Applied Computational Intelligence

Lecture 7

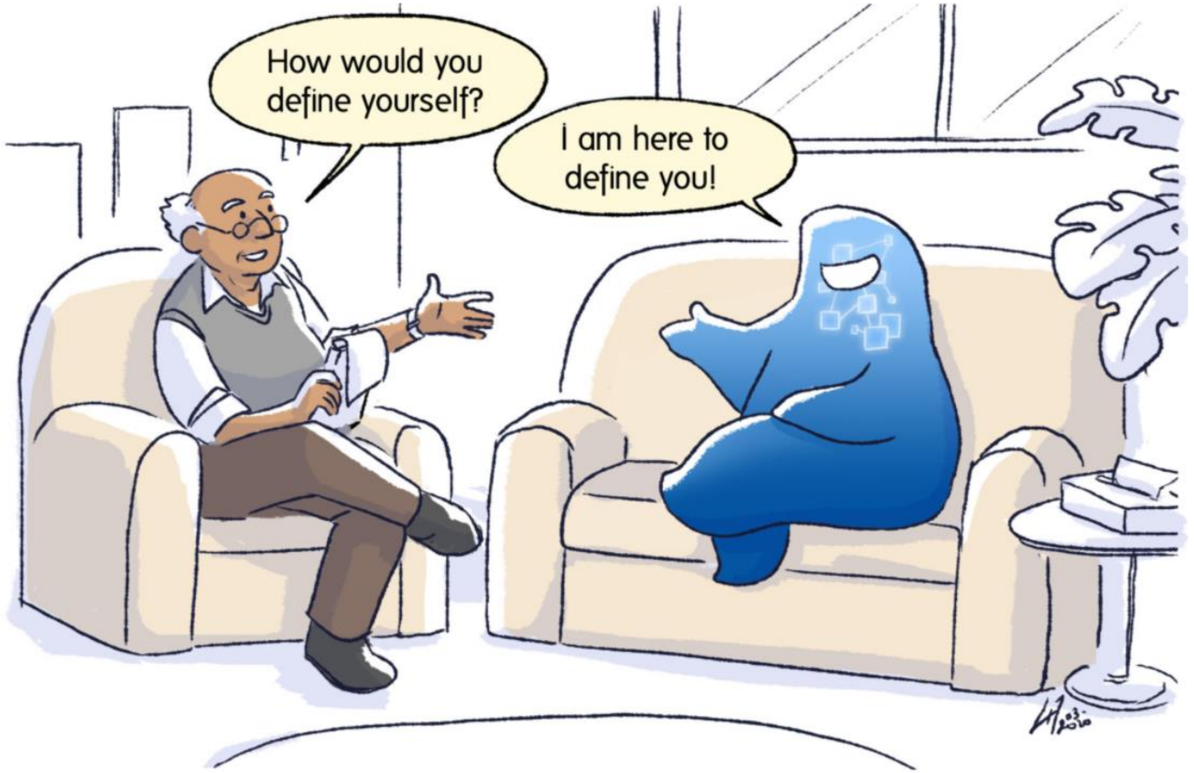


Image source: ScholarOne Manuscripts (manuscriptcentral.com)

Welcome back

- ❑ Project Stage 2
- ❑ Assignment 1
- ❑ Pop quiz

This Session

- Types of Learning Strategies in ML
  - Unsupervised
  - Supervised
  - Reinforcement

Supervised, Unsupervised, and Reinforcement Learning

- **Supervised**: Training data includes **desired output** (labels can be in the form of continuous or discrete).
  - The agent has access to both input and output percepts.

- **Unsupervised**: Training data does **not include desired output**.
  - The agent has no hint at all about the correct outputs

- **Reinforcement**: The model is not trained on sample data. Rather, an agent performs sequence of actions interacting with an environment and learns by itself through rewards or penalties (**trial and error**)
  - The agent receives some evaluation of its action (penalty, like a hefty bill for rear-ending the car in front)

Types of Machine Learning Problems

- Clustering (generally unsupervised)
  - K-means
  - Mean-shift
  - Probabilistic mixture models (e.g., GMM)

- Classification (generally supervised)
  - Logistic regression
  - SVM
  - ANN

Types of Machine Learning Problems Cont.

