Identity**: $d(i,i) = 0$
  3. **Symmetry**: $d(i,j) = d(j,i)$
  4. **Triangle Inequality**: $d(i,j) \leq d(i,k) + d(k,j)$



- Properties which obey these conditions are known as *metrics*.

Distances

- Most distance measurements of interest are metrics:
  - Euclidean distance ($d(i,j) = \sqrt{(i_1 - j_1)^2 + (i_2 - j_2)^2 + \ldots + (i_n - j_n)^2}$)
  - L_1 distance ($d(i,j) = \sum_{k=1}^{n} |\, i_k - j_k \,|$, sum of absolute coordinate differences)
  - L_∞ distance ($d(i,j) = max_k(|\, i_k - j_k \,|)$, maximum absolute coordinate difference)
  - String Edit distance (number of changes to turn one string into another):
    - The Hamming distance is a type of string edit distance.

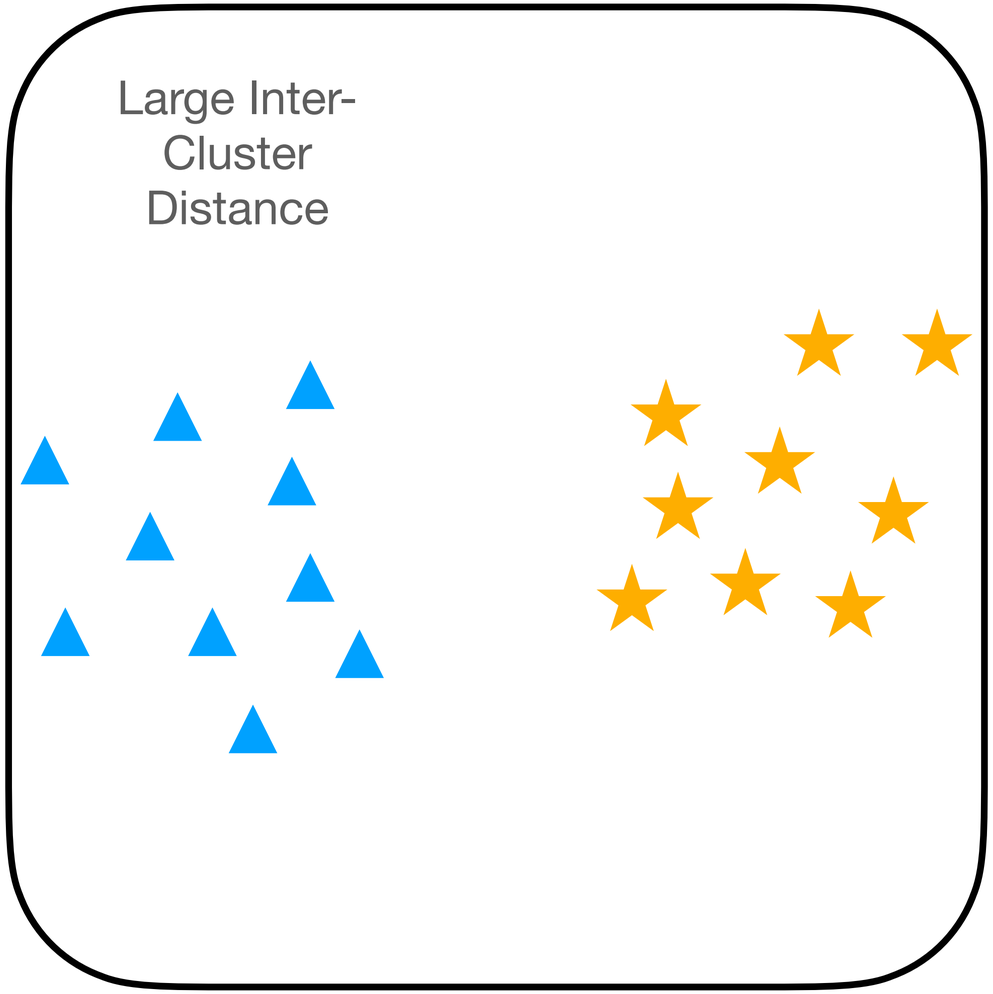
Clustering

Objectives

- Divide data into k clusters C so that
  - Inter-cluster distance (between clusters) is large
    - The clusters are **well-separated**
  - Intra-cluster distance (within cluster) is small
    - The clusters are **coherent**

Clustering

Inter- vs. Intra-Cluster Distance



Clustering

Objectives

- Divide data into k clusters C so that
  - Inter-cluster distance (between clusters) is large
    - The clusters are **well-separated**
  - Intra-cluster distance (within cluster) is small
    - The clusters are **coherent**

- Formalise this into a mathematical objective.
  - Eg. Formalise some measure of **Inter** distance $D(C)$,
  - Formalise some measure of **Intra** distance $T(C)$,
  - Try to maximise $(D(C) - T(C))$

Acknowledgments

- Han, J., Pei, J. and Kamber, M., 2012. *Data Mining: Concepts and Techniques*. Elsevier.
- Graham Cormode [Warwick, CS910]
- Florin Ciucu [Warwick, CS430/CS910]

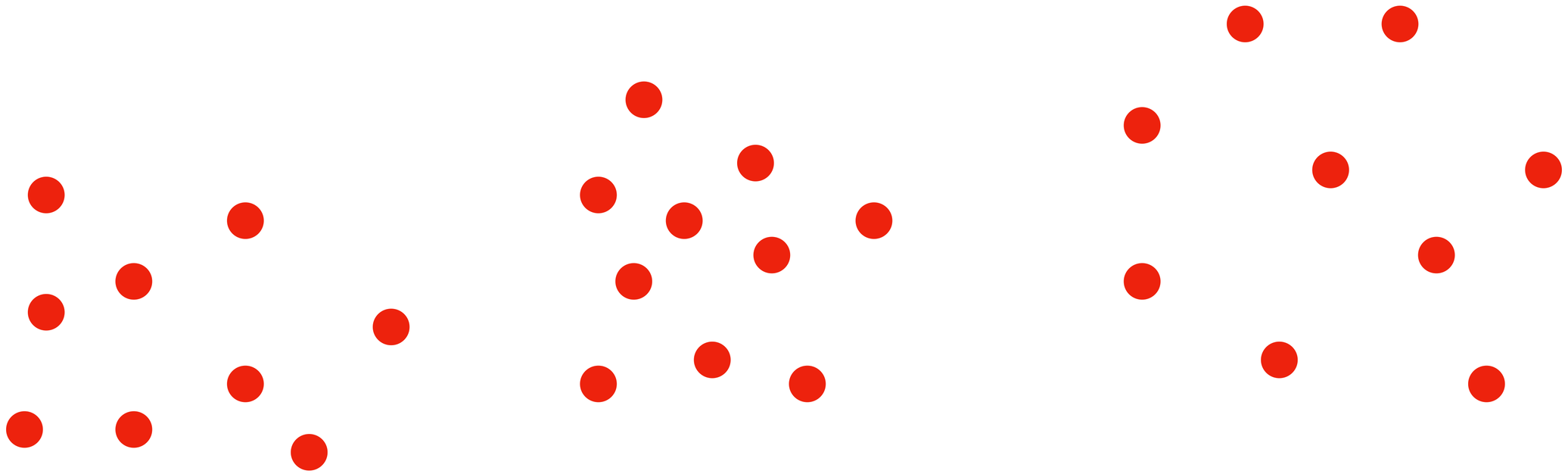
Part B: Clustering Algorithms

k-Means

1. Specify k (how many clusters you want).
2. Choose k objects (or points) at random from the data as cluster centres.
3. All data objects are assigned to their closest cluster centre (using squared Euclidean distance).
4. The mean (or *centroid*) of each cluster is calculated. These become the new cluster centres.
5. Repeat steps 3-4 until convergence.

k-Means

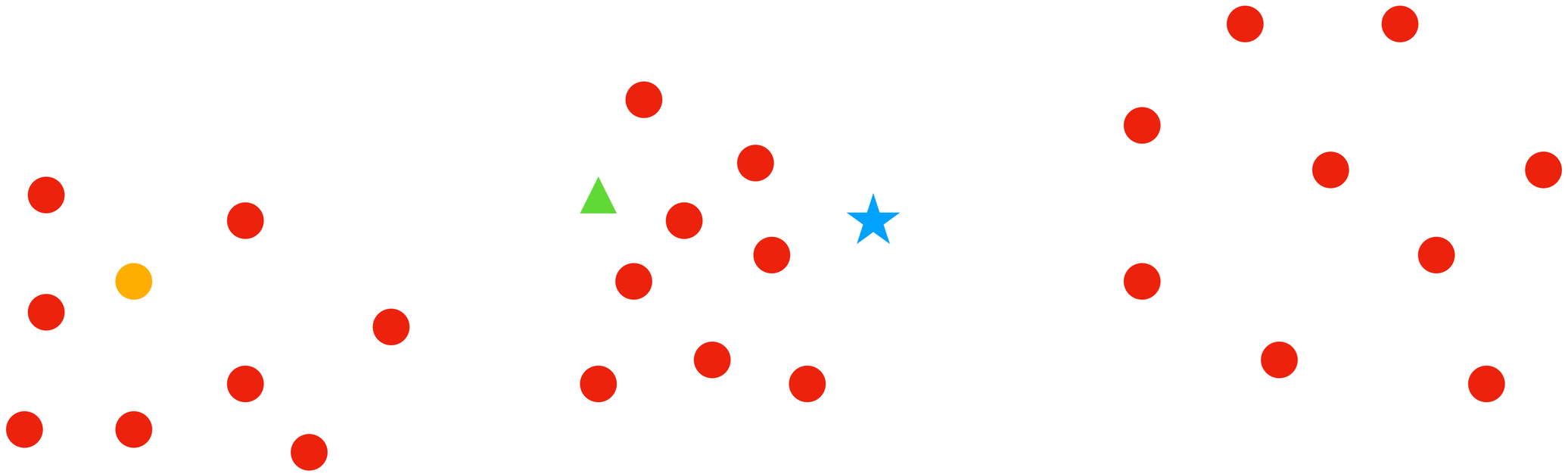
Example



Start with this data.

k-Means

Example

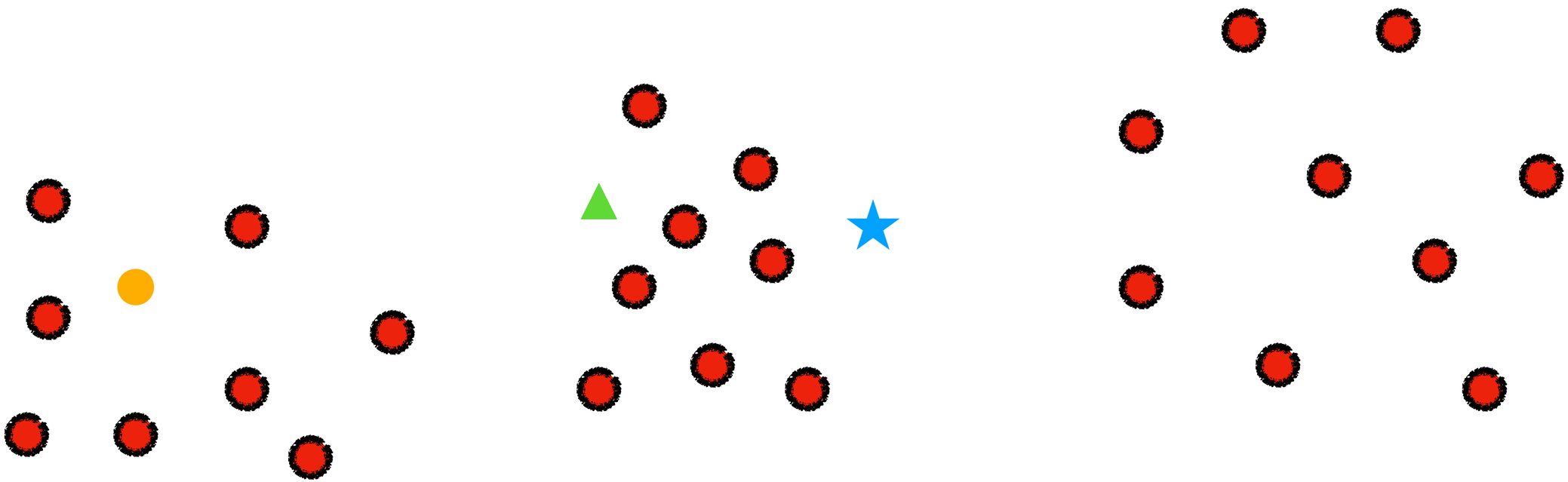


1. Specify k (lets say 3).

2. Choose k objects at random from the data as cluster centres.

k-Means

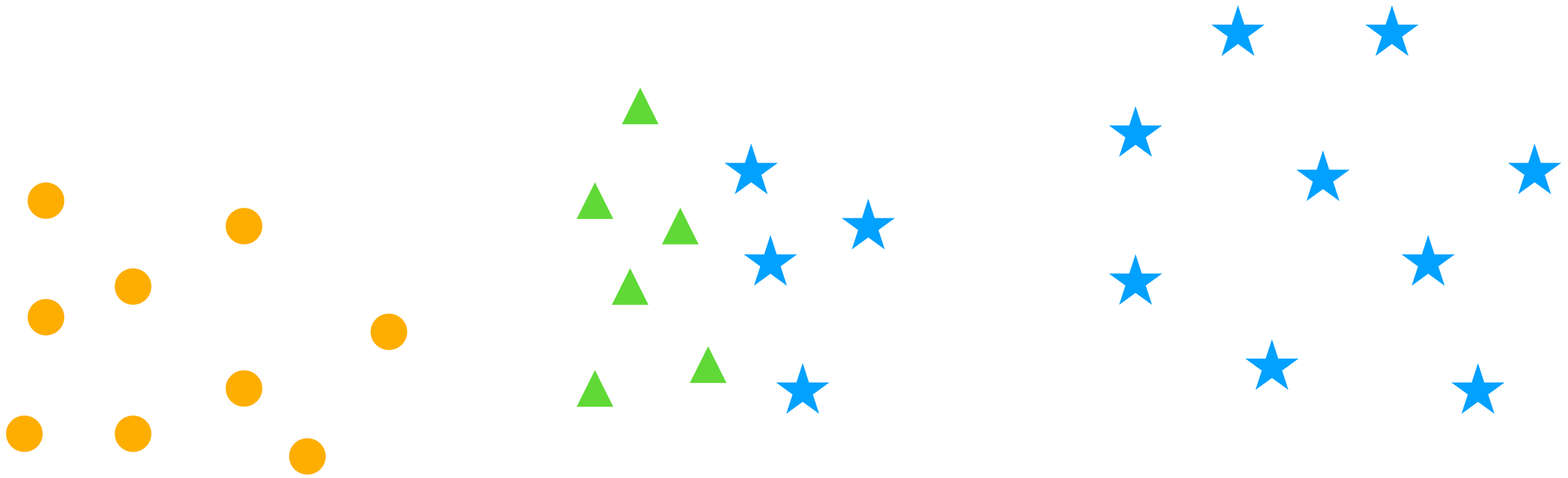
Example



3. All data objects are assigned to their closest cluster centre (using squared Euclidean distance).

k-Means

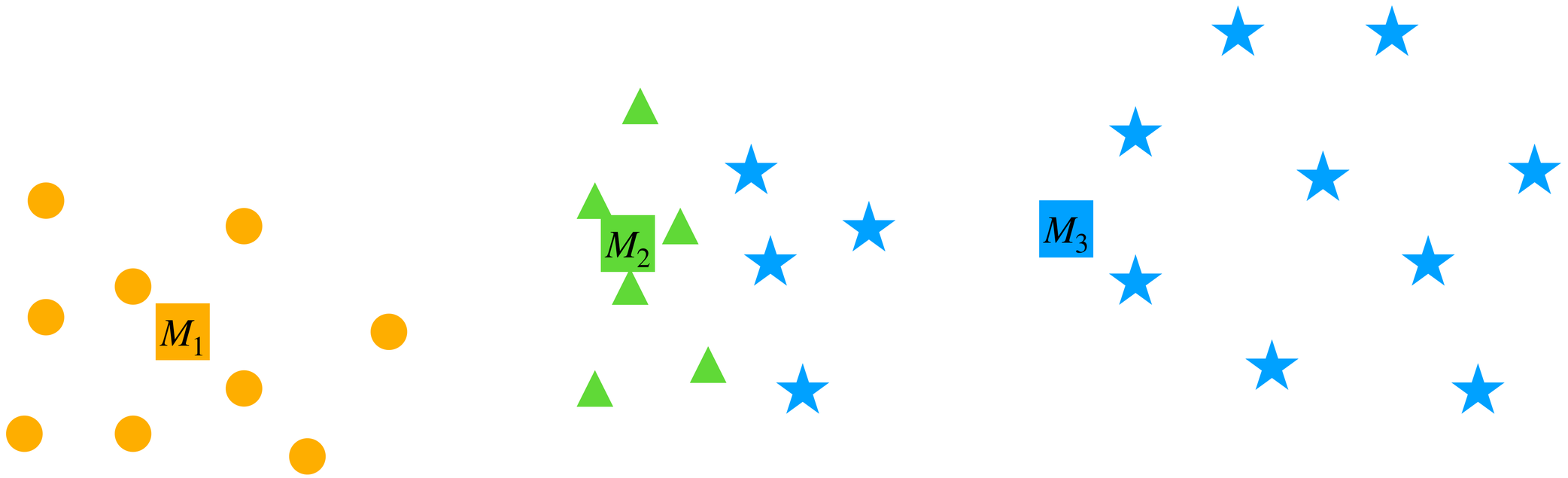
Example



3. All data objects are assigned to their closest cluster centre (using squared Euclidean distance).

k-Means

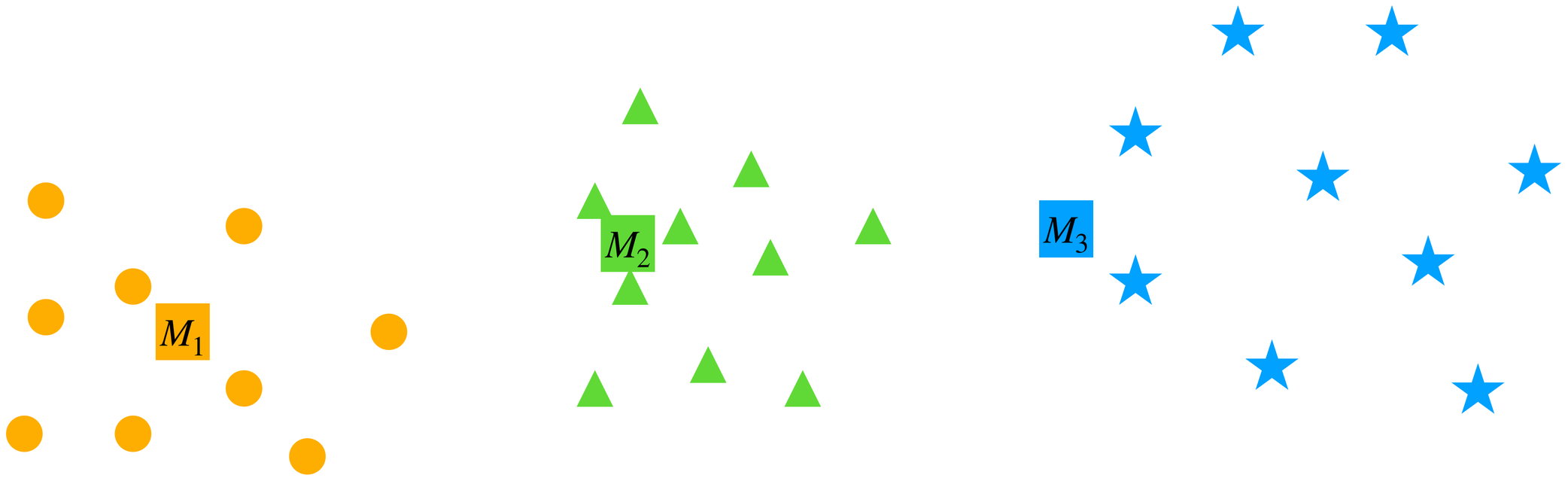
Example



4. The mean (or *centroid*) of each cluster is calculated. These become the new cluster centres.

k-Means

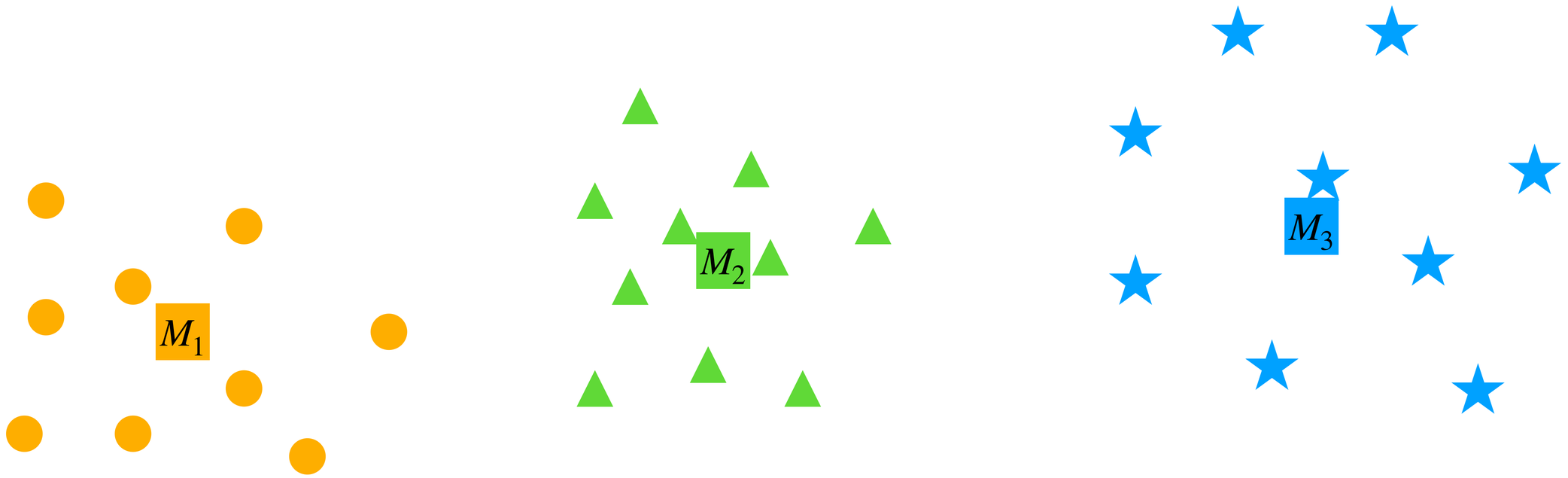
Example



5. Repeat steps 3-4 until convergence.

k-Means

Example



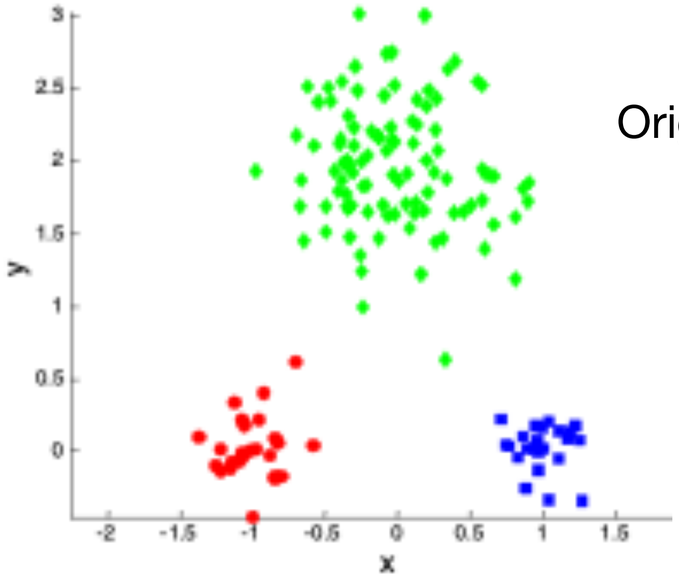
5. Repeat steps 3-4 until convergence.

