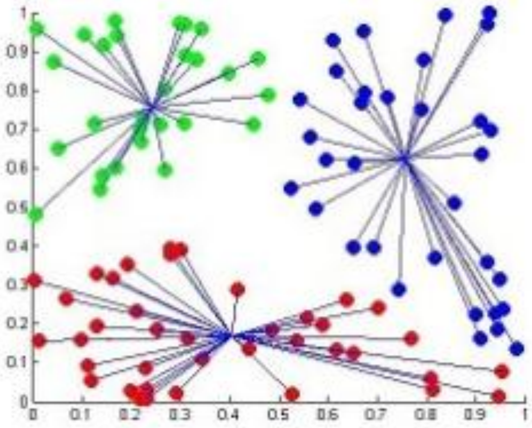
Resources

- [Video](https://www.youtube.com/watch?v=65G5xkWXTTk&ab_channel=mohammadrkieh) (recording of presentation)

https://www.youtube.com/watch?v=65G5xkWXTTk&ab_channel=mohammadrkieh

- PDF (notes in pdf format)

Fuzzy Clustering



Mohammad Rkieh (104928868)

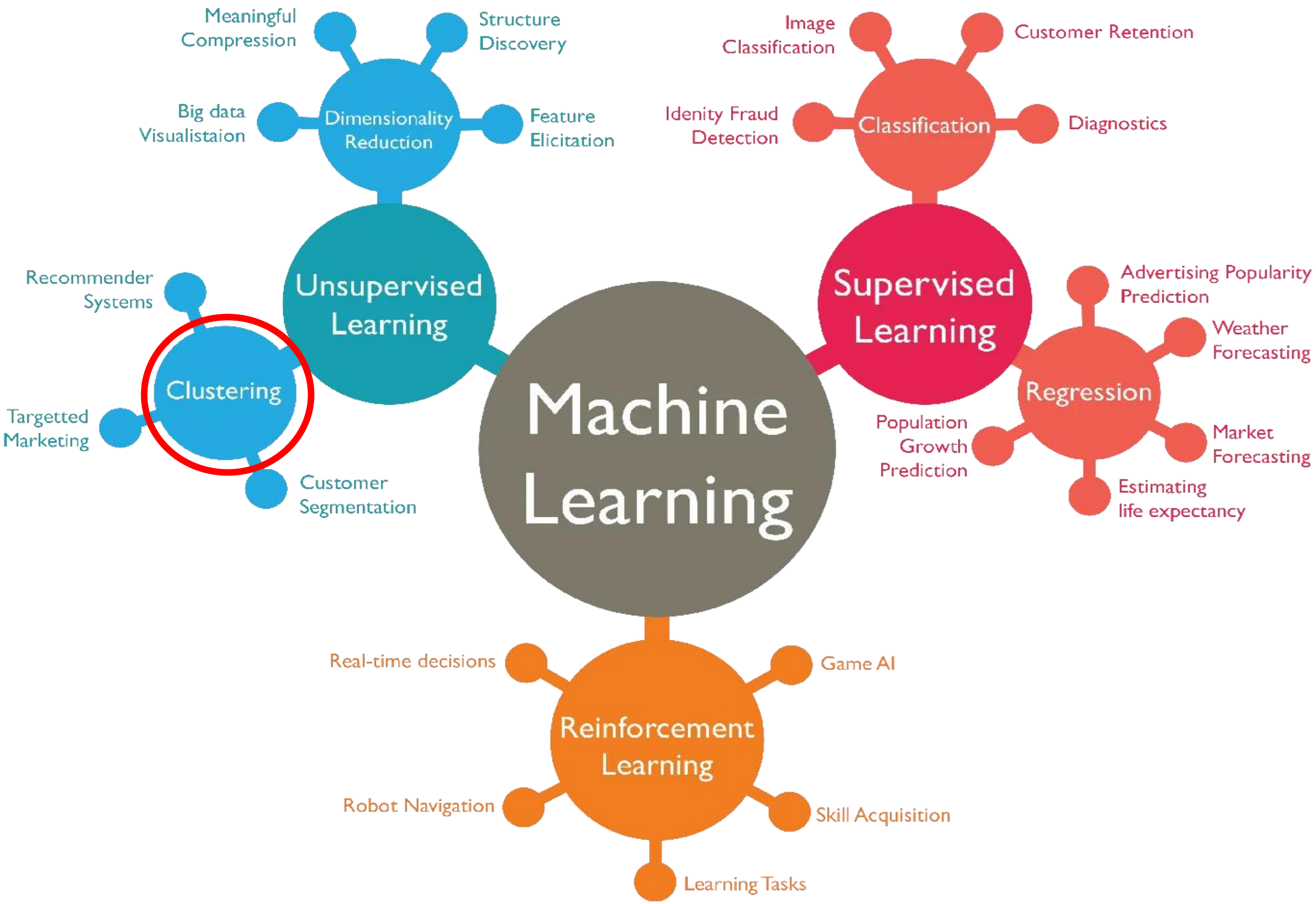
Instructor: Dr. Yasser Alginahi

Date: 3rd of November, 2023

Table of Contents

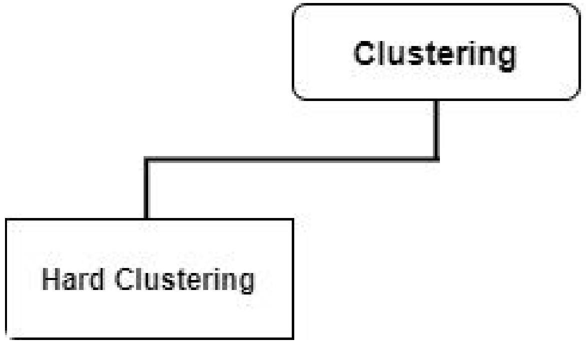
Introduction

- Unsupervised-learning
- Clustering

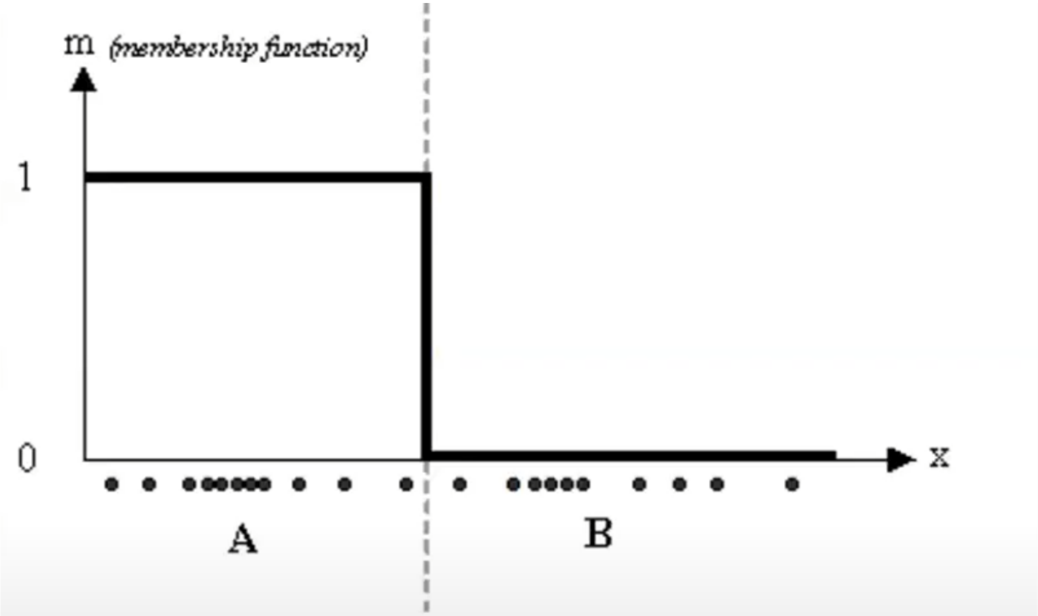


Machine Learning Diagram [1]