- Regression (generally supervised)
  - Linear regression
  - Polynomial regression
  - Random forests

- Association (generally unsupervised)
  - Association rule
  - Apriori

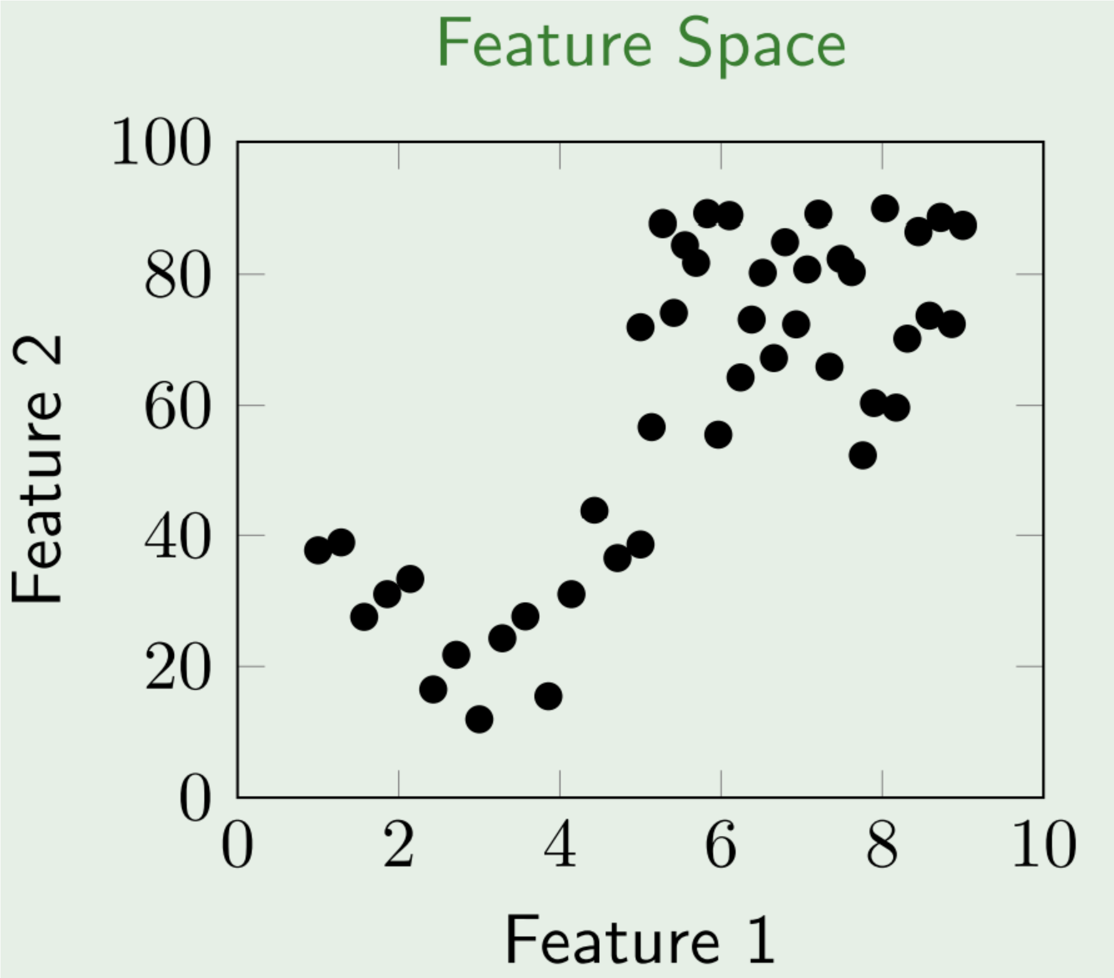
Clustering

- It can be considered the most important **unsupervised learning** problem; it deals with finding a structure in a collection of **unlabeled data**.