k-Means

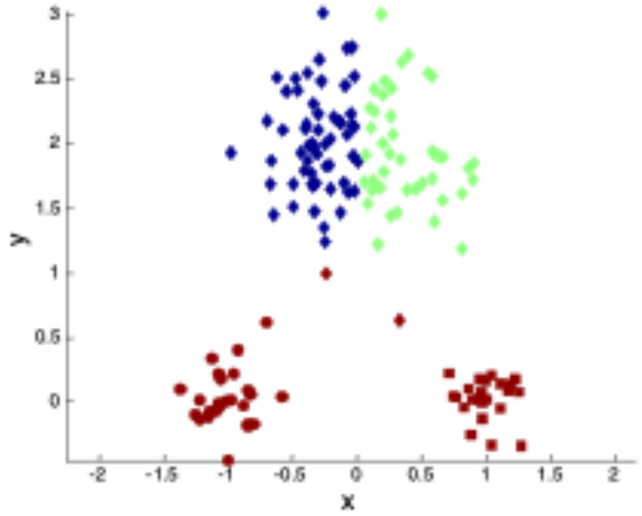
Advantages and Disadvantages

- Advantages:
  - Simple to understand.
  - Easy to compute.
- Disadvantages:
  - k must be manually specified.
  - Very sensitive to initial cluster centres.
  - Outliers can use up clusters.

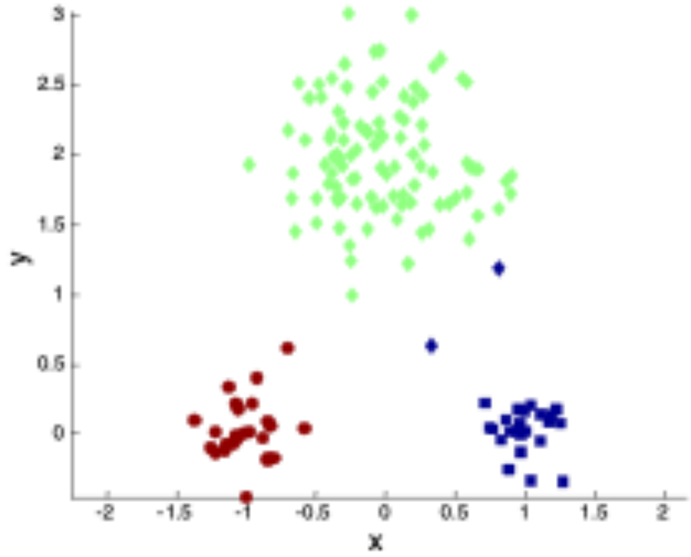
Two Different k-Means Clusterings



Original Data



Sub-optimal Clustering



Optimal Clustering

k-Means

Possible Improvements

- Run algorithm multiple times and choose the best result (the one with smallest total squared distance between data objects and closest cluster centre).

- Use improved versions of k-Means, such as **k-Means++** (chooses successive initial cluster centres with probability that is proportional to the square of the distance from the closest cluster centre).

k-Means Evaluation in Weka

```
=== Run information ===

Scheme:       weka.clusterers.SimpleKMeans -init 0 -max-candidates 100 -periodic-pruning 10000 -min-density 2.0 -t1 -1.25 -t2 -1.0 -N 3 -A "weka.core
Relation:     iris
Instances:    150
Attributes:   5
              sepallength
              sepalwidth
              petallength
              petalwidth
Ignored:
              class
Test mode:    Classes to clusters evaluation on training data

=== Clustering model (full training set) ===


kMeans
======

Number of iterations: 6
Within cluster sum of squared errors: 6.998114004826762

Initial starting points (random):

Cluster 0: 6.1,2.9,4.7,1.4
Cluster 1: 6.2,2.9,4.3,1.3
Cluster 2: 6.9,3.1,5.1,2.3

Missing values globally replaced with mean/mode
```

k-Means Evaluation in Weka

```
                                Cluster#
Attribute       Full Data           0           1           2
                  (150.0)      (61.0)      (50.0)      (39.0)
=============================================================
sepallength       5.8433      5.8885       5.006      6.8462
sepalwidth         3.054      2.7377       3.418      3.0821
petallength       3.7587      4.3967       1.464      5.7026
petalwidth        1.1987       1.418       0.244      2.0795



Time taken to build model (full training data) : 0.01 seconds

=== Model and evaluation on training set ===

Clustered Instances

0       61 ( 41%)
1       50 ( 33%)
2       39 ( 26%)


Class attribute: class
Classes to Clusters:

  0  1  2  <-- assigned to cluster
  0 50  0 | Iris-setosa
 47  0  3 | Iris-versicolor
 14  0 36 | Iris-virginica

Cluster 0 <-- Iris-versicolor
Cluster 1 <-- Iris-setosa
Cluster 2 <-- Iris-virginica

Incorrectly clustered instances :       17.0     11.3333 %
```

k-Means

- The quality of our clustering can be measured by the within-cluster variation:

$$E = \sum_{i=1}^{k} \sum_{p \in C_i} dist(\mathbf{p}, \mathbf{c_i})^2$$

- where E is the sum of the squared error for all objects in the data set, $\mathbf{p}$ is a given object, and $\mathbf{c_i}$ is the centroid of cluster C_i.

- Problem with k-Means: sensitivity to outliers.
  - When outliers are assigned to a cluster, they can dramatically alter the mean value of the cluster. Problem exacerbated due to use of **squared** error!

k-Medoids

- Consider seven objects in 1D space: 4,5,6,14,15,17,42
- What would be a good clustering? Maybe {4,5,6}, {14,15,17}, with 42 excluded (because it is an outlier)?
- Apply k-Means using k=2, the partitioning ({4,5,6},{14,15,17,42}) has the within-cluster variation:

$$(4-5)^2+(5-5)^2+(6-5)^2+(14-22)^2+(15-22)^2+(17-22)^2+(42-22)^2 = 540$$

as the mean of {4,5,6} is 5, and the mean of {14,15,17,42} is 22.

Han, J., Pei, J. and Kamber, M., 2012. *Data Mining: Concepts and Techniques*. Elsevier.

k-Medoids

- Now let's try ({4,5,6,14},{15,17,42})
- $(4-7.25)^2+(5-7.25)^2+(6-7.25)^2+(14-7.25)^2+(15-24.67)^2+(17-24.67)^2+(42-24.67)^2 = 515.4167$

  as the mean of {4,5,6,14} is 7.25, and the mean of {15,17,42} is 24.67.

- As 515.4167<540, k-Means would assign 14 to the first cluster, due to the outlier (42).
- The centre of {15,17,42} is 24.67, which is much higher than all member of the cluster (except for the outlier).

Han, J., Pei, J. and Kamber, M., 2012. *Data Mining: Concepts and Techniques*. Elsevier.

k-Medoids

- How can we alter k-Means to prevent this problem?
- Instead of taking the mean value of the cluster, we can instead take one representative object per cluster.
- The remaining objects are assigned to the cluster where the representative object is the most similar.

k-Medoids

- Objective: Minimise the sum of dissimilarities between each object $\mathbf{p}$ and its representative object (most similar object), i.e.:

$$E = \sum_{i=1}^{k} \sum_{p \in C_i} dist(\mathbf{p}, \mathbf{o_i})$$

- E is the sum of the absolute error for all objects ($\mathbf{p}$). $\mathbf{o_i}$ is the representative object of cluster C_i.

k-Medoids

- Example k-Medoids algorithm: the Partitioning Around Medoids (Algorithm).

1. Arbitrarily choose the initial representative objects.
2. Replace the representative object with a non representative object, and see how this impacts clustering quality.
3. Step 2 continues until the quality of the resulting clustering cannot be improved.

Kaufman, L. and Rousseeuw, P.J., 2009. Finding groups in data: an introduction to cluster analysis (Vol. 344). John Wiley & Sons.

k-Medoids

- Let $\mathbf{o_1}, \ldots, \mathbf{o_k}$ be the set of current representative objects (medoids).
- Let $\mathbf{o_{random}}$ be a current nonrepresentative object, and we wish to determine whether $\mathbf{o_{random}}$ would be a good replacement for a current medoid $\mathbf{o_j}$ $(1 \leq j \leq k)$.
- Calculate the distance from every object $\mathbf{p}$ to the closest object in the set $\mathbf{o_1}, \ldots, \mathbf{o_{j-1}}, \mathbf{o_{random}}, \mathbf{o_{j+1}}, \ldots, \mathbf{o_k}$.

k-Medoids

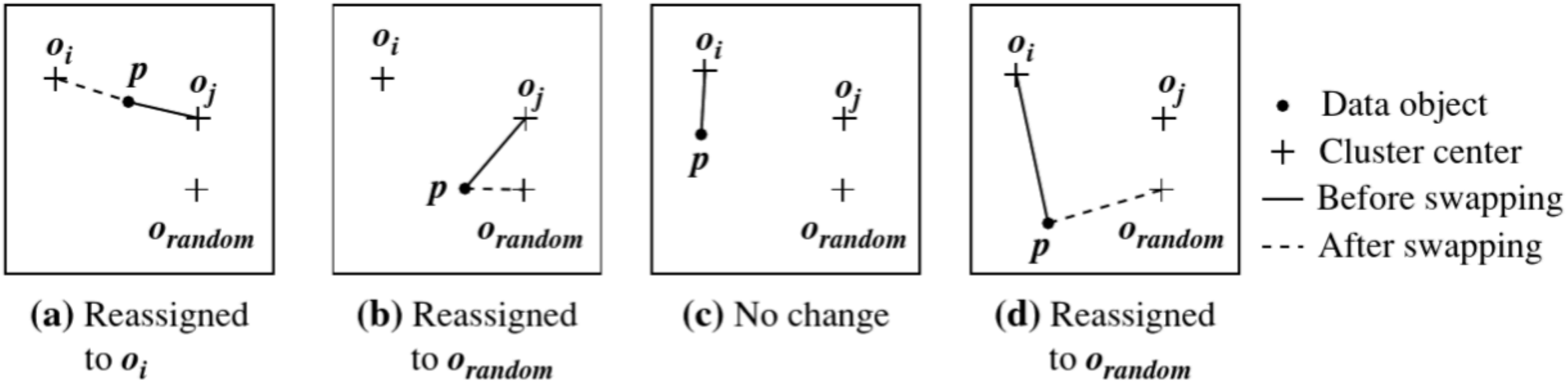


Figure from: Han, J., Pei, J. and Kamber, M., 2012. *Data Mining: Concepts and Techniques*. Elsevier.

k-Medoids

- Each time a reassignment occurs, the sum of dissimilarities between each object p and its representative object changes. Recall:

$$E = \sum_{i=1}^{k} \sum_{p \in C_i} dist(\mathbf{p}, \mathbf{o_i})$$

- The cost function calculates the difference in the absolute-error value if an existing representative object is replaced by a non-representative object.

- If the absolute error E is reduced, $\mathbf{o_j}$ is swapped with $\mathbf{o_{random}}$, otherwise, there is no change.

k-Medoids vs k-Means

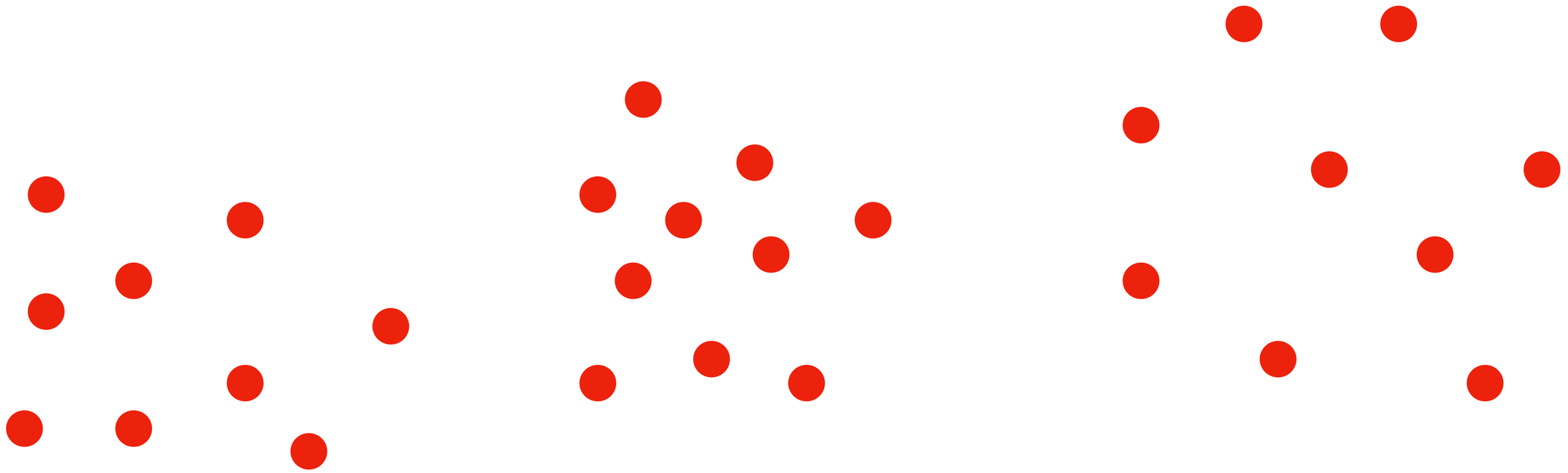
- k-Medoids is more robust than k-Means to noise and outliers.
  - A medoid is less influenced by noise and outliers.
- However, k-Medoids become very costly for large values of n and k.
- Disadvantage of both: k must be specified!

k-Centers

- Want to minimise diameter of each cluster.
  - Pick some object from the data as the first centre. Repeat:
    - For each data object, compute its distance d_{min} from its closest centre
    - Find the data object that maximises d_{min}
    - Add this object to the set of centres
  - Until k centres are picked
- An example is *Farthest First Clustering*.

Farthest First Clustering, k=3

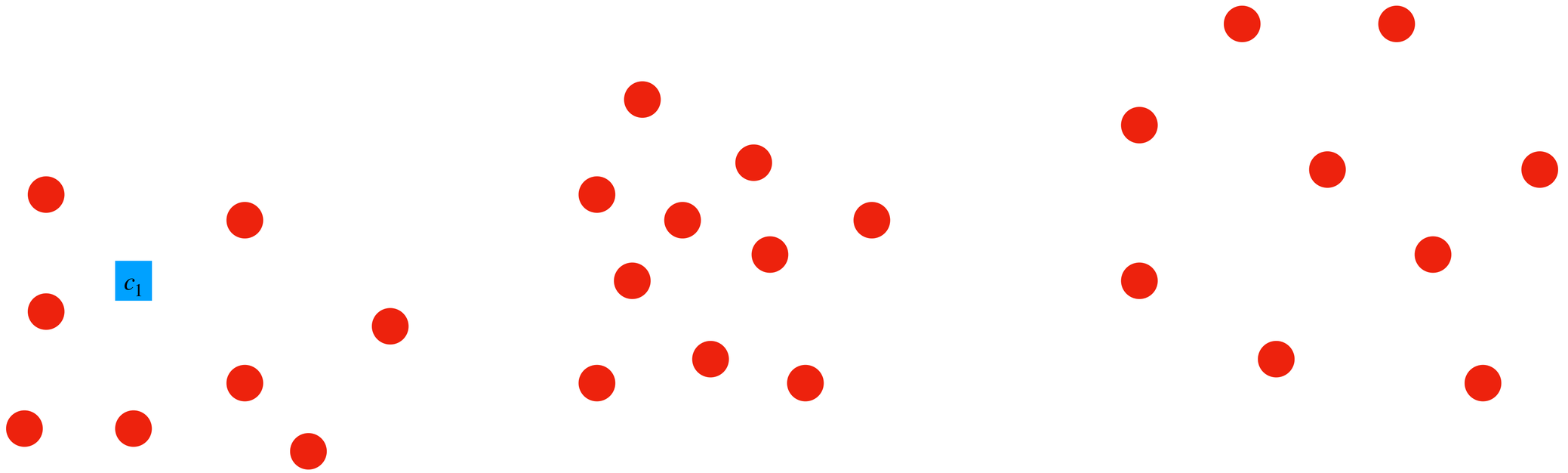
(Also called Furthest Point Clustering)



Initial data.

Farthest First Clustering, k=3

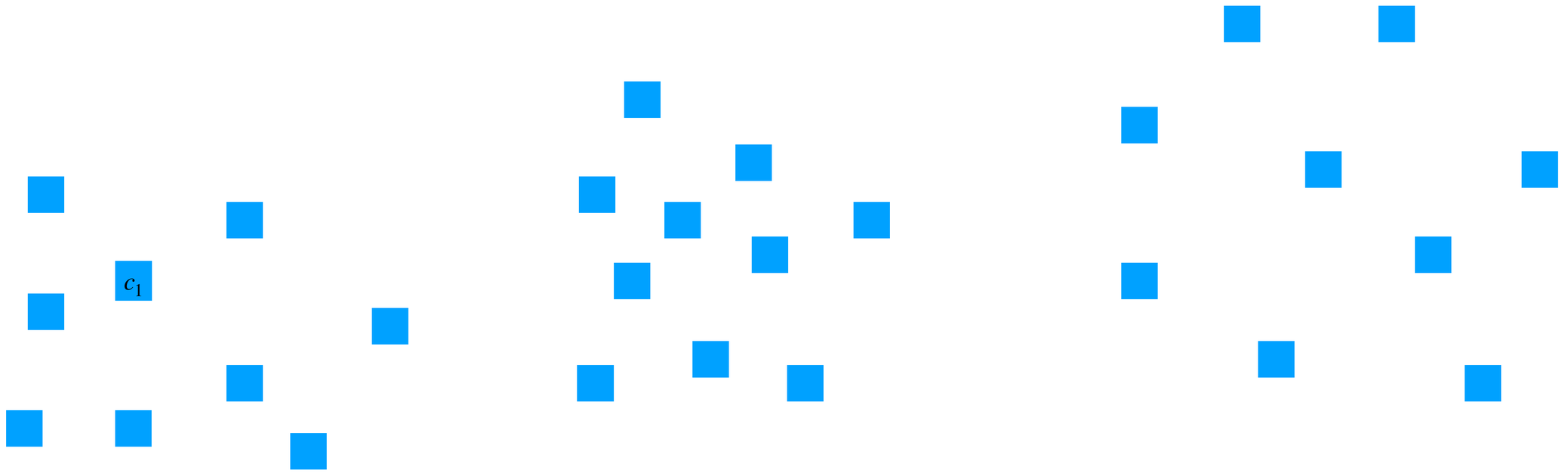
(Also called Furthest Point Clustering)



Step 1: Choose an arbitrary centre, c_1.