- There are no predictions made. Rather, finding the similarities between objects based on the attributes and group similar objects into clusters.
  - organizing objects into groups whose members are similar in some way.
  - objects are "similar" between them within a cluster and are "dissimilar" to the objects belonging to other clusters.
  - **Intra-class** distances are **minimized** while **inter-class** distances are **maximized**.

- Applied in marketing, economics, and various branches of science.

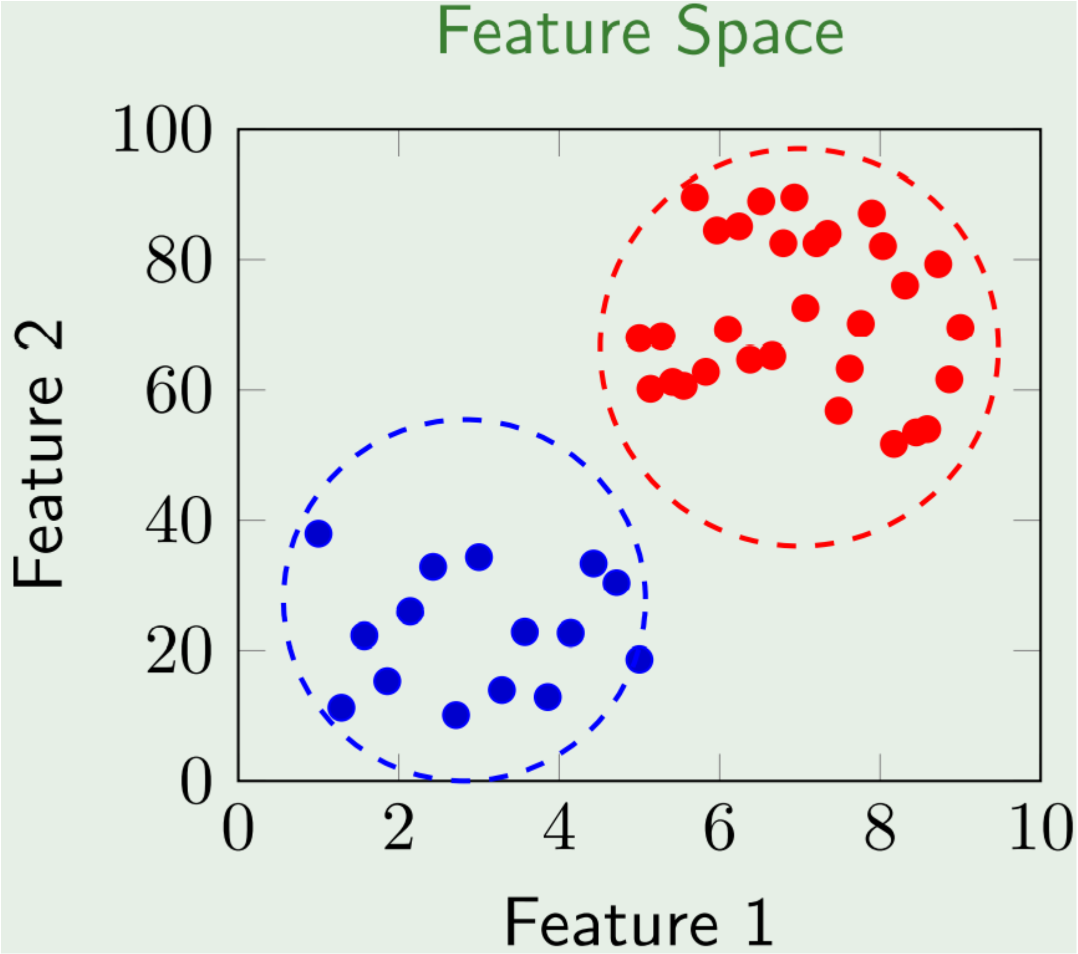
Clustering – Example



Question:

- How many clusters can be identified into which the data can be divided?
- Based on what similarity are the clusters formed?

Clustering – Example Cont.



Ans:

- Two clusters, based on objects' proximity measure (e.g., L_1 distance) two clusters can be identified.

K-means Clustering

- A **computationally faster** unsupervised learning method that solves the clustering problem.
- It divides a given dataset based on **fixed a priori (*k*)**, number of clusters.
- The main idea is to define ***k* centroids**, one **for each cluster**, resulting in **partition of *n* items into *k* clusters**.
- The **centroids** are determined, as the **arithmetic average** (mean) of each cluster's **d-dimensional vector of attributes**.
- In the previous example, $d = 2$.

K-means Clustering – Algorithm

- Four steps.
  - **Initialization:** Set initial group centroids $C = \{c_1, c_2, \ldots, c_k\}$ into the space represented by the dataset $X = \{x_1, x_2, \ldots, x_n\}$, where x_i represents an individual instances or objects in the space ($k < n$).
  - **Assignment:** Assign each instance to the group that has the closest centroid.
  - **Update:** Recalculate the positions of the centroids, $C = C'$.
  - **Convergence:** Repeat Steps 2 and 3 until the centroids are converged (no longer move).

K-means – Two Important Computations

- Computation of **dissimilarity matrix** based on a cost function:

1. For each instance x_i:

- Find nearest centroid c_j using $\arg\min_j D(x_i, c_j)$, where $x \in \mathbf{R}^{n\times d}$ and $c \in \mathbf{R}^{K\times d}$. n - number of samples, d - number of attributes
- D is the $n \times K$ distance matrix between every instance and every centroid.
  Note: Choice of distance measure, e.g., Euclidean distance (L_2) is depends on the the data or application.
- Assign the instance x_i to cluster c_j, where $i \in \{1, 2, ..., n\}$ and $j \in \{1, 2, ...K\}$.

K-means – Two Important Computations Cont.

- Computation of **cluster centroids**:

2. For each cluster c_j:

- New centroid c_j = mean of all the instances, let's say set $\mathcal{A}$ assigned to the cluster c_j in the previous step.
  $c_j(a) = \frac{1}{n_j} \sum_{\mathcal{A}_i \rightarrow c_j} \mathcal{A}_i(a)$.

K-means – Global Indicator

- **Recall:** **Intra-class** distances are to be **minimized**

Objective function: Global indicator

- Finally, this K-means aims at **minimizing a cost function**, e.g., a squared error function, like $J = \sum_{j=1}^{k} \sum_{i=1}^{n} ||x_i^{(j)} - c_j||^2$, where $||x_i^{(j)} - c_j||^2$ is a chosen distance measure between an instance x_i and the cluster centroid c_j, is an indicator of the distance of the n instances from their **respective cluster centres**.

K-means – Properties

- Optimality and complexity:

Characteristics:

- It does not necessarily find the most optimal configuration, corresponding to the global objective function minimum.
- It is sensitive to the initially chosen cluster centroids. It can be run multiple times to reduce this effect.
- Complexity: $O(\#iterations \times \#clusters \times \#instances \times \#attributes)$.