Farthest First Clustering, k=3

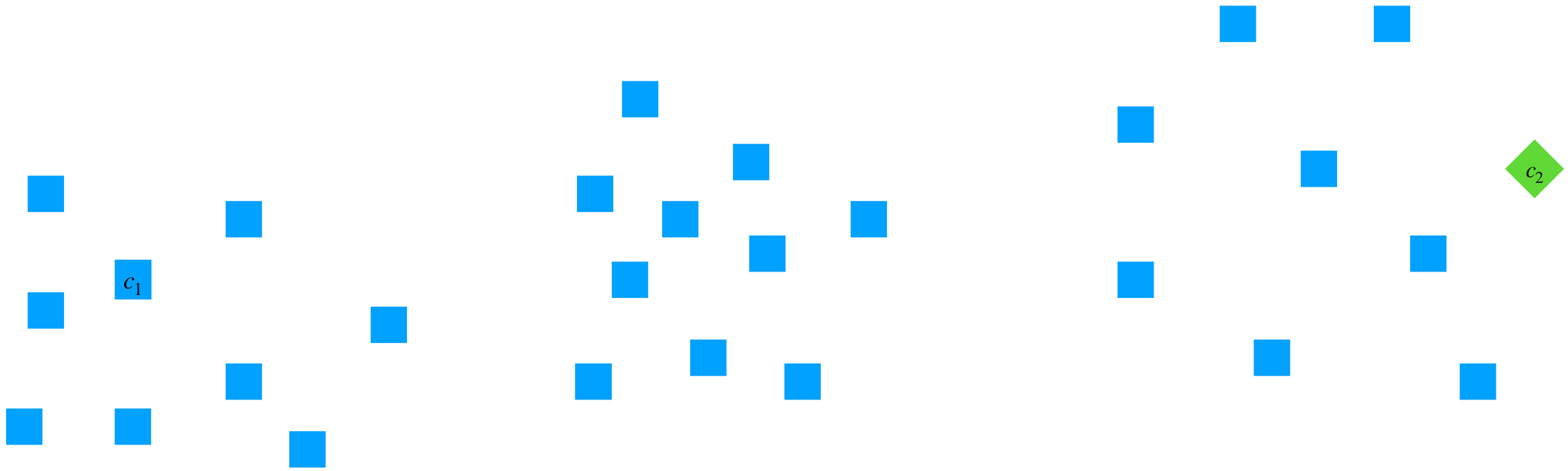
(Also called Furthest Point Clustering)



Step 2: Assign objects to their closest centre.

Farthest First Clustering, k=3

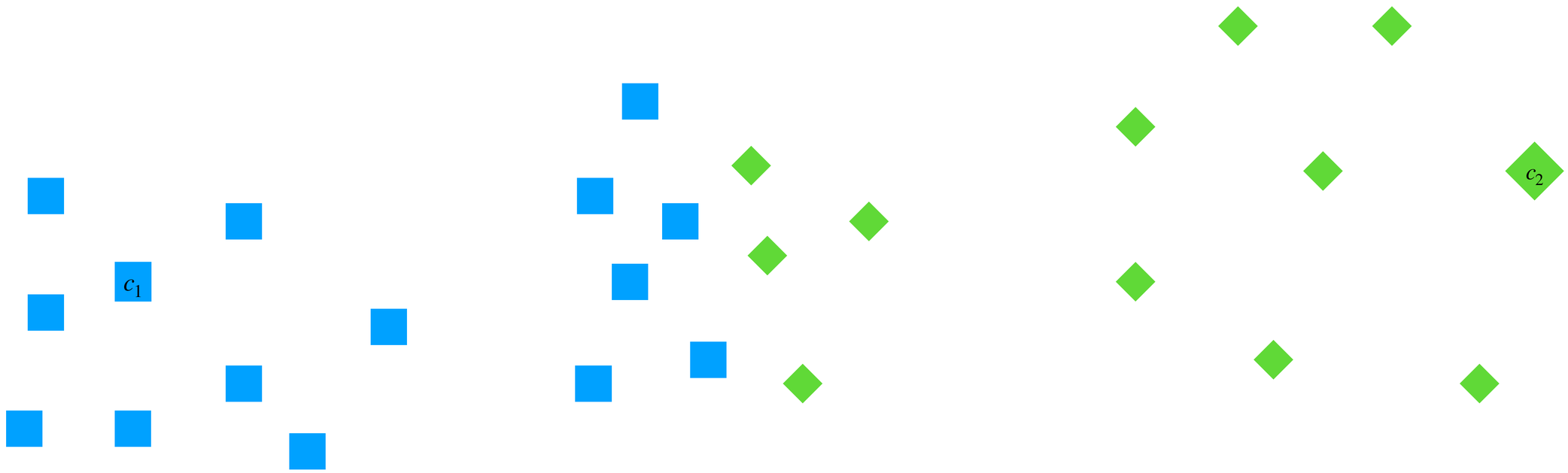
(Also called Furthest Point Clustering)



Step 3: Select the next centre, c_2, to be the one farthest from its closest centre.

Farthest First Clustering, k=3

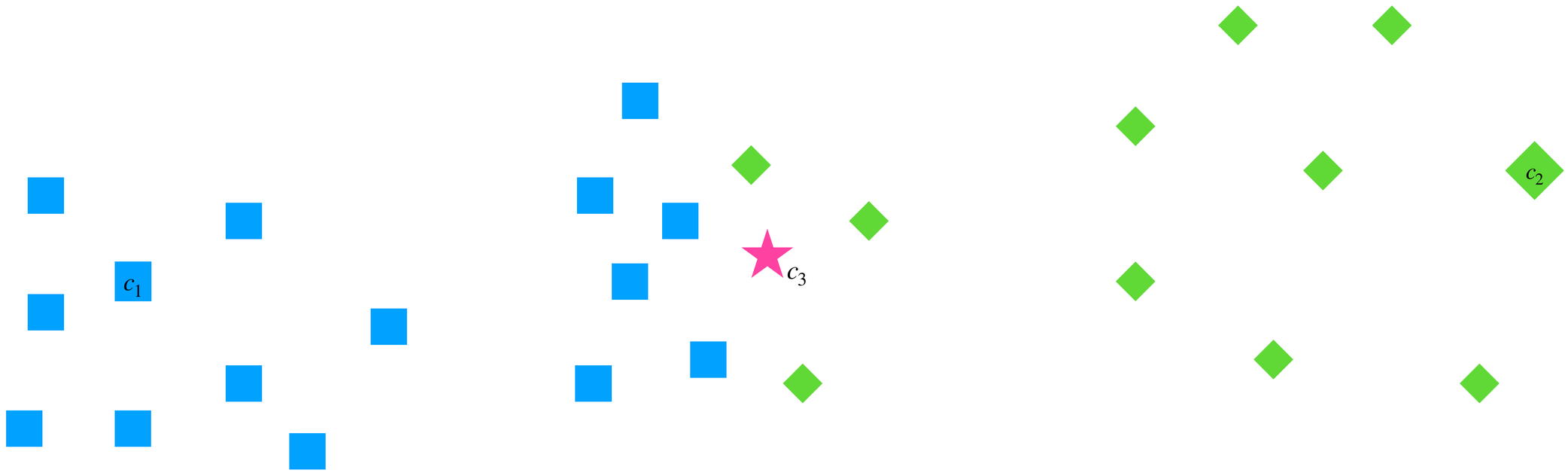
(Also called Furthest Point Clustering)



Step 4: Assign objects to their closest centre.

Farthest First Clustering, k=3

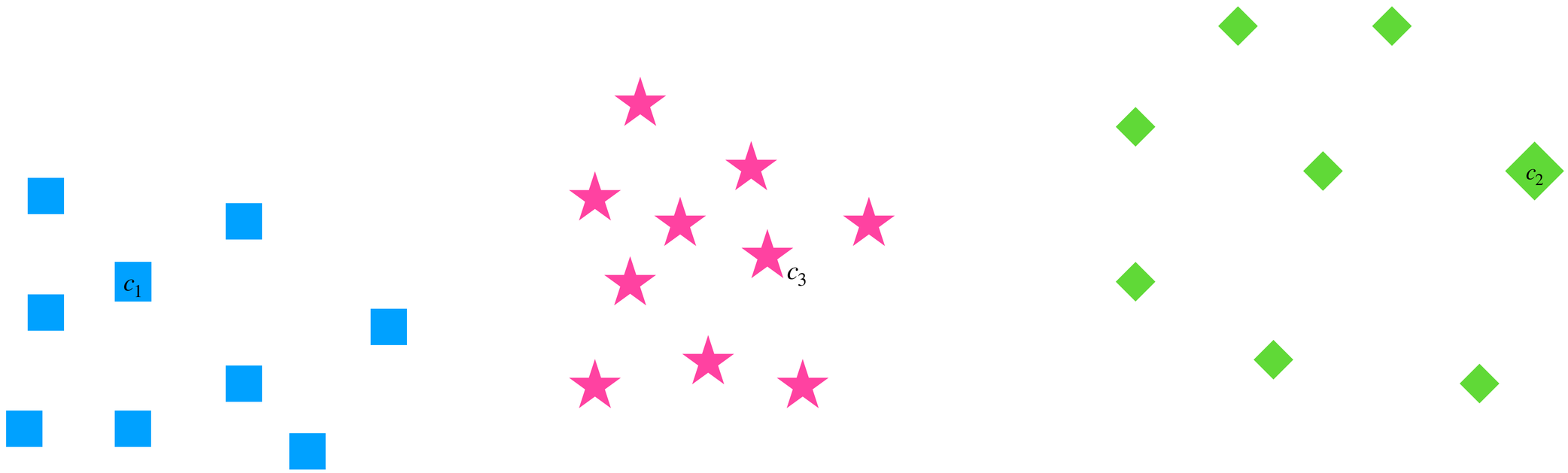
(Also called Furthest Point Clustering)



Step 5: Select the next centre, c_3, to be the one farthest from its closest centre.

Farthest First Clustering, k=3

(Also called Furthest Point Clustering)



Step 6: Assign objects to their closest centre.

Cluster Evaluation

- Harder to evaluate quality of a clustering
  - Can measure how well a method does based on its own criteria
  - But each method has different objectives
- Can use a (withheld) class attribute to define a “true” clustering
  - Assumes that the points cluster well based on class
  - Good for evaluating new methods against known data
  - Not good for evaluating known methods against new data!
  - Study the **confusion matrix** comparing class to clusters

Cluster Evaluation

- Will use the famous *iris* data set to evaluate clustering
  - Contains 50 samples from each of three types of iris (flower)
  - Measures sepal length, sepal width, petal length, petal width
  - In the Weka default data sets as *iris.arff*

Farthest First Clustering in Weka

```
Scheme:weka.clusterers.FarthestFirst -N 3 -S 1
Relation:     iris
Instances:    150
Attributes:   5
              sepallength
              sepalwidth
              petallength
              petalwidth
Ignored:      class
Test mode:Classes to clusters evaluation on training data

=== Model and evaluation on training set ===
Cluster centroids:

Cluster 0
  7.7 3.0 6.1 2.3
Cluster 1
  4.3 3.0 1.1 0.1
Cluster 2
  4.9 2.5 4.5 1.7
Time taken to build model (full training data) : 0 seconds
```

We set $k = 3$ (using domain knowledge)

Farthest First Clustering in Weka

```
=== Model and evaluation on training set ===
Clustered Instances
0       41 ( 27%)
1       50 ( 33%)
2       59 ( 39%)
Class attribute: class
Classes to Clusters:
  0  1  2  <-- assigned to cluster
  0 50  0 | Iris-setosa
  6  0 44 | Iris-versicolor
 35  0 15 | Iris-virginica

Cluster 0 <-- Iris-virginica
Cluster 1 <-- Iris-setosa
Cluster 2 <-- Iris-versicolor

Incorrectly clustered instances :  21.0  14       %
```

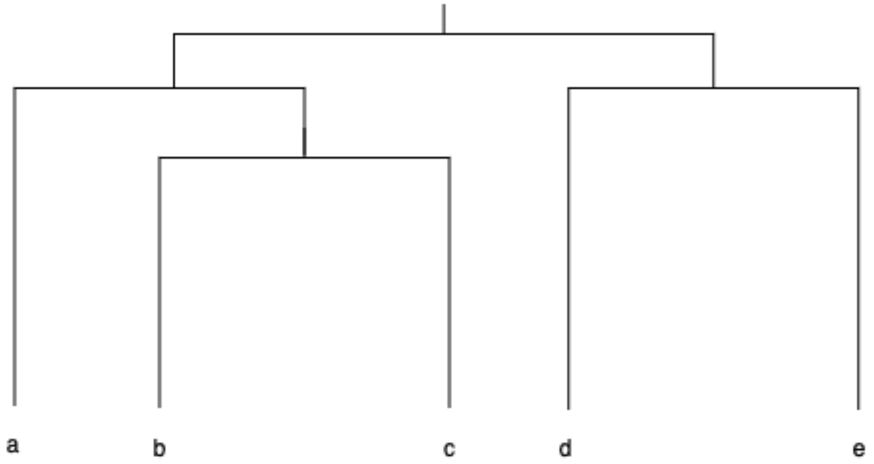
Acknowledgments

- Han, J., Pei, J. and Kamber, M., 2012. *Data Mining: Concepts and Techniques*. Elsevier.
- Graham Cormode [Warwick, CS910]
- Florin Ciucu [Warwick, CS430/CS910]

Part C: Clustering Algorithms 2

Hierarchical Clustering

- We may wish for data to be organised hierarchically:
  - For example, may want a cluster to represent all modules at the university. You could then partition the cluster into smaller clusters, where each cluster represents modules from an individual department.
    - Could then divide clusters even further, based on year of study!
- Useful for data visualisation!
  - Can view hierarchy of data as a tree.



Hierarchical Clustering

- Hierarchical clustering creates a hierarchical decomposition of a set of data objects.
- Two different types:
  - **Agglomerative Approach** (also called a **Bottom-Up Approach**): Starts with each object in a separate cluster, and continuously merges clusters until a terminating condition is met.
  - **Divisive Approach** (also called a **Top-Down Approach**): Starts with all objects in the same cluster, and divides the clusters until each object is in its own cluster, or a terminating condition is met.

Hierarchical Clustering

Agglomerative Clustering

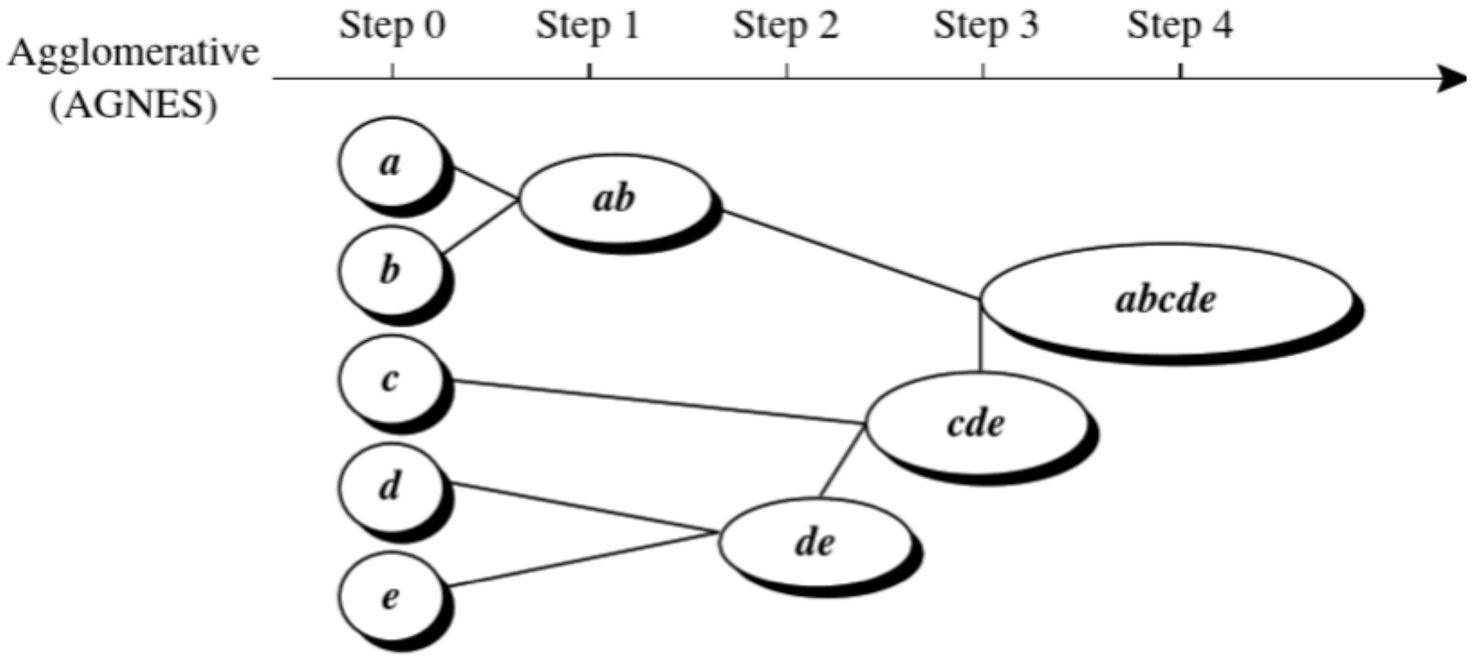
- **Agglomerative Approach** (also called a **Bottom-Up Approach**): Starts with each object in a separate cluster, and continuously merges clusters until a terminating condition is met.
- **Merging Step**: Find the two clusters that are the closest to each other (by some similarity measure), and merge into one cluster.

Hierarchical Clustering

Agglomerative Clustering - Example

- AGNES (***AG****glomerative* ***NES****ting*) is an example of an agglomerative hierarchical clustering approach.

- Let X be a set of five data objects, $\{a, b, c, d, e\}$.



Han, J., Pei, J. and Kamber, M., 2012. *Data Mining: Concepts and Techniques*. Elsevier.

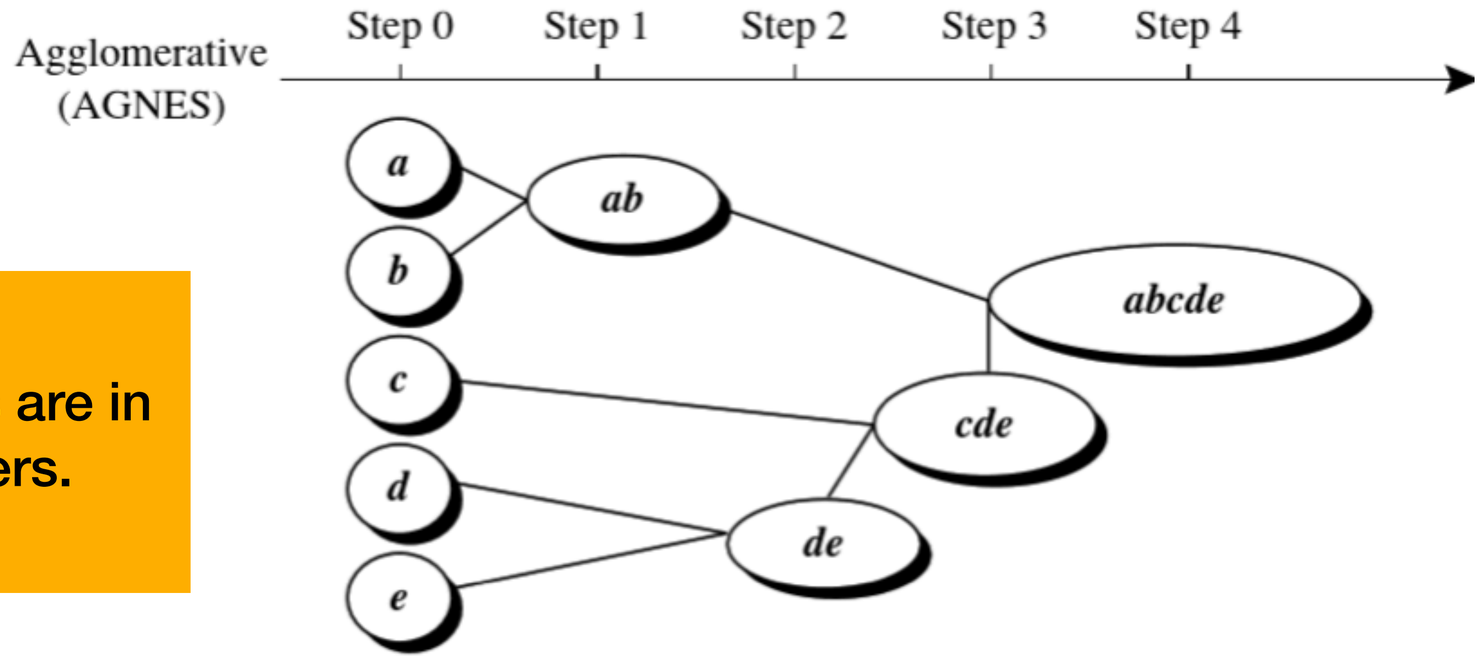
Kaufman, L. and Rousseeuw, P.J., 2009. Finding groups in data: an introduction to cluster analysis (Vol. 344). John Wiley & Sons.