K-means – Working Example

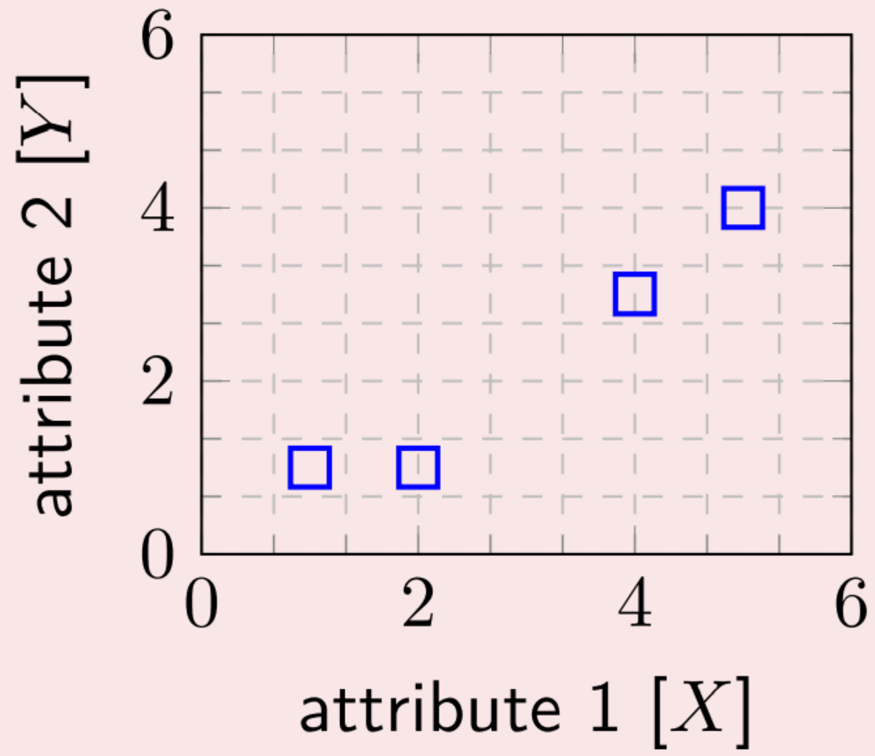
- **Task: divide** the objects (instances) **into two clusters**

Input data

Instance	X	Y
A	1	1
B	2	1
C	4	3
D	5	4

- Each instance represents one point with two-attribute coordinate (X, Y), as in the attribute space shown in the figure.

Attribute space



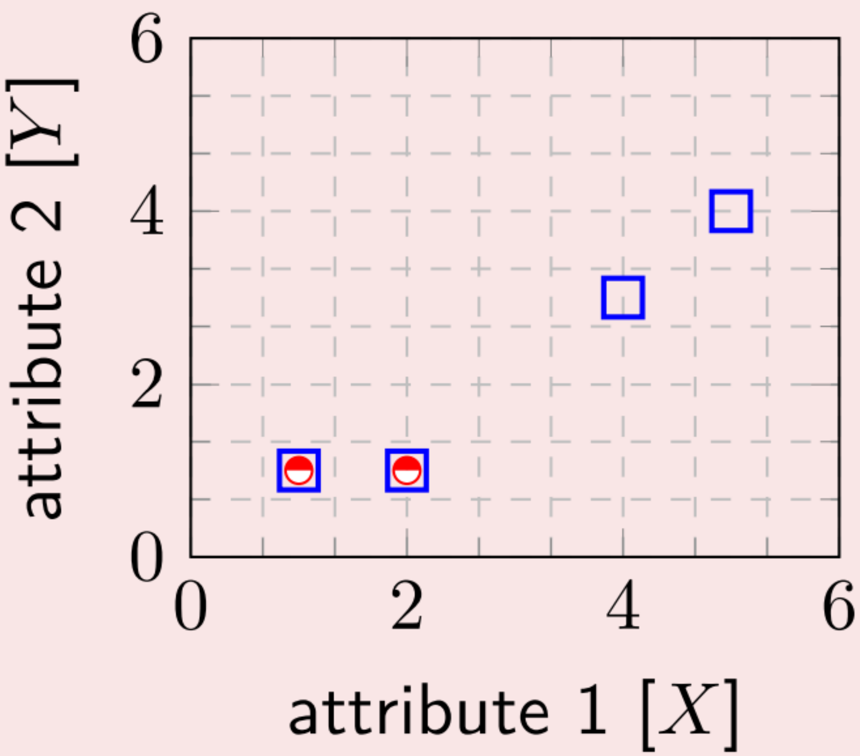
K-means – Working Example Cont.

Step1: Initialization

Instance	X	Y
A	1	1
B	2	1
C	4	3
D	5	4

- Suppose we use instance A and instance B as the initial centroids. Let c_1 and c_2 denote the coordinate of the centroids, then $c_1 = (1,1)$ and $c_2 = (2,1)$.

Attribute space



K-means – Working Example Cont.

Step2: Cluster assignment

- Instance-Centroid distance: Calculate the distance between cluster centroid to each instance. Let us use Euclidean measure: $\sqrt{(x_i^{(j)} - c_j)^2}$.
- Column: instance
- 1st row: distance of each instance to the 1st centroid,
- 2nd row: distance of each instance to the 2nd centroid.

- Distance matrix at iteration 0
- $$D^0 = \begin{array}{cc} \begin{bmatrix} 0 & 1 & 3.61 & 5 \\ 1 & 0 & 2.83 & 4.24 \end{bmatrix} & \begin{array}{ll} C_1 = (1,1) & group1 \\ C_2 = (2,1) & group2 \end{array} \\ \begin{array}{cccc} A & B & C & D \end{array} & \\ \begin{bmatrix} 1 & 2 & 4 & 5 \\ 1 & 1 & 3 & 4 \end{bmatrix} & \begin{array}{l} X \\ Y \end{array} \end{array}$$

- For example, distance from instance $C = (4,3)$ to 1st centroid $c_1 = (1,1)$ is $\sqrt{(4-1)^2 + (3-1)^2} = 3.61$ and to 2nd centroid $c_2 = (2,1)$ is $\sqrt{(4-2)^2 + (3-1)^2} = 2.83$

K-means – Working Example Cont.

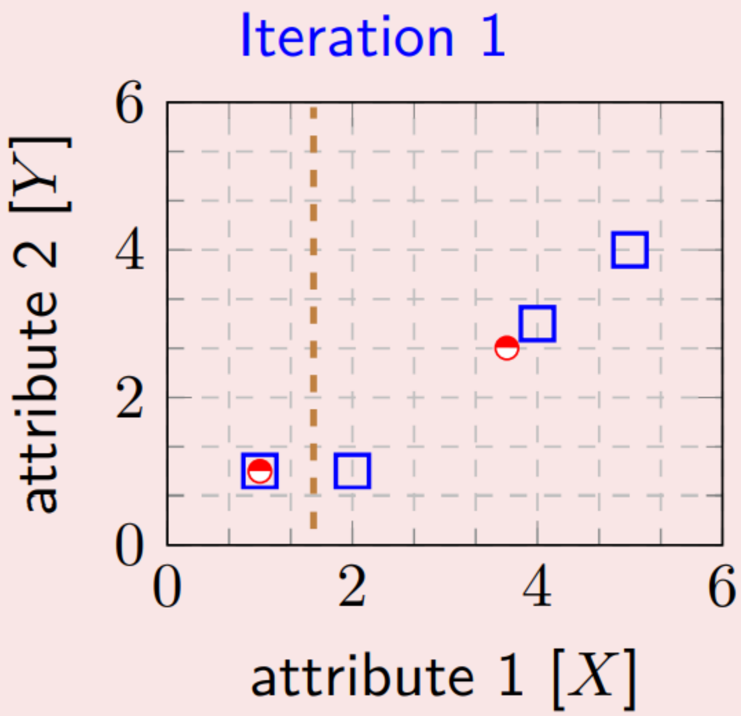
Step2: Cluster assignment cont.