Hierarchical Clustering

Agglomerative Clustering - Example

- AGNES (***AG****glomerative* ***NES****ting*) is an example of an agglomerative hierarchical clustering approach.

- Let X be a set of five data objects, $\{a, b, c, d, e\}$.



Step 0: All objects are in separate clusters.

Han, J., Pei, J. and Kamber, M., 2012. *Data Mining: Concepts and Techniques*. Elsevier.

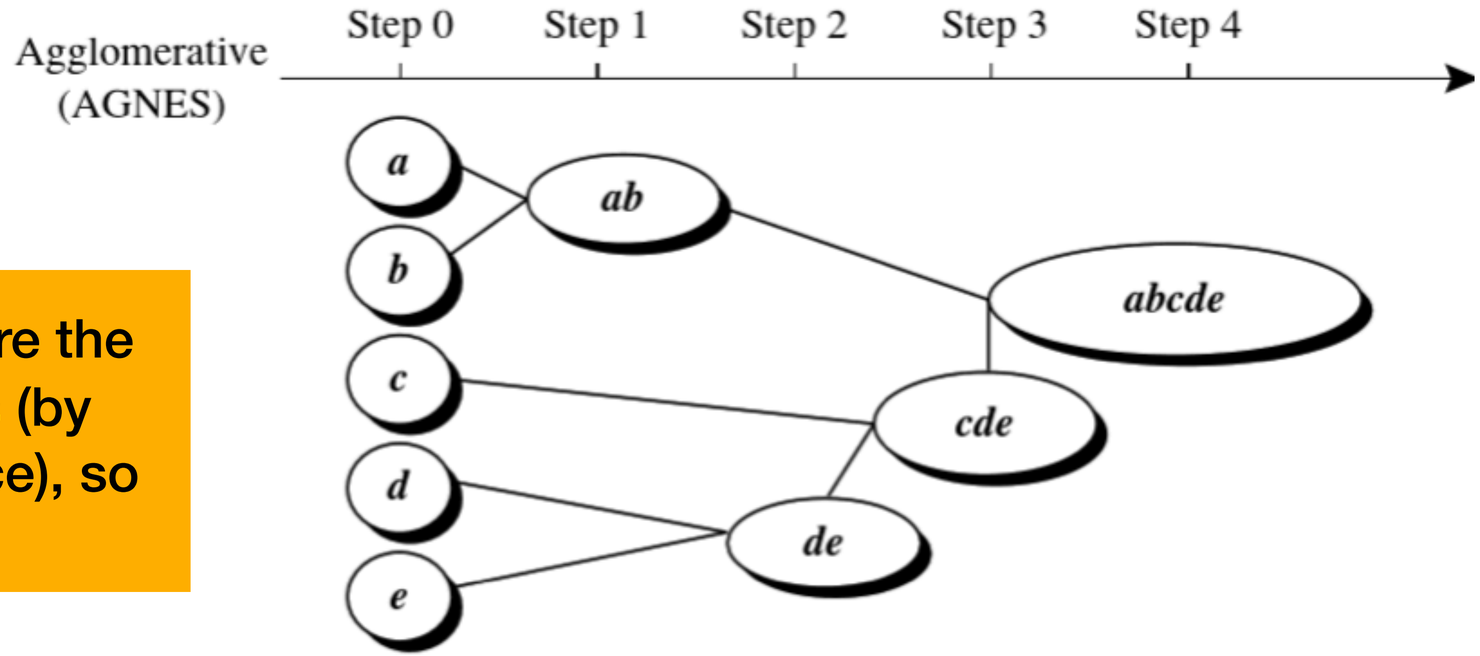
Kaufman, L. and Rousseeuw, P.J., 2009. Finding groups in data: an introduction to cluster analysis (Vol. 344). John Wiley & Sons.

Hierarchical Clustering

Agglomerative Clustering - Example

- AGNES (***AG****glomerative* ***NES****ting*) is an example of an agglomerative hierarchical clustering approach.

- Let X be a set of five data objects, $\{a, b, c, d, e\}$.



Step 1: a and b are the closest clusters (by Euclidean distance), so merge.

Han, J., Pei, J. and Kamber, M., 2012. *Data Mining: Concepts and Techniques*. Elsevier.

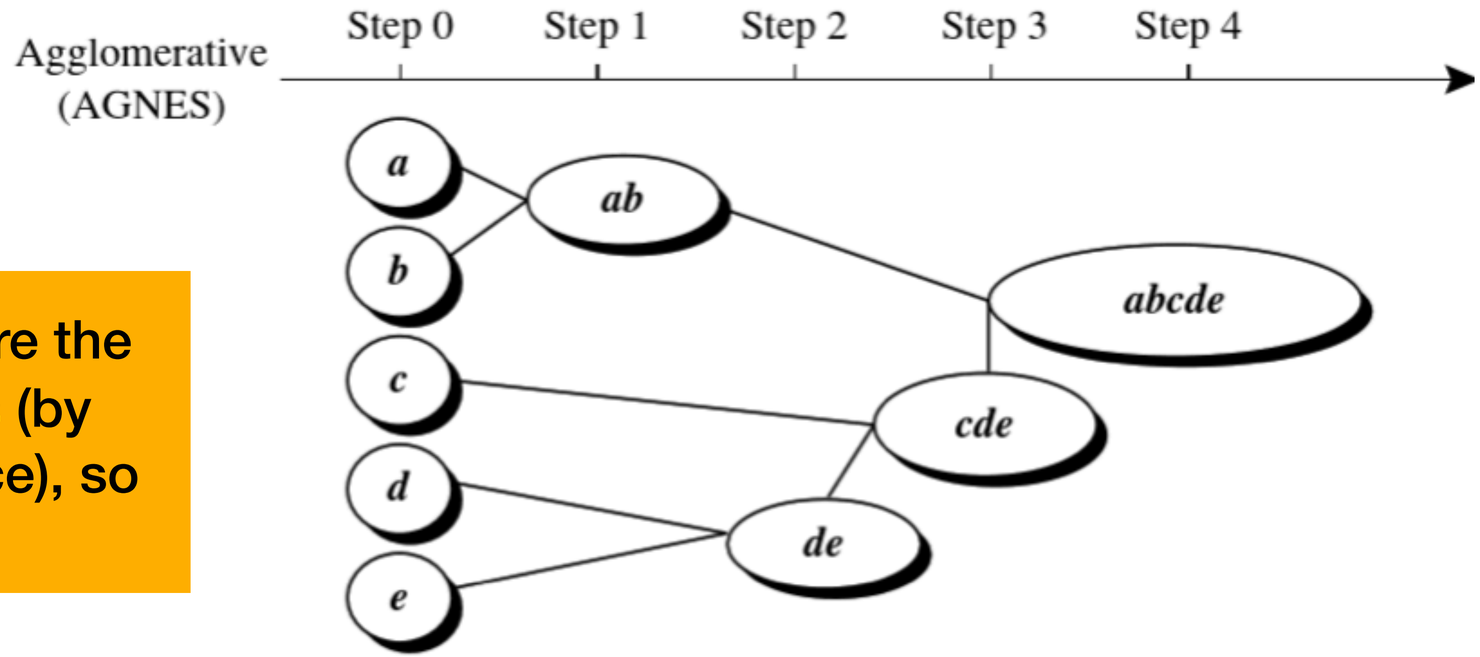
Kaufman, L. and Rousseeuw, P.J., 2009. Finding groups in data: an introduction to cluster analysis (Vol. 344). John Wiley & Sons.

Hierarchical Clustering

Agglomerative Clustering - Example

- AGNES (***AG****glomerative* ***NES****ting*) is an example of an agglomerative hierarchical clustering approach.

- Let X be a set of five data objects, $\{a, b, c, d, e\}$.



Step 2: d and e are the closest clusters (by Euclidean distance), so merge.

Han, J., Pei, J. and Kamber, M., 2012. *Data Mining: Concepts and Techniques*. Elsevier.

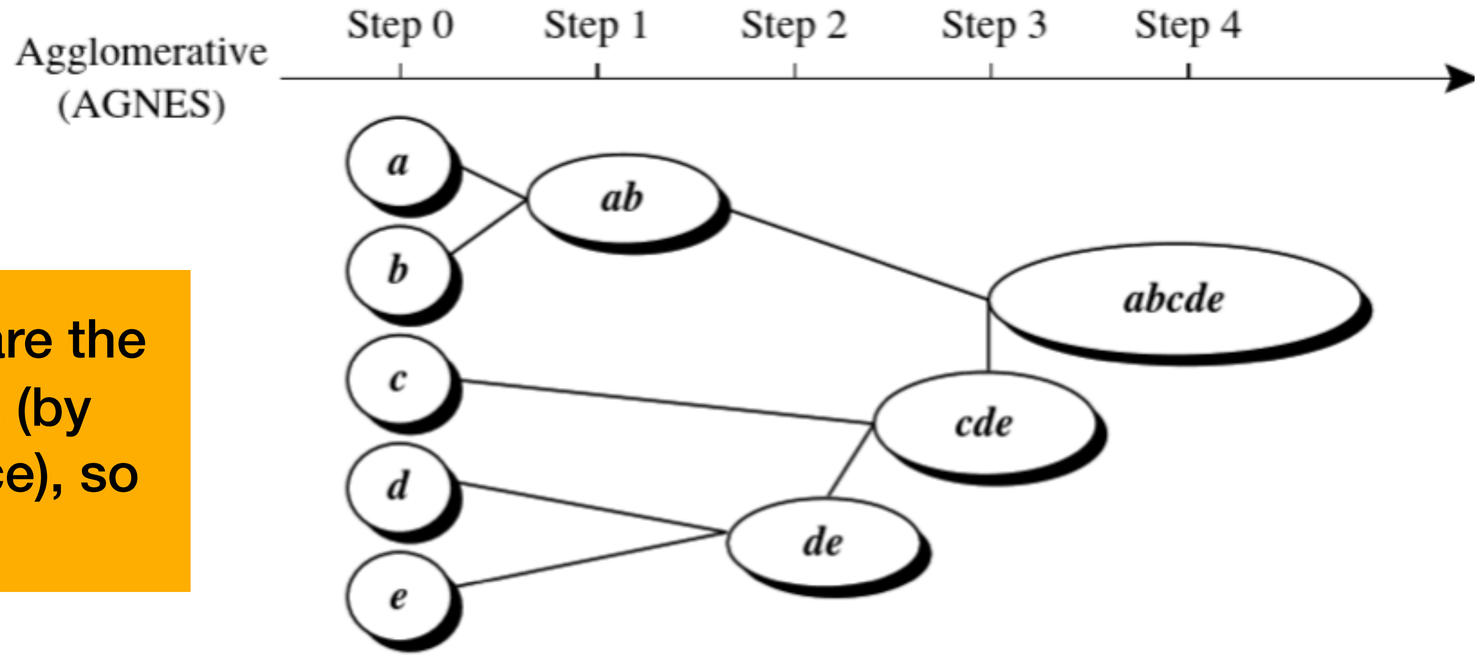
Kaufman, L. and Rousseeuw, P.J., 2009. Finding groups in data: an introduction to cluster analysis (Vol. 344). John Wiley & Sons.

Hierarchical Clustering

Agglomerative Clustering - Example

- AGNES (***AG****glomerative* ***NES****ting*) is an example of an agglomerative hierarchical clustering approach.

- Let X be a set of five data objects, $\{a, b, c, d, e\}$.



Step 3: c and de are the closest clusters (by Euclidean distance), so merge.

Han, J., Pei, J. and Kamber, M., 2012. *Data Mining: Concepts and Techniques*. Elsevier.

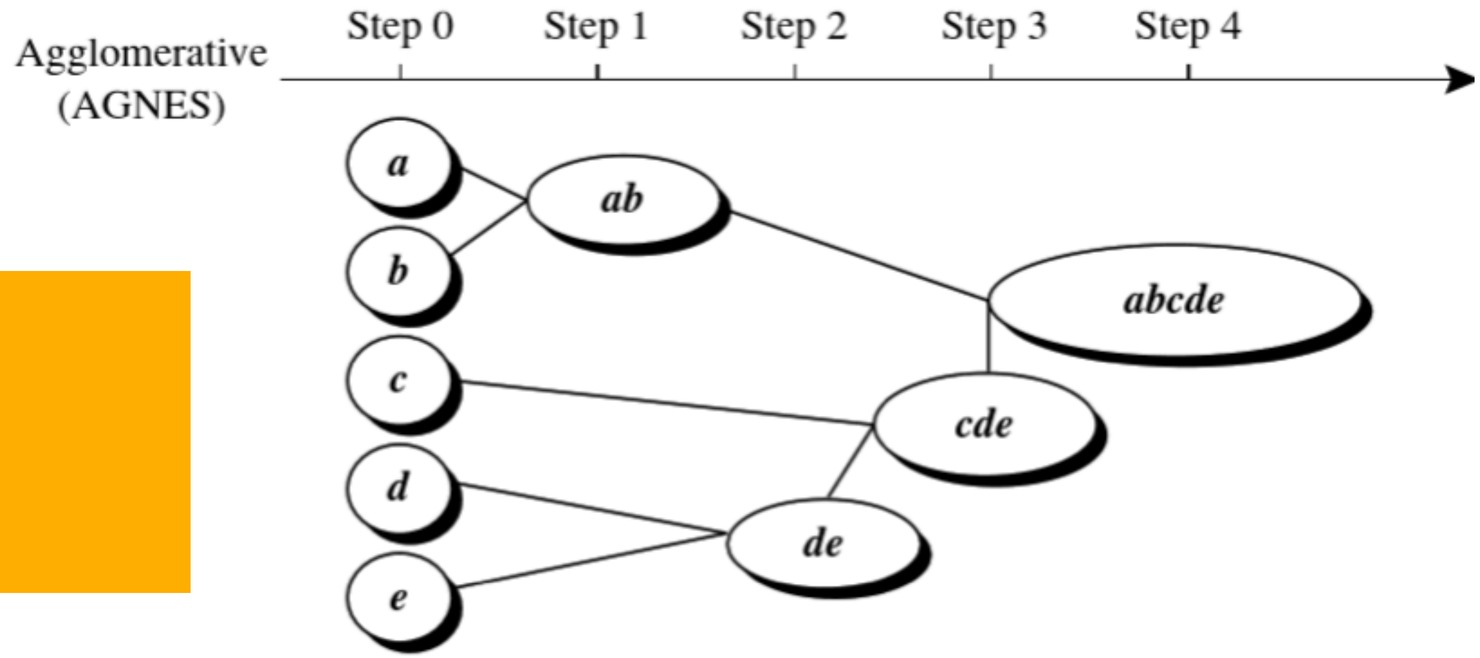
Kaufman, L. and Rousseeuw, P.J., 2009. Finding groups in data: an introduction to cluster analysis (Vol. 344). John Wiley & Sons.

Hierarchical Clustering

Agglomerative Clustering - Example

- AGNES (***AG****glomerative* ***NES****ting*) is an example of an agglomerative hierarchical clustering approach.

- Let X be a set of five data objects, $\{a, b, c, d, e\}$.



Etc.

Han, J., Pei, J. and Kamber, M., 2012. *Data Mining: Concepts and Techniques*. Elsevier.

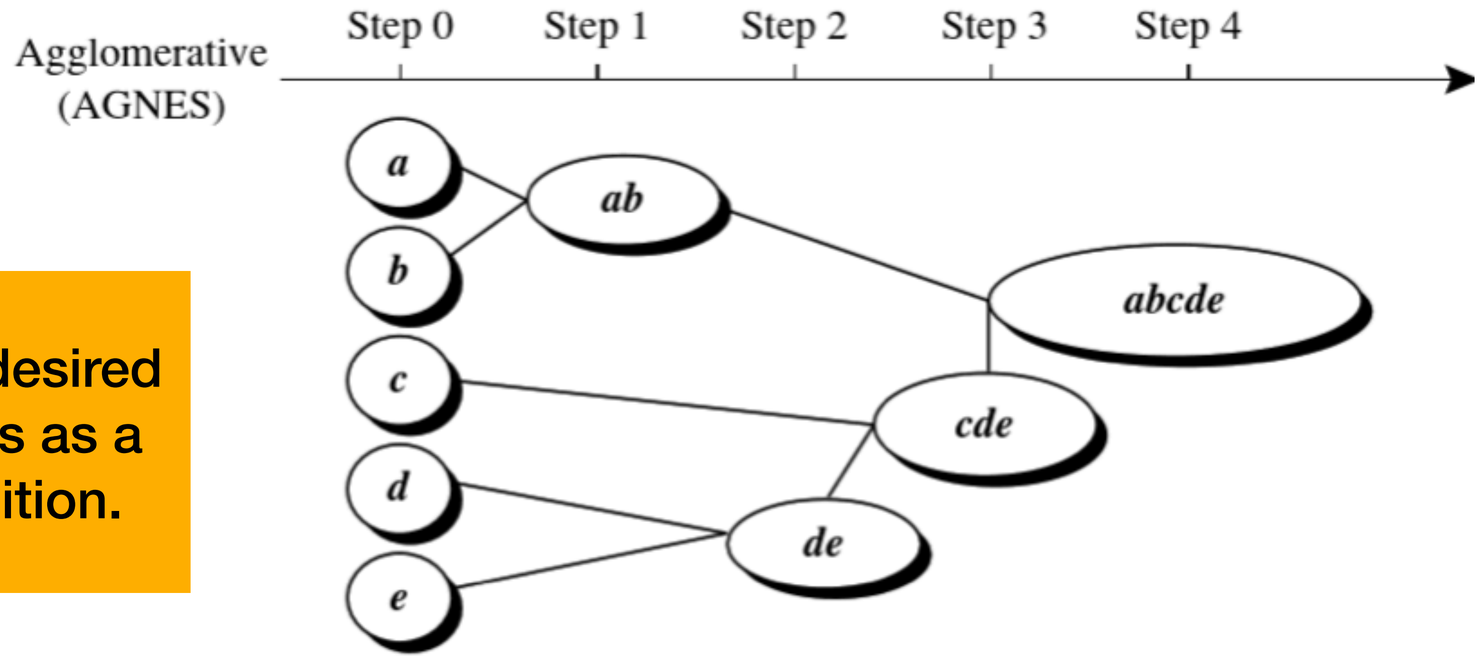
Kaufman, L. and Rousseeuw, P.J., 2009. Finding groups in data: an introduction to cluster analysis (Vol. 344). John Wiley & Sons.

Hierarchical Clustering

Agglomerative Clustering - Example

- AGNES (***AG****glomerative* ***NES****ting*) is an example of an agglomerative hierarchical clustering approach.

- Let X be a set of five data objects, $\{a, b, c, d, e\}$.



We can specify a desired number of clusters as a terminating condition.

Han, J., Pei, J. and Kamber, M., 2012. *Data Mining: Concepts and Techniques*. Elsevier.

Kaufman, L. and Rousseeuw, P.J., 2009. Finding groups in data: an introduction to cluster analysis (Vol. 344). John Wiley & Sons.

Hierarchical Clustering

Divisive Clustering

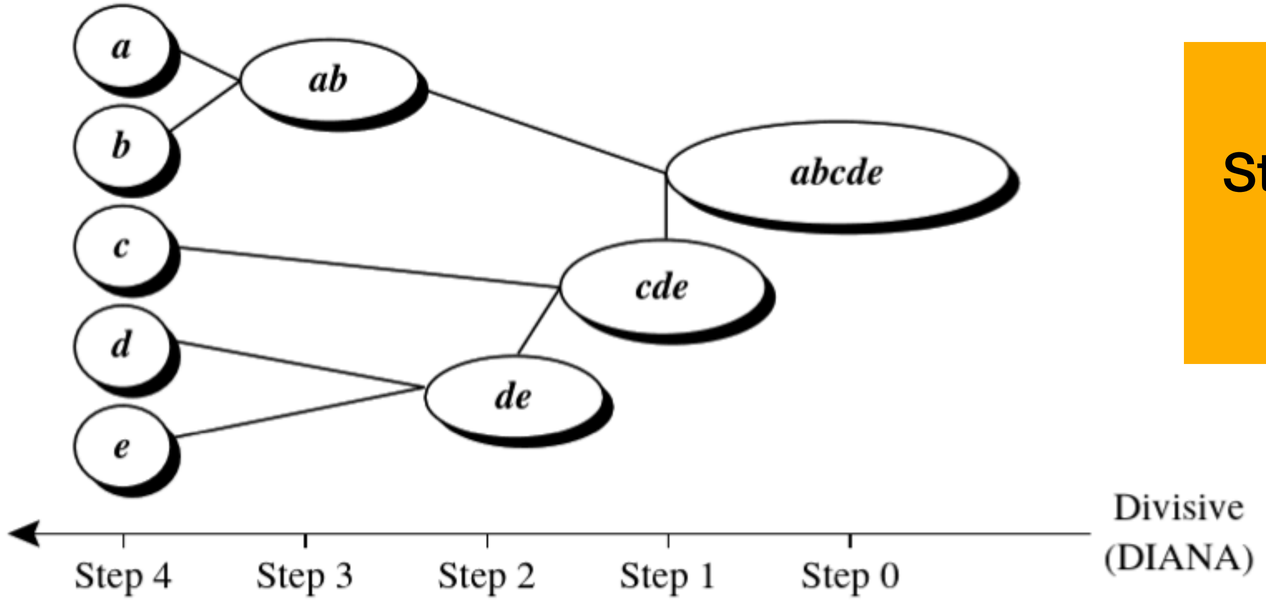
- **Divisive Approach** (also called a **Top-Down Approach**): Starts with all objects in the same cluster, and divides the clusters until each object is in its own cluster, or a terminating condition is met.
- **Splitting Step**: Split according to some principle, example in next few slides.