- Assign each instance based on the minimum distance:
  $I_q \to C_i \quad if \quad ||I_q - c_i||^2 < ||I_q - c_j||^2$, i.e., use $\arg\min_j D(I_q, c_j)$,
  where $\{i, j\} \in 1, 2, ..., k \ \ (j \neq i)$ and $q = 1, 2, ..., Q$.
- $$D^0 = \begin{bmatrix} 0 & 1 & 3.61 & 5 \\ 1 & 0 & 2.83 & 4.24 \end{bmatrix} \begin{matrix} C_1 = (1,1) & group1 \\ C_2 = (2,1) & group2 \end{matrix}$$
  $$G^0 = \begin{array}{c} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix} \begin{matrix} group1 \\ group2 \end{matrix} \\ \begin{matrix} A & B & C & D \end{matrix} \end{array}$$
- Thus, instance **A** is assigned to **Group 1**, while other instances **B**, **C**, and **D** are assigned to **Group 2**.
- The element of Group matrix (G) is 1 if and only if the object is assigned to that group.

K-means – Working Example Cont.

Step3: Cluster update - Iteration 1

- Update centroids: Knowing the members of each group, now we recompute the new centroid of each group based on these new memberships.
- Group 1 has only one member; thus, the centroid remains in $c_1 = (1, 1)$.
- Group 2 now has three members, thus the updated centroid is the mean coordinate of the attributes among the three members: $c_2 = \left(\frac{2+4+5}{3}, \frac{1+3+4}{3}\right) = (3.67, 2.67)$.



K-means – Working Example Cont.

Step3: Cluster update - Iteration 1

- Compute the distance of all instances to the **distance new centroids**. Similar to Step 2, the distance matrix at iteration 1 is:
- $D^1 = \begin{bmatrix} 0 & 1 & 3.61 & 5 \\ 3.14 & 2.36 & 0.47 & 1.89 \end{bmatrix} \begin{matrix} C_1 = (1.00, 1.00) & group1 \\ C_2 = (3.67, 2.67) & group2 \end{matrix}$

  $\begin{matrix} A & B & C & D \end{matrix}$

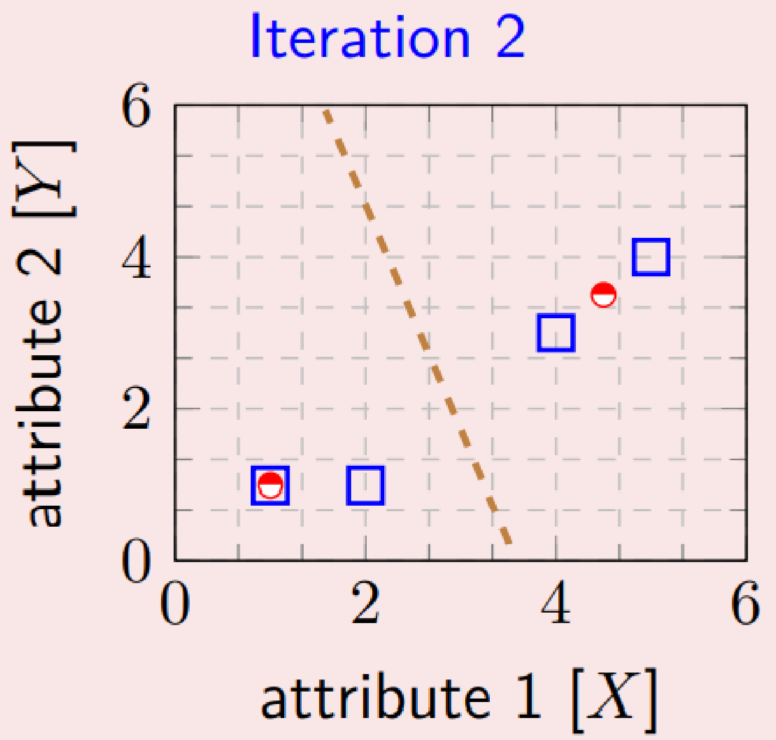
  $\begin{bmatrix} 1 & 2 & 4 & 5 \\ 1 & 1 & 3 & 4 \end{bmatrix} \begin{matrix} X \\ Y \end{matrix}$
- Similar to Step 2, each **instance is assigned** to a cluster using $\arg\min_j D(I_q, c_j)$. Based on the new distance matrix, the instance B is assigned to Group 1 while all the other instances remain.
- The new Group matrix: $G^1 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \begin{matrix} group1 \\ group2 \end{matrix}$

K-means – Working Example Cont.