Hierarchical Clustering

Divisive Clustering - Example

- DIANA (***DI****visive* ***ANA****lysis*) is an example of a divisive hierarchical clustering approach.

- Let X be a set of five data objects, $\{a, b, c, d, e\}$.



Step 0: All objects are in the same cluster.

Kaufman, L. and Rousseeuw, P.J., 2009. Finding groups in data: an introduction to cluster analysis (Vol. 344). John Wiley & Sons.

Han, J., Pei, J. and Kamber, M., 2012. *Data Mining: Concepts and Techniques*. Elsevier.

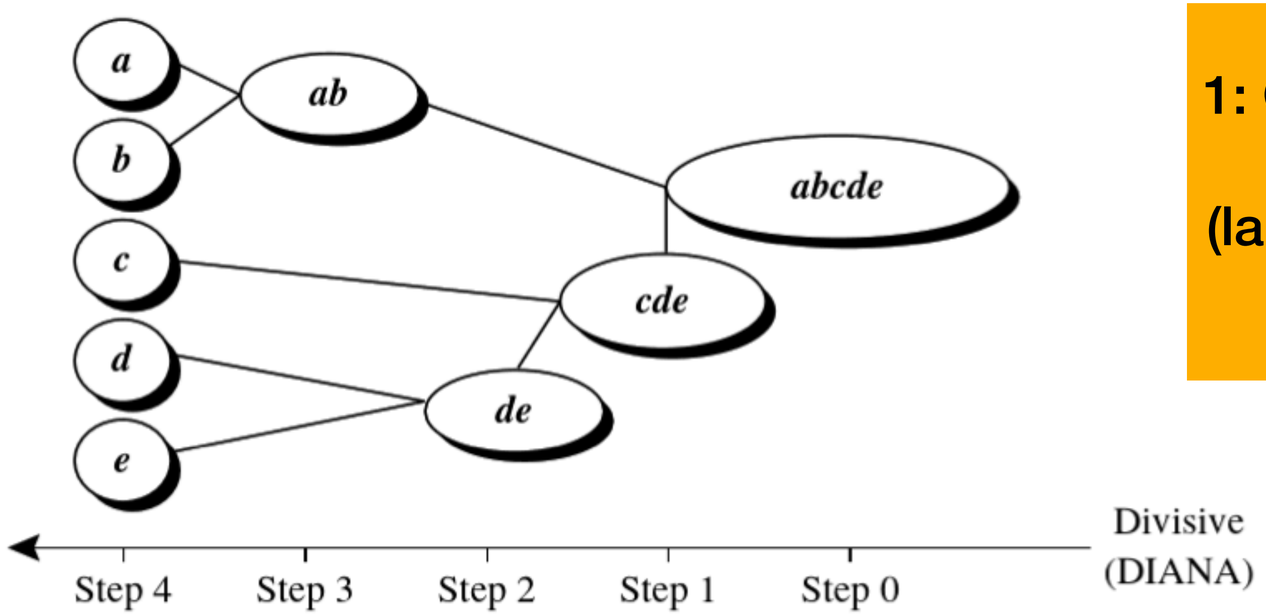
https://stat.ethz.ch/R-manual/R-patched/library/cluster/html/diana.html

Hierarchical Clustering

Divisive Clustering - Example

- DIANA (***DI****visive* ***ANA****lysis*) is an example of a divisive hierarchical clustering approach.

- Let X be a set of five data objects, $\{a, b, c, d, e\}$.



1: Choose the cluster with the largest *diameter* (largest distance between any two objects)

Kaufman, L. and Rousseeuw, P.J., 2009. Finding groups in data: an introduction to cluster analysis (Vol. 344). John Wiley & Sons.

Han, J., Pei, J. and Kamber, M., 2012. *Data Mining: Concepts and Techniques*. Elsevier.

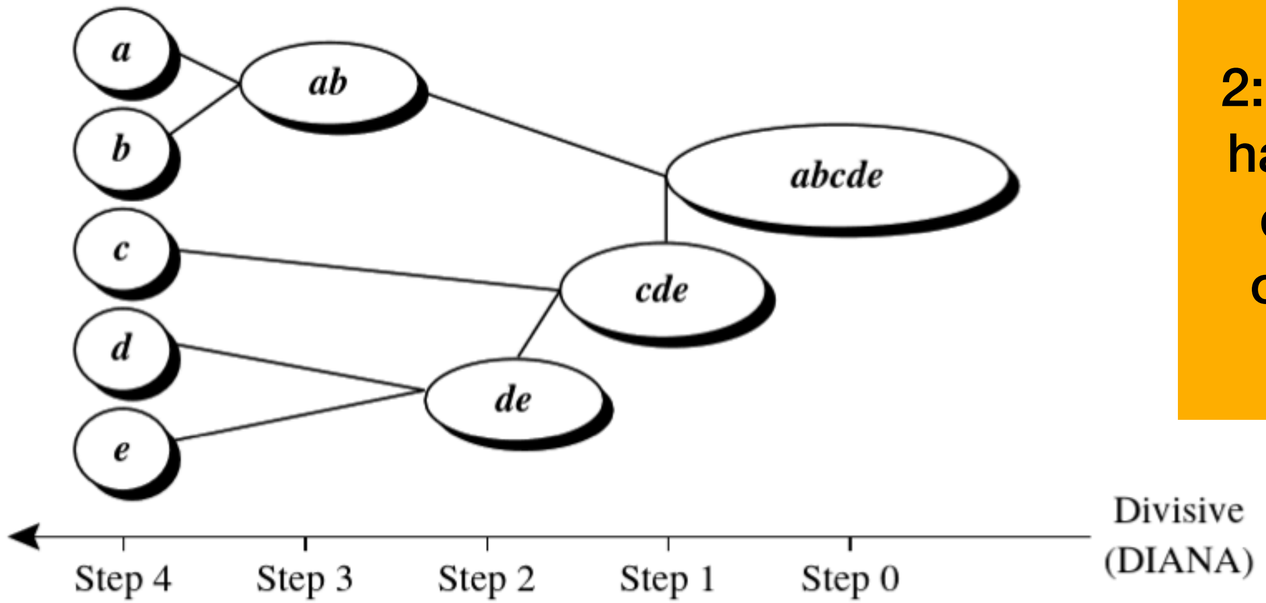
https://stat.ethz.ch/R-manual/R-patched/library/cluster/html/diana.html

Hierarchical Clustering

Divisive Clustering - Example

- DIANA (***DI****visive* ***ANA****lysis*) is an example of a divisive hierarchical clustering approach.

- Let X be a set of five data objects, $\{a, b, c, d, e\}$.



2: Find the object which has the largest average distance to the other objects of the cluster.

Kaufman, L. and Rousseeuw, P.J., 2009. Finding groups in data: an introduction to cluster analysis (Vol. 344). John Wiley & Sons.

Han, J., Pei, J. and Kamber, M., 2012. *Data Mining: Concepts and Techniques*. Elsevier.

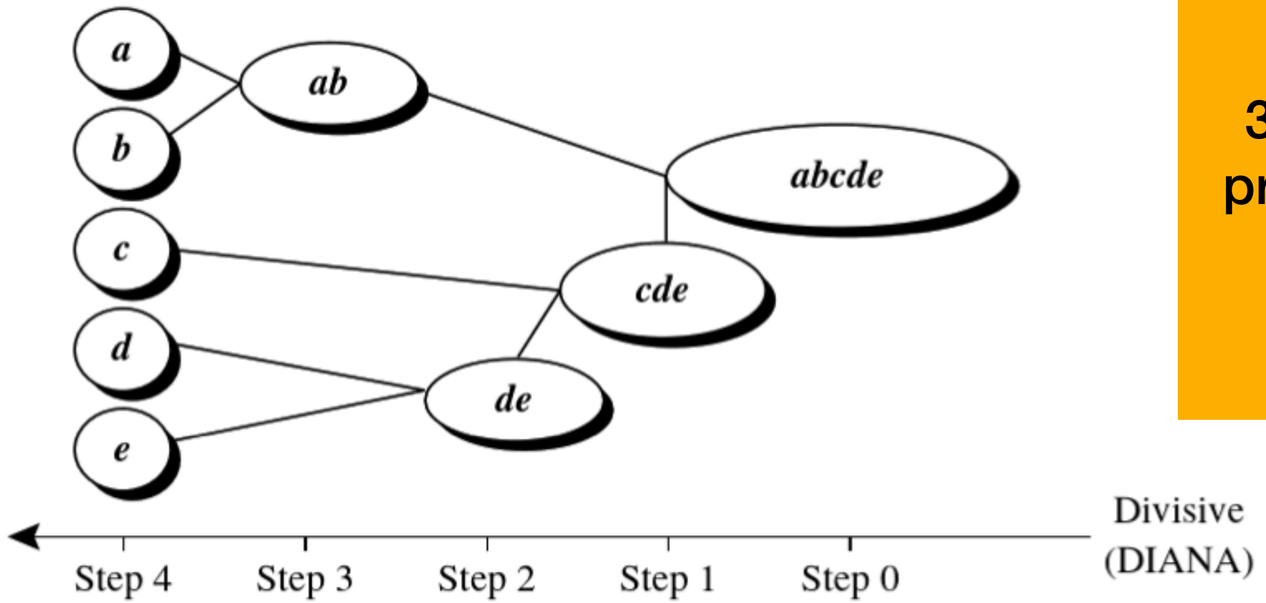
https://stat.ethz.ch/R-manual/R-patched/library/cluster/html/diana.html

Hierarchical Clustering

Divisive Clustering - Example

- DIANA (***DI****visive* ***ANA****lysis*) is an example of a divisive hierarchical clustering approach.

- Let X be a set of five data objects, $\{a, b, c, d, e\}$.



3: The object from the previous step initiates a *splinter group*.

Kaufman, L. and Rousseeuw, P.J., 2009. Finding groups in data: an introduction to cluster analysis (Vol. 344). John Wiley & Sons.

Han, J., Pei, J. and Kamber, M., 2012. *Data Mining: Concepts and Techniques*. Elsevier.

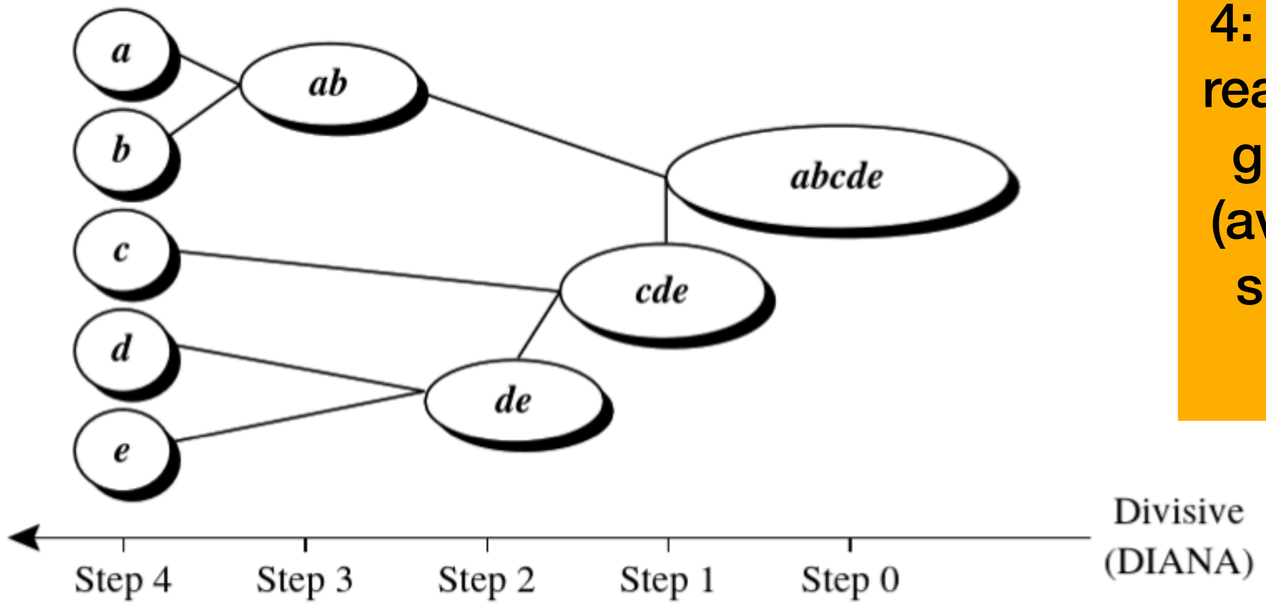
https://stat.ethz.ch/R-manual/R-patched/library/cluster/html/diana.html

Hierarchical Clustering

Divisive Clustering - Example

- DIANA (***DI****visive* ***ANA****lysis*) is an example of a divisive hierarchical clustering approach.

- Let X be a set of five data objects, $\{a, b, c, d, e\}$.



4: Remaining objects are reassigned to the splinter group if they are closer (average distance) to the splinter group than the original cluster.

Kaufman, L. and Rousseeuw, P.J., 2009. Finding groups in data: an introduction to cluster analysis (Vol. 344). John Wiley & Sons.

Han, J., Pei, J. and Kamber, M., 2012. *Data Mining: Concepts and Techniques*. Elsevier.

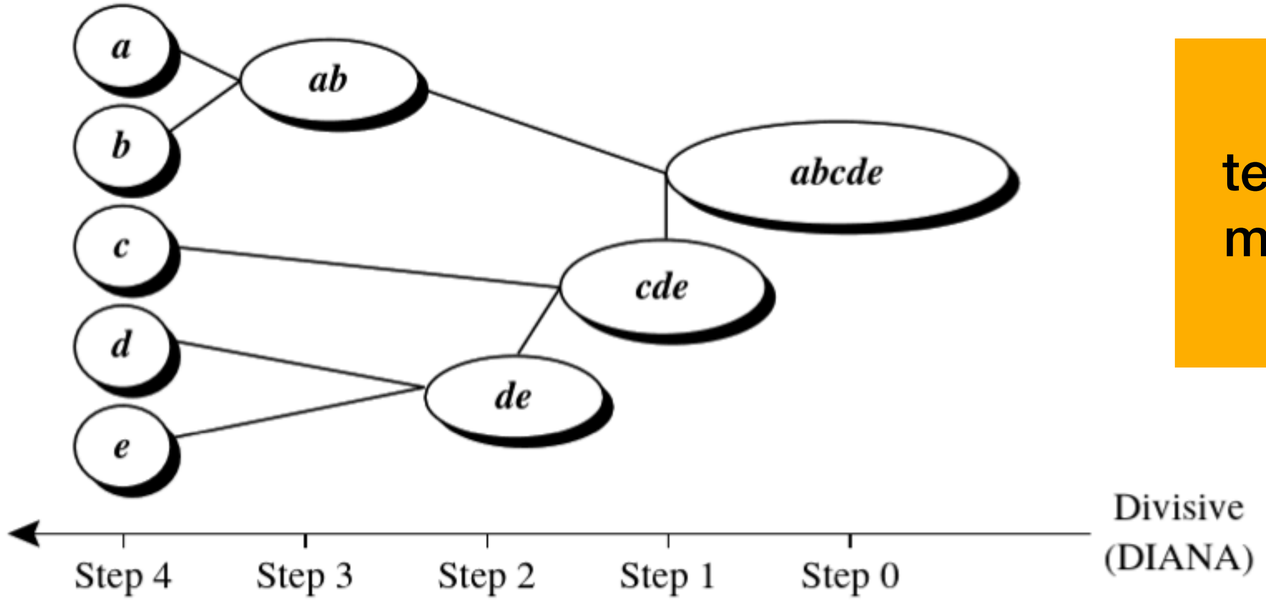
https://stat.ethz.ch/R-manual/R-patched/library/cluster/html/diana.html

Hierarchical Clustering

Divisive Clustering - Example

- DIANA (***DI****visive* ***ANA****lysis*) is an example of a divisive hierarchical clustering approach.

- Let X be a set of five data objects, $\{a, b, c, d, e\}$.



Repeat 1 to 4 until terminating condition is met or all objects are in separate clusters.

Kaufman, L. and Rousseeuw, P.J., 2009. Finding groups in data: an introduction to cluster analysis (Vol. 344). John Wiley & Sons.

Han, J., Pei, J. and Kamber, M., 2012. *Data Mining: Concepts and Techniques*. Elsevier.

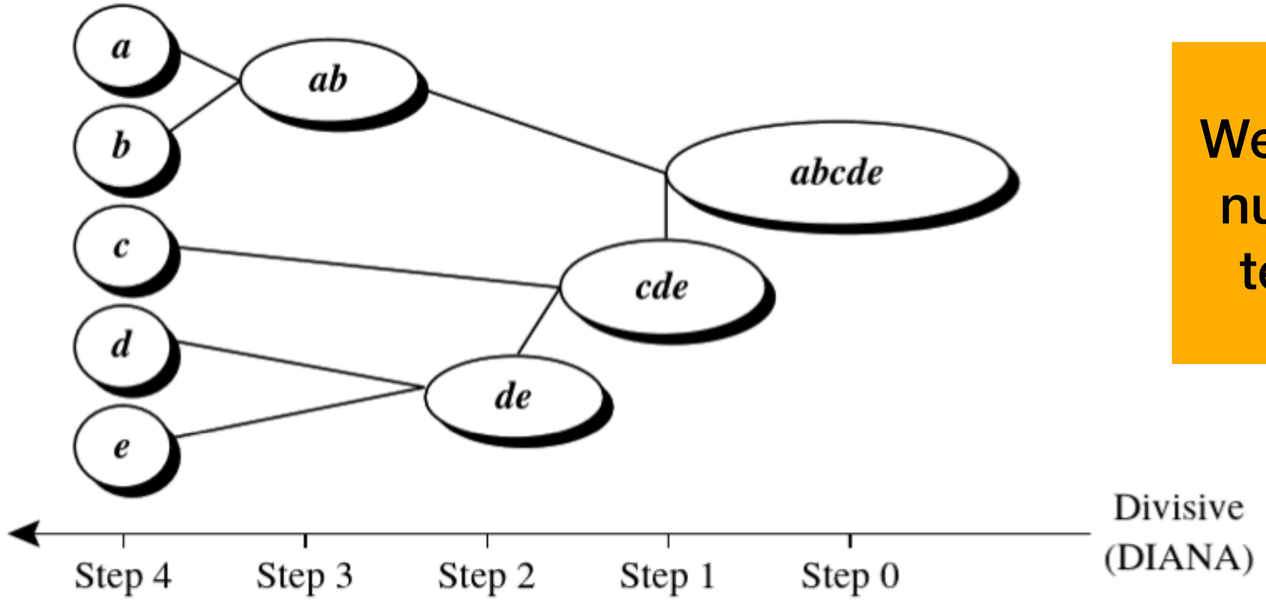
https://stat.ethz.ch/R-manual/R-patched/library/cluster/html/diana.html

Hierarchical Clustering

Divisive Clustering - Example

- DIANA (***DI****visive* ***ANA****lysis*) is an example of a divisive hierarchical clustering approach.

- Let X be a set of five data objects, $\{a, b, c, d, e\}$.



We can specify a desired number of clusters as a terminating condition.

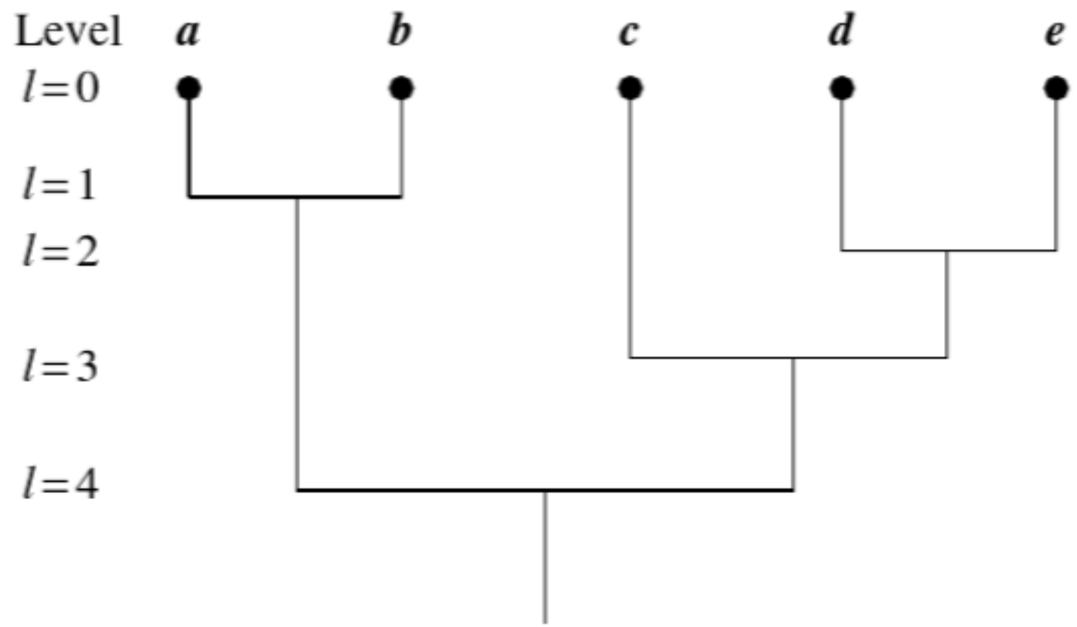
Kaufman, L. and Rousseeuw, P.J., 2009. Finding groups in data: an introduction to cluster analysis (Vol. 344). John Wiley & Sons.

Han, J., Pei, J. and Kamber, M., 2012. *Data Mining: Concepts and Techniques*. Elsevier.

https://stat.ethz.ch/R-manual/R-patched/library/cluster/html/diana.html

Dendrograms

- A ***Dendrogram*** is a tree-like structure that shows how objects are grouped together (agglomerative clustering) or partitioned (divisive clustering).
- Very common for visualisation of clusters.
- Can cut off at certain levels to choose a certain number of clusters (e.g. splitting at $l = 2$ would leave three clusters (ab, c, de)).



Han, J., Pei, J. and Kamber, M., 2012. *Data Mining: Concepts and Techniques*. Elsevier.

Distance Measures

- How can we measure the distance between clusters to find the closest pair?

  1. **Single-Link**: $d(C_1, C_2) = min\ D(c_1 \in C_1, c_2 \in C_2)$
  2. **Complete-Link**: $d(C_1, C_2) = max\ D(c_1 \in C_1, c_2 \in C_2)$
  3. **Average-Link**: $d(C_1, C_2) = avg\ D(c_1 \in C_1, c_2 \in C_2)$

  where $d(C_1, C_2)$ is the distance between clusters C_1 and C_2, and $D(c_1 \in C_1, c_2 \in C_2)$ is the distance between two objects, $c_1 \in C_1$ and $c_2 \in C_2$.

Distance Measures

- How can we measure the distance between clusters to find the closest pair?

  1. **Single-Link**: $d(C_1, C_2) = min\ D(c_1 \in C_1, c_2 \in C_2)$
  2. **Complete-Link**: $d(C_1, C_2) = max\ D(c_1 \in C_1, c_2 \in C_2)$
  3. **Average-Link**: $d(C_1, C_2) = avg\ D(c_1 \in C_1, c_2 \in C$

  where $d(C_1, C_2)$ is the distance between clusters C_1 and
  $D(c_1 \in C_1, c_2 \in C_2)$ is the distance between two objects, $c_1 \in C_1$ and
  $c_2 \in C_2$.

Distance is determined by a single pair of elements (one in each cluster).

Distance Measures

- How can we measure the distance between clusters to find the closest pair?

1. **Single-Link**: $d(C_1, C_2) = min\ D(c_1 \in C_1, c_2 \in C_2)$
2. **Complete-Link**: $d(C_1, C_2) = max\ D(c_1 \in C_1, c_2 \in$
3. **Average-Link**: $d(C_1, C_2) = avg\ D(c_1 \in C_1, c_2 \in C_2)$

Distance is determined by the average of all distances from pairs of elements (one in each cluster).

where $d(C_1, C_2)$ is the distance between clusters C_1 and C_2, and $D(c_1 \in C_1, c_2 \in C_2)$ is the distance between two objects, $c_1 \in C_1$ and $c_2 \in C_2$.

Hierarchical Clustering

Cost (Agglomerative Example)

- Hierarchical Clustering can be costly to implement:
  - Initially, must compute distance from every object to every other object.
  - Each merge requires a new computation of distances involving the merged clusters.
- **This limits scalability**: with only few hundred thousand points, the clustering could take days or months.
  - Need clustering methods that can scale to allow processing of large data sets.

Single Link Clustering in Weka

```
=== Run information ===
Scheme:weka.clusterers.HierarchicalClusterer -N 3 -L SINGLE -P -A
    "weka.core.EuclideanDistance -R first-last"
Relation:     iris
=== Model and evaluation on training set ===
Time taken to build model (full training data) : 0.17 seconds
Clustered Instances
0       49 ( 33%)
1        1 (  1%)
2      100 ( 67%)

Classes to Clusters:
  0  1  2  <-- assigned to cluster
 49  1  0 | Iris-setosa
  0  0 50 | Iris-versicolor
  0  0 50 | Iris-virginica
Cluster 0 <-- Iris-setosa
Cluster 1 <-- No class
Cluster 2 <-- Iris-versicolor

Incorrectly clustered instances : 51.0  34      %
```

Average Link Clustering in Weka

```
Scheme:weka.clusterers.HierarchicalClusterer -N 3 -L AVERAGE -P -A
    "weka.core.EuclideanDistance -R first-last"
Relation:     iris
Test mode:Classes to clusters evaluation on training data
=== Model and evaluation on training set ===
Time taken to build model (full training data) : 0.08 seconds

Clustered Instances
0       50 ( 33%)
1       67 ( 45%)
2       33 ( 22%)

Classes to Clusters:
  0  1  2  <-- assigned to cluster
 50  0  0 | Iris-setosa
  0 50  0 | Iris-versicolor
  0 17 33 | Iris-virginica

Cluster 0 <-- Iris-setosa
Cluster 1 <-- Iris-versicolor
Cluster 2 <-- Iris-virginica
Incorrectly clustered instances : 17.0   11.3333 %
```

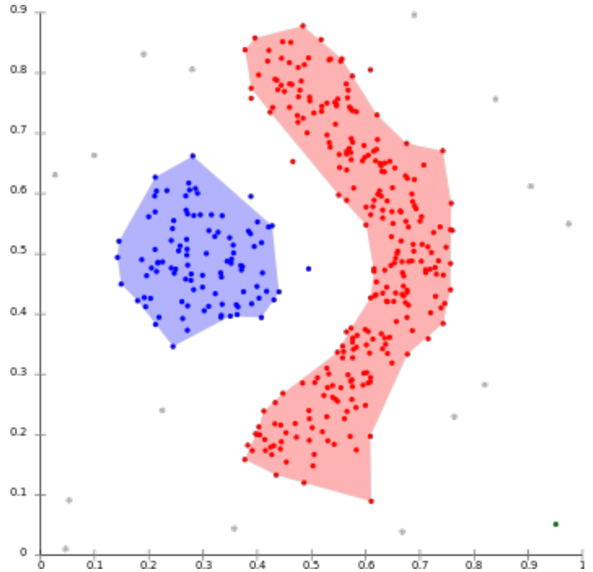
Acknowledgments

- Han, J., Pei, J. and Kamber, M., 2012. *Data Mining: Concepts and Techniques*. Elsevier.
- Graham Cormode [Warwick, CS910]
- Florin Ciucu [Warwick, CS430/CS910]

Part D: Clustering Algorithms 3 & Clustering Evaluation

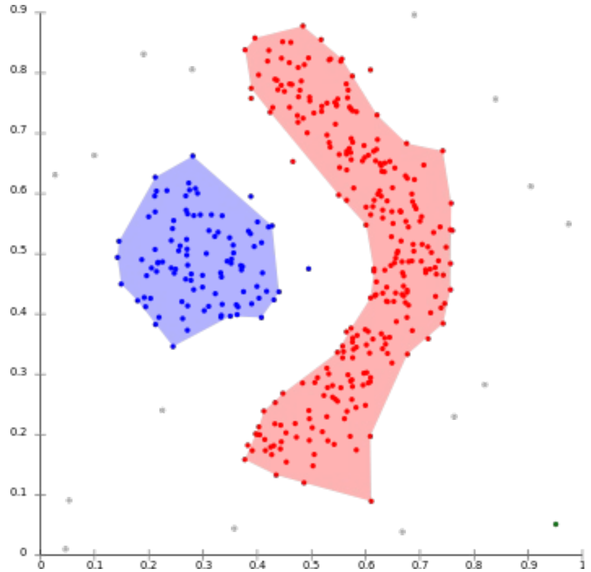
Density-Based Clustering Methods

- Many other clustering methods cluster based on the distance between objects.
  - Leads to spherical clusters, struggles with arbitrary shaped clusters.
- General idea: Continue growing a cluster as long as the nearby neighbourhood is greater than some threshold.
  - Good for filtering out outliers!



Density-Based Clustering Methods

- Density-based methods seek clusters that are locally dense
  - Dense meaning there are many data points per unit volume.
  - Separated by sparse regions of space.
  - Potentially allows finding complex cluster shapes.
  - May be more robust to noise
- Use a number of density parameters to determine when to extend a cluster.



DBSCAN

- DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is an example of a density-based clustering method.
- It finds *core objects* (objects with dense neighbourhoods).
- Uses two parameters:

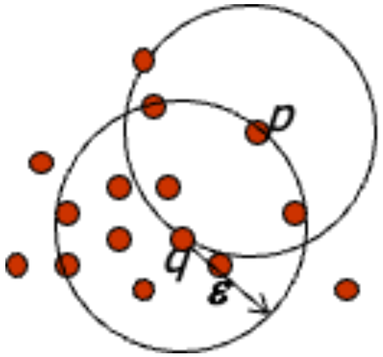
    1. ϵ (epsilon/eps): the radius to search for neighbours, based on a distance function $d()$.
    2. $minPts$: the minimum number of points in the neighbourhood of a point to extend the current cluster.

DBSCAN

- The ϵ-neighbourhood (N_ϵ) of an object $\mathbf{o}$ is the space within a radius ϵ centred at object $\mathbf{o}$.
  - $N_\epsilon(\mathbf{q}) = \{\mathbf{p} \in \text{data} \,|\, d(\mathbf{p}, \mathbf{q}) \leq \epsilon\}$
- The density of a neighbourhood is the number of objects within the neighbourhood.
- An object is a *core object* if at least $minPts$ objects are within its ϵ -neighbourhood.

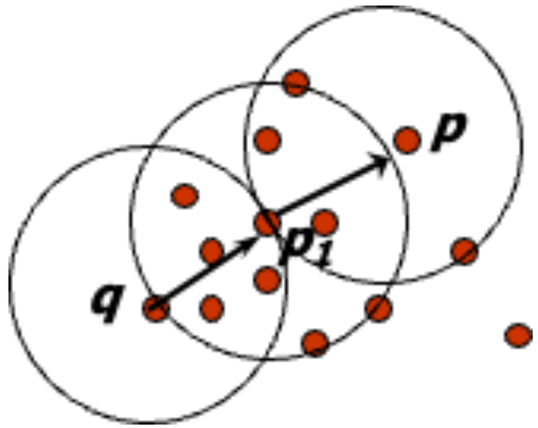
DBSCAN

- **Directly Density-Reachable**: For a **core** object $\mathbf{q}$ and an object $\mathbf{p}$, $\mathbf{p}$ is directly density-reachable from $\mathbf{q}$ (under ϵ, $minPts$) if $\mathbf{p}$ is within the ϵ-neighbourhood of $\mathbf{q}$.
  - Or, $\mathbf{p} \in N_\epsilon(\mathbf{q})$ and $|N_\epsilon(\mathbf{q})| \geq minPts$



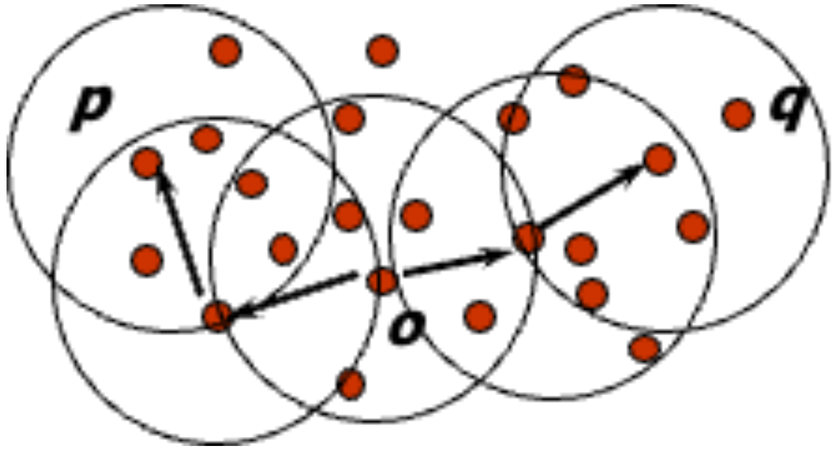
MinPts = 8

- **Density-Reachable**: An object $\mathbf{p}$ is density-reachable from an object $\mathbf{q}$ (under ϵ, $minPts$) if there exists a chain of objects $\mathbf{p_1}, \mathbf{p_2}, \ldots, \mathbf{p_n}$, such that $\mathbf{p_1} = \mathbf{q}$, $\mathbf{p_n} = \mathbf{p}$ and $\mathbf{p_{i+1}}$ is directly density-reachable from $\mathbf{p_i}$.
  - Not symmetric!

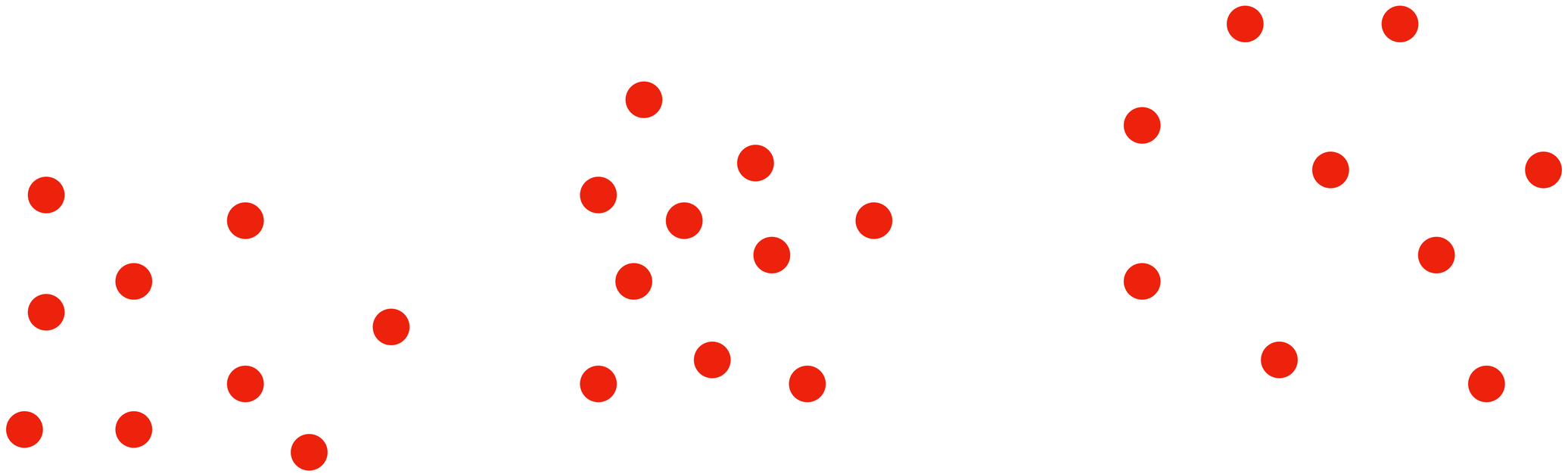


DBSCAN

- **Density-Connected**: An object $\mathbf{p}$ is density-connected to an object $\mathbf{q}$ (under ϵ, $minPts$) if there is an object $\mathbf{o}$ such that both $\mathbf{p}$ and $\mathbf{q}$ are density-reachable from $\mathbf{o}$.
  - Symmetric!



DBSCAN

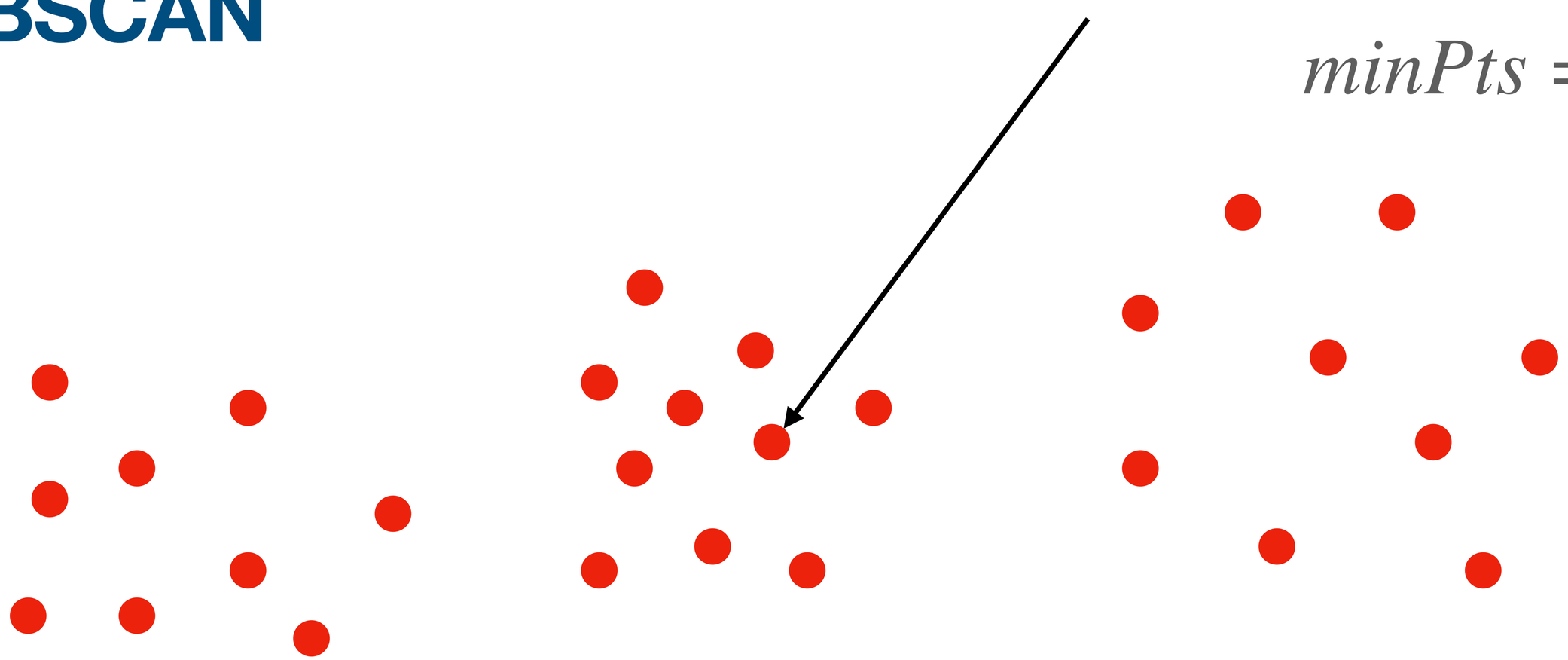


Start with this data.

DBSCAN

$\epsilon = 5$

$minPts = 3$

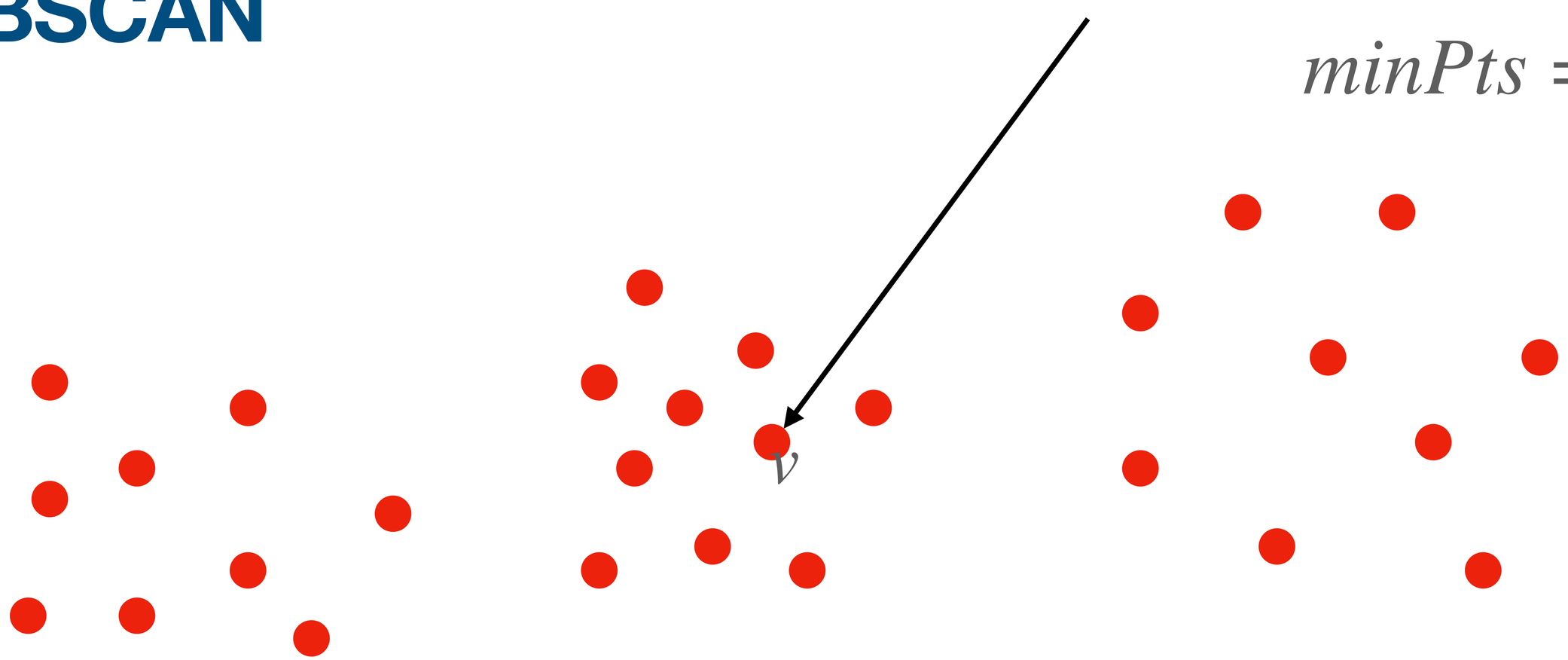


Select random object (lets call it **p**).