Step3: Cluster update - Iteration 2

- Update centroids: Now, **repeat** calculating the new coordinates of the centroids based on new members assigned in the previous iteration.

- Group 1 and 2 both have two members, thus the new centroids are:
  $c_1 = \left(\frac{1+2}{2}, \frac{1+1}{2}\right) = (1.5, 1)$, and
  $c_2 = \left(\frac{4+5}{2}, \frac{3+4}{2}\right) = (4.5, 3.5)$.



K-means – Working Example Cont.

Step3: Cluster update - Iteration 2

- Again compute the distance of all instances to the new centroids. Similar to Step 2, the distance matrix at iteration 2 is:
- $D^1 = \begin{bmatrix} 0.5 & 0.5 & 3.20 & 4.61 \\ 4.30 & 3.54 & 0.71 & 0.71 \end{bmatrix} \begin{array}{ll} C_1 = (1.5, 1.0) & group1 \\ C_2 = (4.5, 3.5) & group2 \end{array}$

$$\begin{array}{cccc} A & B & C & D \end{array}$$

$$\begin{bmatrix} 1 & 2 & 4 & 5 \\ 1 & 1 & 3 & 4 \end{bmatrix} \begin{array}{l} X \\ Y \end{array}$$

- Similar Iteration 1, again, each instance is assigned to a cluster based on the minimum distance. $G^0 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \begin{array}{l} group1 \\ group2 \end{array}$

K-means – Working Example Cont.

Step4: Termination

- Comparing the grouping of last iteration and current iteration reveals that the **instances do not move** and **remain in the same groups**.
- Thus, the computation of the K-mean clustering has reached its stability and no more iteration is needed.
- **Final clusters:**

Instance	X	Y	Group
A	1	1	Group1
B	2	1	Group1
C	4	3	Group2
D	5	4	Group2

Please classify a test sample $t = (3, 4)$.

- **Trained model:** final centroids are the the classifier's attributes.
- **At Inference time:** a test sample will be checked for nearest centroid and assigned to a group, like a **nearest neighbor classifier**

This Session

- Dimensionality Reduction
  - Feature Extraction
  - Feature Selection

Dimensionality Reduction

- Curse of dimensionality:
  - Many complex analysis tasks like regression or classification much more difficult in high-dimensional spaces than in low-dimensional ones.
  - Algorithmic complexity - time and memory consumption
  - Workload distribution - vectorized, parallel, or distributed architectures
  - Hard for visual presentation

- Solution:
  - Dimensionality reduction.

Dimensionality Reduction Cont.

- Dimensionality Reduction:
  - A reduced representation of the data set.
  - Smaller in volume yet produces the same (or almost the same) results as the raw high dimensional data.
  - Data encoding schemes:
    - ✓ Attribute subset selection, like removing irrelevant attributes
    - ✓ Attribute construction, like creating a small set of more useful attributes derived from the original set.
    - ✓ Data compression techniques, like principal components analysis or wavelet transforms

Dimensionality Reduction Cont.

- **Example:**
  - **Input** attributes: a, b, c, d
  - **Target** attribute: t
    - ✓Model: $m(\{a, b, c, d\}, \{t\})$
    - ✓New attribute: $z = f(a, b)$
    - ✓Model: $m(\{z, c, d\}, \{t\})$

⇦**Attribute construction**

  - If we find c has insignificant discriminability between clusters or classes or c ≈ 0, and can be removed the model.
    - ✓$M(\{a, b, d\}, \{t\})$

⇦**Attribute subset selection**