DBSCAN

$\epsilon = 5$

$minPts = 3$

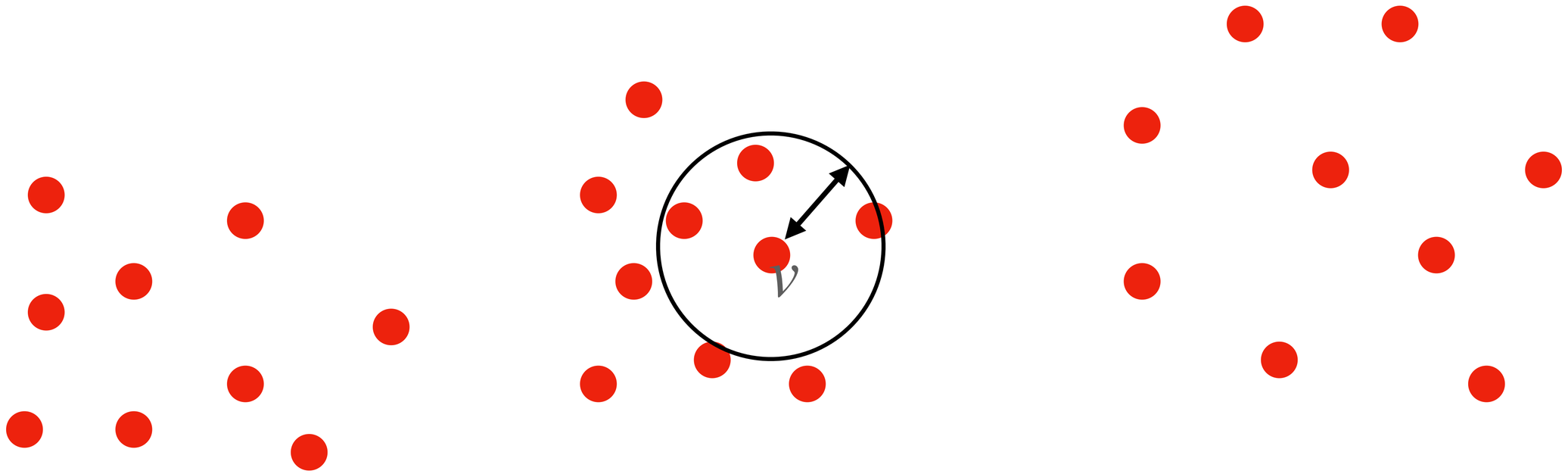


Mark **p** as visited.

DBSCAN

$\epsilon = 5$

$minPts = 3$



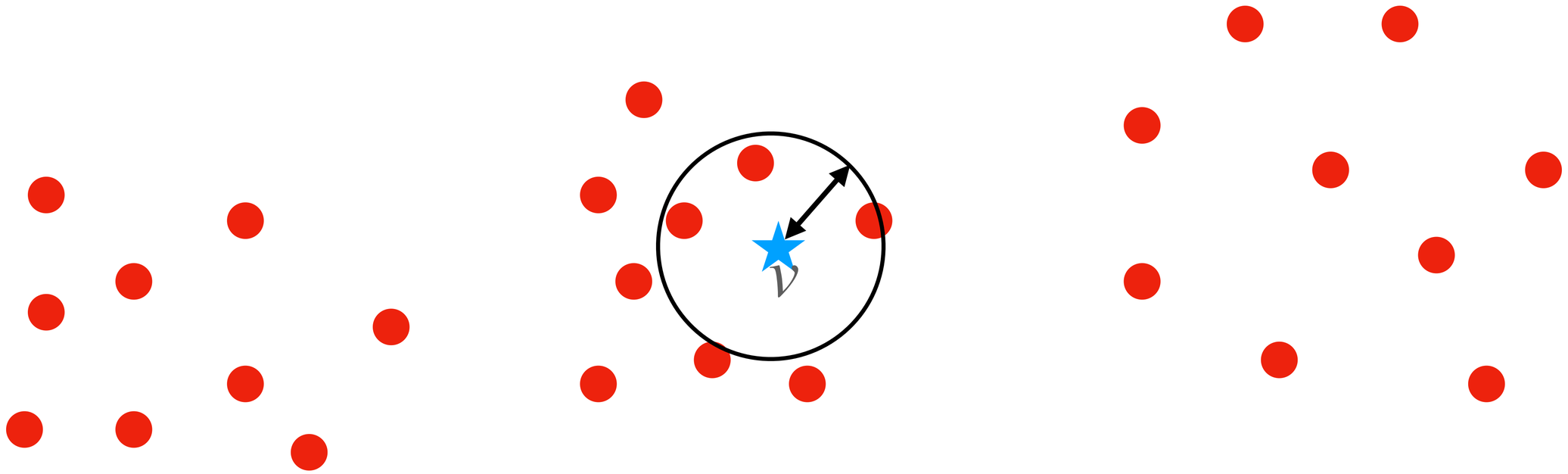
Are 3 objects (including original) within a distance of 5 of $\mathbf{p}$? Yes

(Otherwise we would mark $\mathbf{p}$ as noise and select a different unvisited random object)

DBSCAN

$\epsilon = 5$

$minPts = 3$

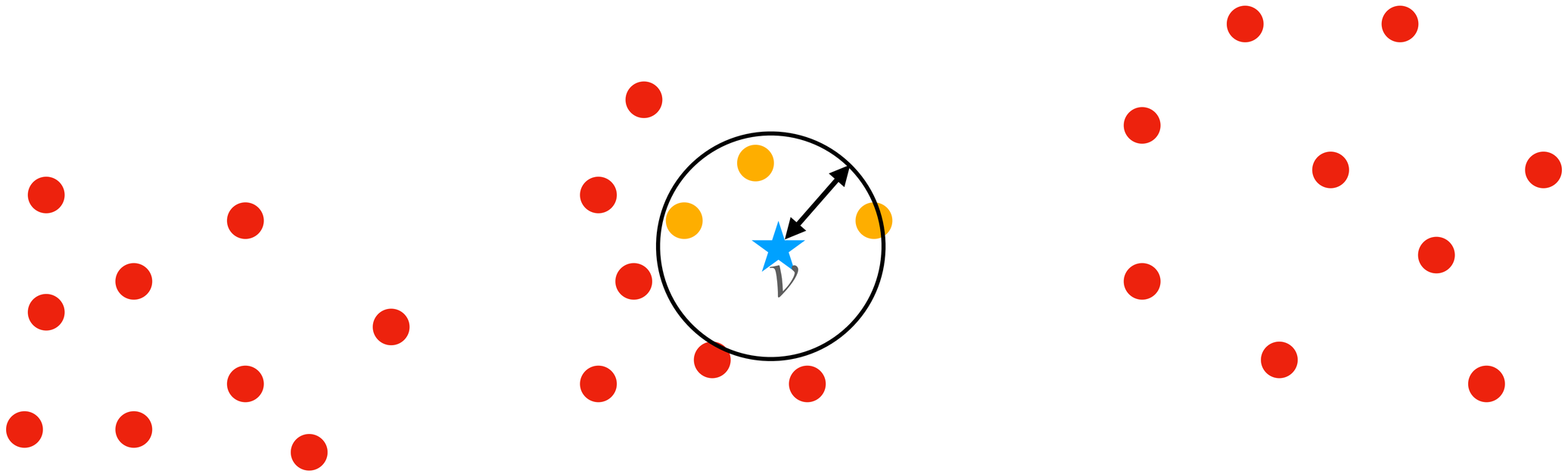


Create a new cluster C, and add $\mathbf{p}$ to it.

DBSCAN

$\epsilon = 5$

$minPts = 3$

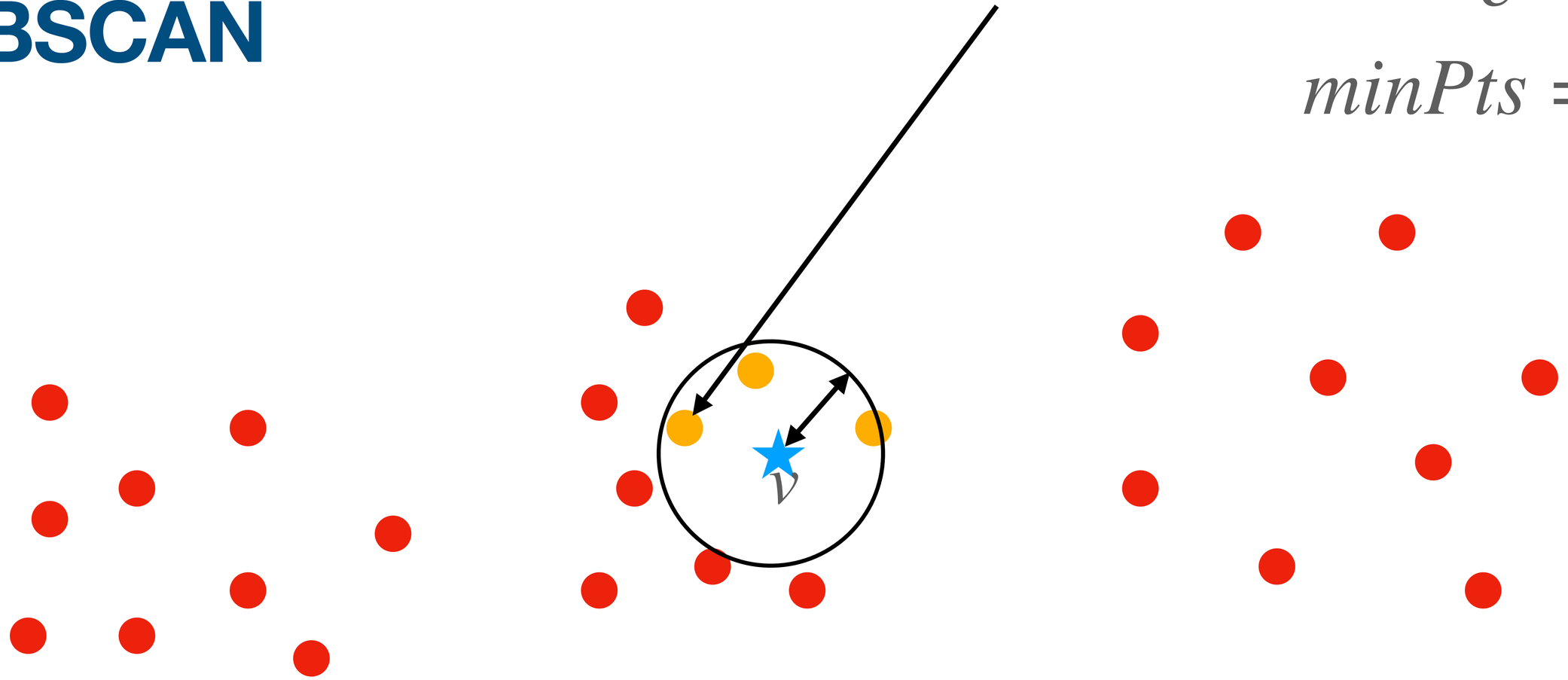


Let N be the set of objects in the ϵ-neighbourhood of $\mathbf{p}$.

DBSCAN

$\epsilon = 5$

$minPts = 3$

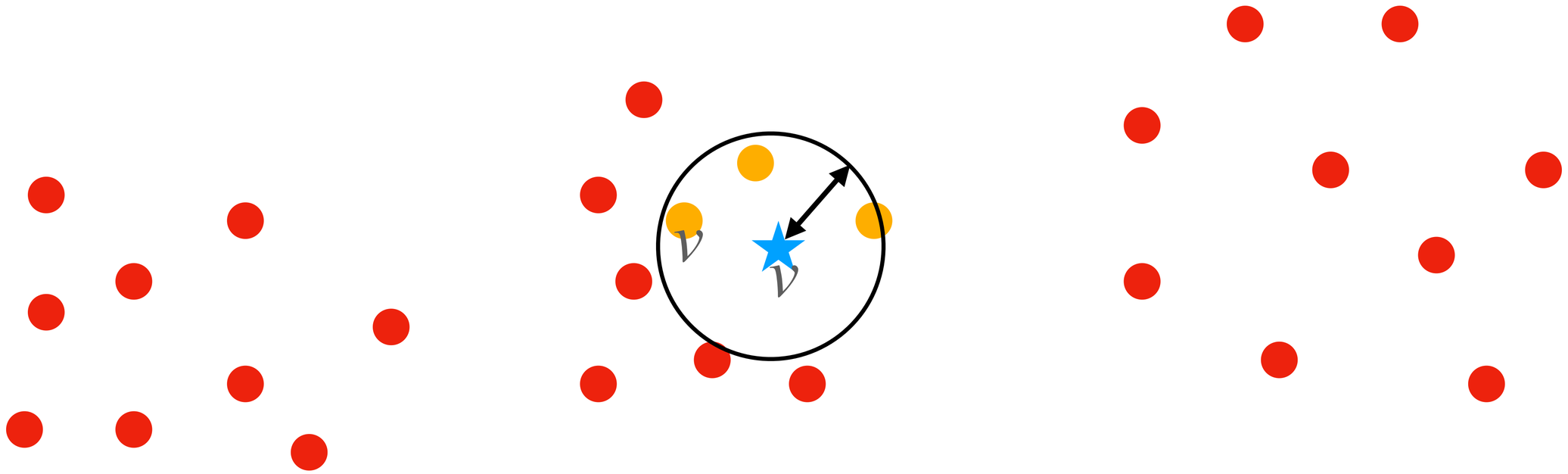


For each object $\mathbf{p}'$ in N...
(let's focus on the left one for now...)

DBSCAN

$$\epsilon = 5$$

$$minPts = 3$$

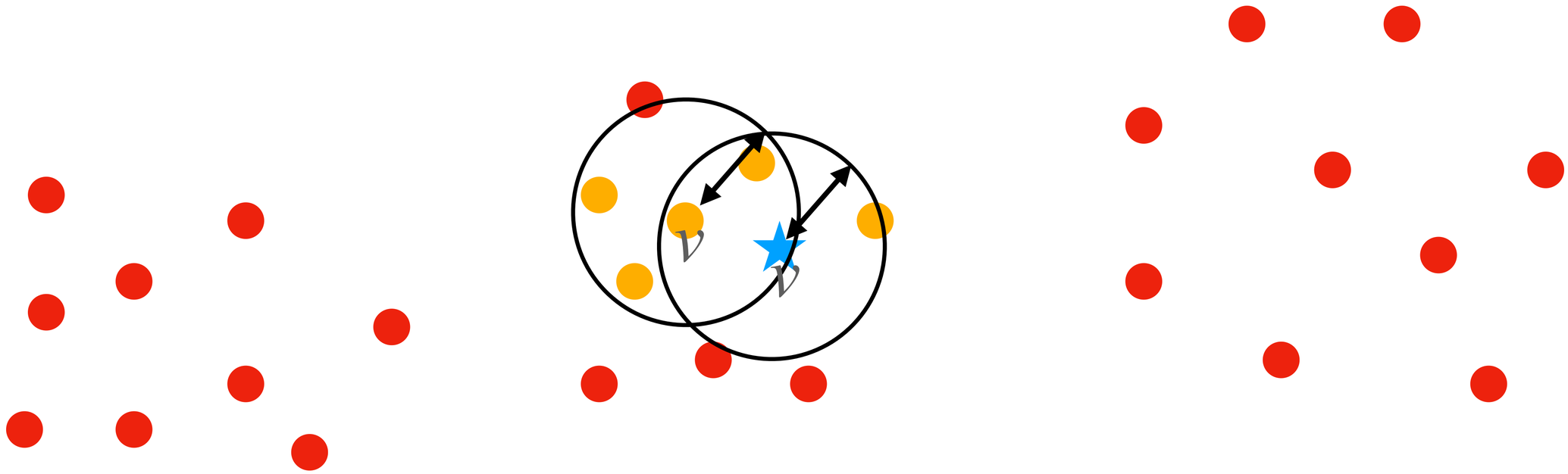


If $\mathbf{p}'$ is unvisited, 1) mark it as visited.

DBSCAN

$\epsilon = 5$

$minPts = 3$

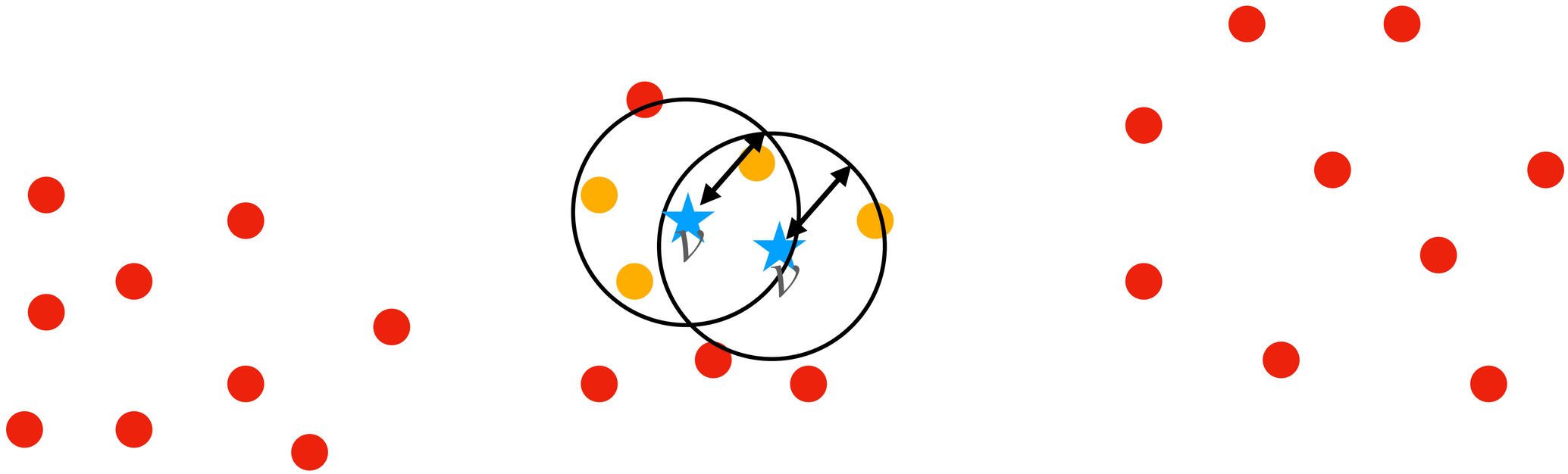


If $\mathbf{p}'$ is unvisited, 2) if the ϵ -neighbourhood of $\mathbf{p}'$ has at least $minPts$ objects, add those objects to N.

DBSCAN

$\epsilon = 5$

$minPts = 3$

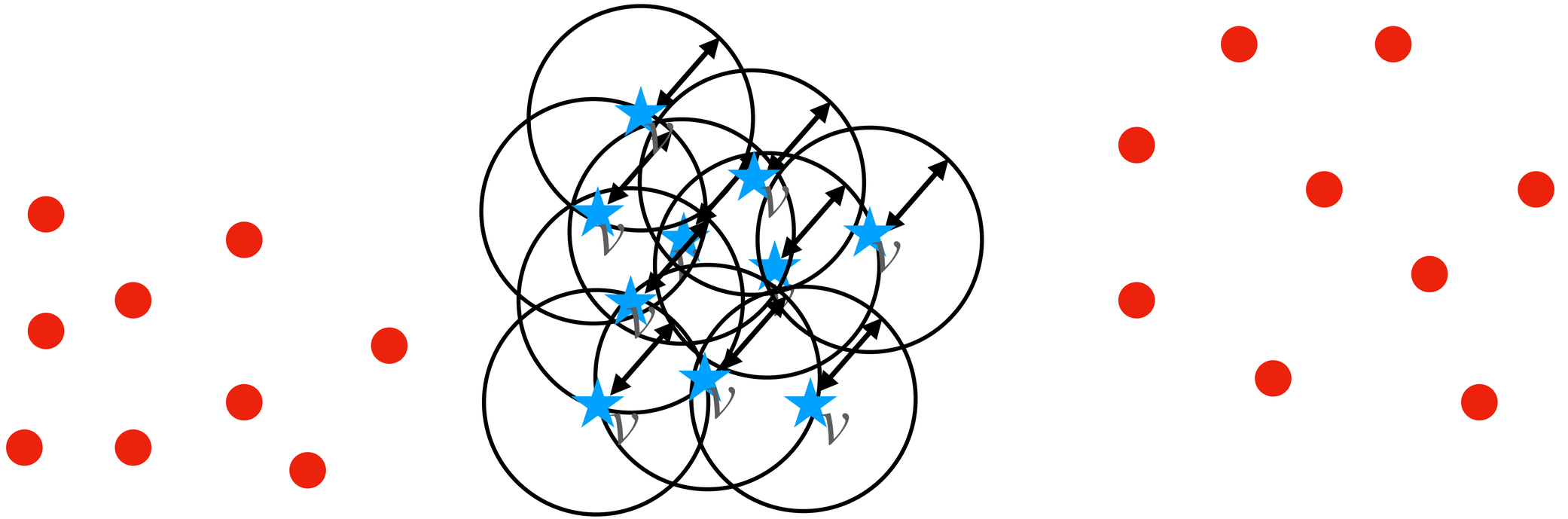


If $\mathbf{p}'$ is not yet a member of any cluster, add it to C.

DBSCAN

$\epsilon = 5$

$minPts = 3$

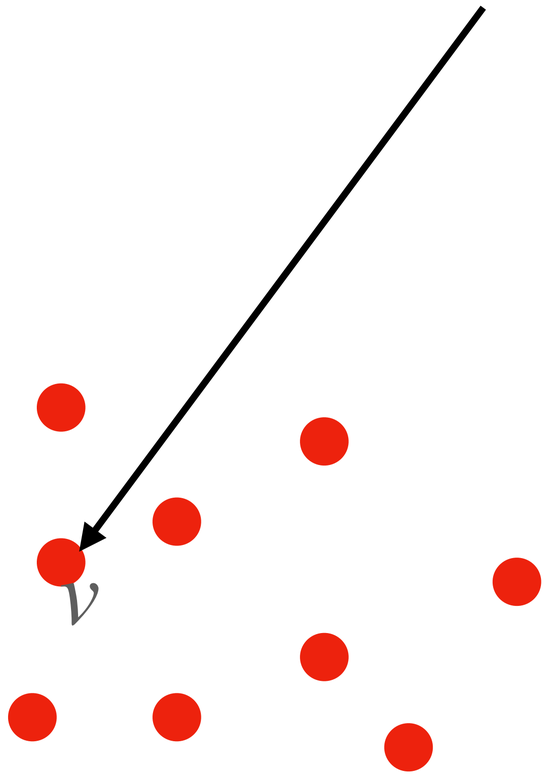


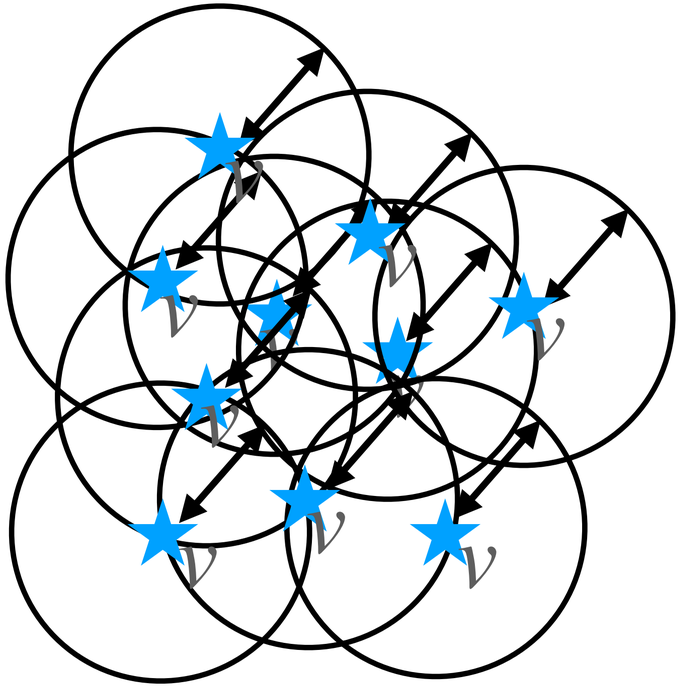
Continue for all objects in N...

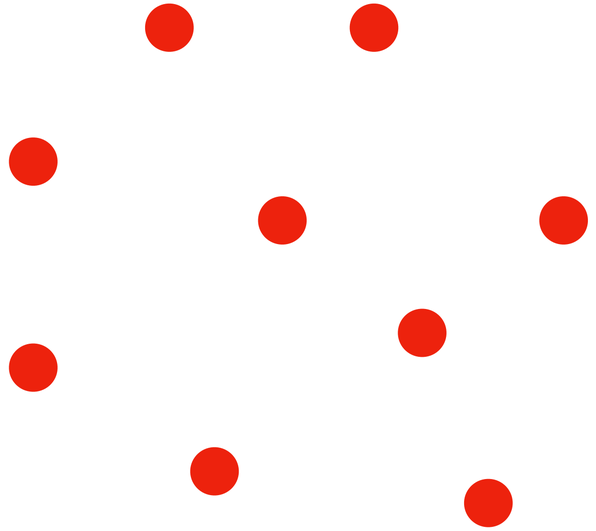
DBSCAN

$\epsilon = 5$

$minPts = 3$







Then randomly choose another object, mark it visited, and repeat…

DBSCAN

Algorithm

Algorithm: DBSCAN: a density-based clustering algorithm.

Input:

- D: a data set containing n objects,
- ϵ: the radius parameter, and
- *MinPts*: the neighborhood density threshold.

Output: A set of density-based clusters.

Method:

```
(1)  mark all objects as unvisited;
(2)  do
(3)      randomly select an unvisited object p;
(4)      mark p as visited;
(5)      if the ε-neighborhood of p has at least MinPts objects
(6)          create a new cluster C, and add p to C;
(7)          let N be the set of objects in the ε-neighborhood of p;
(8)          for each point p' in N
(9)              if p' is unvisited
(10)                 mark p' as visited;
(11)                 if the ε-neighborhood of p' has at least MinPts points,
                         add those points to N;
(12)             if p' is not yet a member of any cluster, add p' to C;
(13)         end for
(14)         output C;
(15)     else mark p as noise;
(16) until no object is unvisited;
```

Han, J., Pei, J. and Kamber, M., 2012. *Data Mining: Concepts and Techniques*. Elsevier.

DBSCAN

- Given a data set D, a subset $C \subseteq D$ is a cluster if:

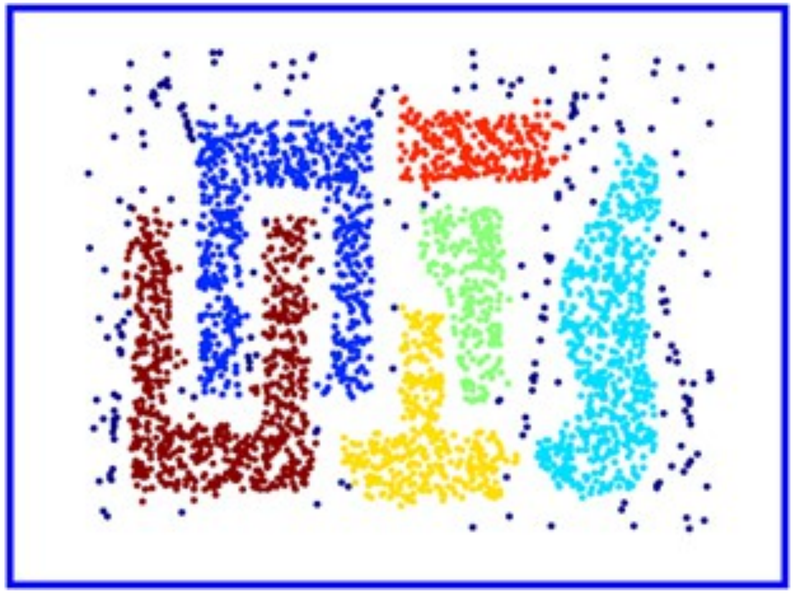
  1. For any two objects $\mathbf{o_1}, \mathbf{o_2} \in C$, $\mathbf{o_1}$ and $\mathbf{o_2}$ are density-connected, and

  2. There does not exist an object $\mathbf{o} \in C$ and another object $\mathbf{o}' \in (D - C)$ such that $\mathbf{o}$ and $\mathbf{o}'$ are density-connected.

DBSCAN

Advantages

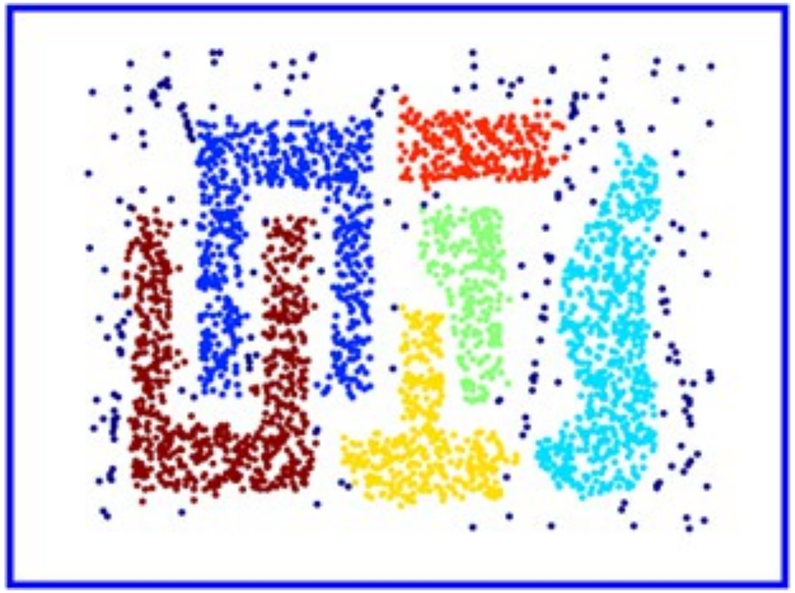
- Advantages:
  - Number of clusters is not a parameter.
  - Can find unusually shaped clusters.
  - Tolerant of noise.



DBSCAN

Disadvantages

- Disadvantages:
  - Setting parameters ϵ and $minPts$ may not be intuitive.
  - Can be slow to execute, in higher dimensions
  - Does not adapt to local variations in density.



DBSCAN Evaluation in Weka

```
=== Run information ===
Scheme:weka.clusterers.DBSCAN -E 0.12 -M 5 -I
    weka.clusterers.forOPTICSAndDBScan.Databases.SequentialDatabase -D
    weka.clusterers.forOPTICSAndDBScan.DataObjects.EuclideanDataObject
Relation:     iris
Test mode:Classes to clusters evaluation on training data
DBSCAN clustering results
==========================================================================
===============
Clustered DataObjects: 150
Number of attributes: 4
Epsilon: 0.12; minPoints: 5
Index: weka.clusterers.forOPTICSAndDBScan.Databases.SequentialDatabase
Distance-type:
weka.clusterers.forOPTICSAndDBScan.DataObjects.EuclideanDataObject
Number of generated clusters: 3
Elapsed time: .02
Time taken to build model (full training data) : 0.02 seconds
```

Note: experimented with many parameter choices to get $\epsilon = 0.12$, $minPts = 5$.

DBSCAN Evaluation in Weka

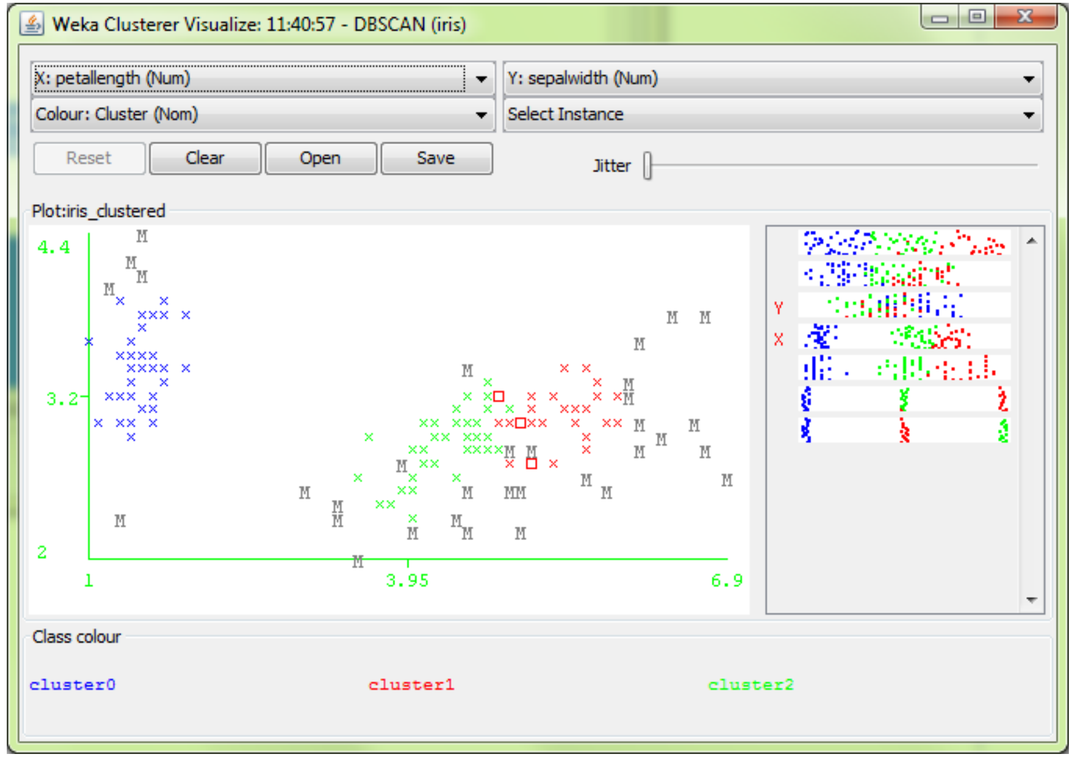
```
=== Model and evaluation on training set ===
Clustered Instances

0   45 ( 39%)

1   31 ( 27%)

2   38 ( 33%)

Unclustered instances : 36
Class attribute: class
Classes to Clusters:
  0  1  2  <-- assigned to cluster
 45  0  0 | Iris-setosa
  0  3 37 | Iris-versicolor
  0 28  1 | Iris-virginica
Cluster 0 <-- Iris-setosa
Cluster 1 <-- Iris-virginica
Cluster 2 <-- Iris-versicolor
Incorrectly clustered instances : 4.0     2.6667 %
```



Expectation Maximisation

- Expectation Maximization (EM) is more general form of k-means
  - Assume that the data is generated by some particular distribution
  - E.g., by k Gaussian distributions with unknown mean and variance
- Expectation Maximization (EM) looks for parameters of the distribution that agree best with the data.

- Also proceeds by repeating an alternating procedure:
  - Given current estimated distribution, compute likelihood for each data point being in each cluster
    $P[x_i \text{ in cluster } C_j \mid \text{ set of cluster parameters } C_1 \ldots C_k]$
  - From likelihoods, data and clusters, recompute parameters of distributions
  - Compute parameters of cluster
    $C_j \mid P[x_i \text{ in } C_1], P[x_i \text{ in } C_2]$ ...
  - Repeat until result stabilises or after sufficient iterations

Expectation Maximisation in Weka

```
Scheme:weka.clusterers.EM -I 100 -N 3 -M 1.0E-6 -S 100
Relation:     iris
Test mode:Classes to clusters evaluation on training data
=== Model and evaluation on training set ===
Number of clusters: 3
                Cluster
Attribute            0       1       2
                  (0.41)  (0.33)  (0.25)
=======================================
sepallength
  mean           5.9275   5.006  6.8085
  std. dev.      0.4817  0.3489  0.5339
sepalwidth
  mean           2.7503   3.418  3.0709
  std. dev.      0.2956  0.3772  0.2867
petallength
  mean           4.4057   1.464  5.7233
  std. dev.      0.5254  0.1718  0.4991
petalwidth
  mean           1.4131   0.244  2.1055
  std. dev.      0.2627  0.1061  0.2456
Time taken to build model (full training data) : 0.36 seconds
```

Expectation Maximisation in Weka

```
=== Model and evaluation on training set ===

Clustered Instances
0       64 ( 43%)
1       50 ( 33%)
2       36 ( 24%)

Log likelihood: -2.055

Class attribute: class
Classes to Clusters:
  0  1  2  <-- assigned to cluster
  0 50  0 | Iris-setosa
 50  0  0 | Iris-versicolor
 14  0 36 | Iris-virginica

Cluster 0 <-- Iris-versicolor
Cluster 1 <-- Iris-setosa
Cluster 2 <-- Iris-virginica

Incorrectly clustered instances : 14.0   9.3333 %
```

Hard Clustering vs. Soft Clustering

- So far we have discussed clustering algorithms where each object can only be assigned to one cluster.
- Consider the following situation:
  - We wish to cluster University of Warwick blogs, where each cluster represents a module (CS430, CS910, CS909, CS917, CS118, …)
  - A student/staff member may write a blog about multiple modules - this blog relates to multiple clusters.
- Idea: Could a clustering algorithm allow us to assign blogs to clusters, whilst also carrying a **weight** representing the membership?

Hard Clustering vs. Soft Clustering

- **Hard Clustering**: Each data object is assigned to one (and only one!) cluster.
  - E.g.: Car is a Toyota or Honda, but cannot be both!
- **Soft Clustering**: Each data object may belong to multiple clusters, with each data object possessing a degree of membership of each cluster.
  - E.g.: Clothing colour. For example, a shirt may be red, blue, or red and blue.

Clustering Evaluation

- How do we know whether the resulting clusters are good, or just useless?
- Extreme cases: Assign all objects to a single cluster, or give each object its own cluster.
- Several methods available (depending on availability of ground-truth), but we will discuss the *Silhouette Coefficient*.

Clustering Evaluation

The Silhouette Coefficient

- Let D be a data set of n objects partitioned into k clusters, $C_1, \ldots, C_k$.
- For each object $\mathbf{o} \in D$, calculate $a(\mathbf{o})$ as the average distance between $\mathbf{o}$ and all other objects in the same cluster as $\mathbf{o}$:

$$a(\mathbf{o}) = \frac{\sum_{\mathbf{o}' \in C_i, \mathbf{o} \neq \mathbf{o}'} dist(\mathbf{o}, \mathbf{o}')}{|C_i| - 1}$$

Clustering Evaluation

The Silhouette Coefficient

- For each object $\mathbf{o} \in D$, calculate $b(\mathbf{o})$ as the minimum average distance from $\mathbf{o}$ to all clusters to which $\mathbf{o}$ does not belong:

$$b(\mathbf{o}) = \min_{C_j:1\leq j\leq k, j\neq i} \left\{ \frac{\sum_{\mathbf{o}'\in C_j} dist(\mathbf{o}, \mathbf{o}')}{|C_j|} \right\}$$

- The *Silhouette Coefficient* of $\mathbf{o}$ is then defined as:

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

Clustering Evaluation

The Silhouette Coefficient

- $s(\mathbf{o})$ is between -1 and 1.
- $a(\mathbf{o})$ reflects the compactness of the cluster to which $\mathbf{o}$ belongs.
  - The smaller $a(\mathbf{o})$ is, the more compact the cluster.
- $b(\mathbf{o})$ is the degree to which $\mathbf{o}$ is separated from other clusters.
  - The larger $b(\mathbf{o})$, the more separated $\mathbf{o}$ is from other clusters.

Clustering Evaluation

The Silhouette Coefficient

- As $s(\mathbf{o})$ approaches 1, the cluster containing $\mathbf{o}$ is compact and $\mathbf{o}$ is far from other clusters (this is good!).
- If the Silhouette Coefficient is negative (i.e., $b(\mathbf{o}) < a(\mathbf{o})$), $\mathbf{o}$ is closer to objects in another cluster than the same cluster (this is bad!).
- So far, we have calculated the Silhouette Coefficient for a single object:
  - We can compute the average Silhouette Coefficient for all objects in the data set to measure the overall quality of clustering.

Clustering

Summary

- Clustering aims to find meaningful groups in the data.
- Many approaches to clustering, including:
  - Bottom-up: Hierarchical Agglomerative Clustering
  - Iterative: k-means
  - Density-based: DBSCAN
- We can use Weka to perform clustering, etc.
- We can analyse cluster quality using the Silhouette Coefficient.

Acknowledgments

- Han, J., Pei, J. and Kamber, M., 2012. *Data Mining: Concepts and Techniques*. Elsevier.
- Graham Cormode [Warwick, CS910]
- Florin Ciucu [Warwick, CS430/CS910]