Limitation of Hard Clustering



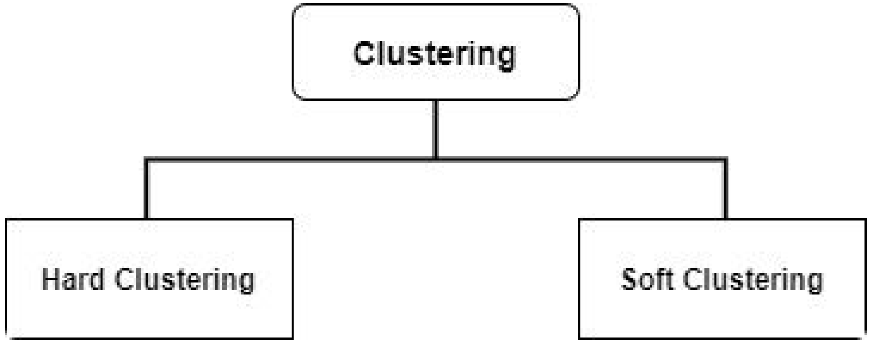
Hard Clustering (k-means)



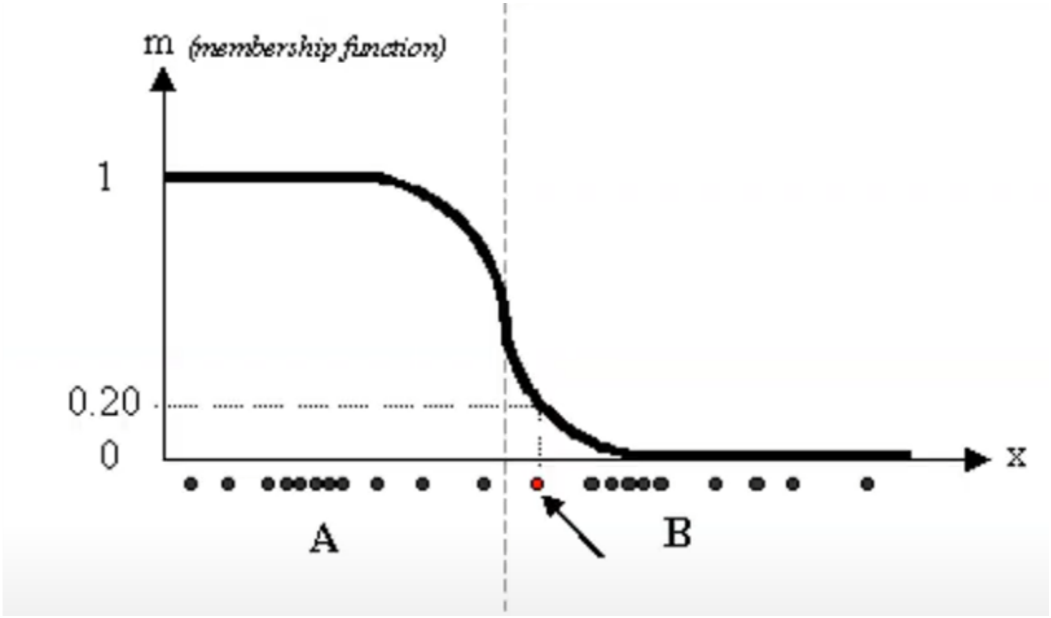
Hard Clustering Membership Function [2]

University of Windsor

Solution



Soft Clustering (fuzzy)



Soft Clustering Membership Function [2]

Fuzzy C-Means (FCM) algorithm

- Flexible clustering algorithm
- Soft Clustering version of K-Means

Applications

- Pattern Recognition
- Marketing
- Medical Diagnosis
- Image Segmentation

Mathematical Concept - Variables

n = number of data points

v_j = j^{th} cluster center

m = fuzziness index m ∈ [1, ∞]

c = number of cluster centers

u_{ij} = membership of i^{th} data to j^{th} cluster center

x_i = i^{th} of d-dimensional measured data

d_{ij} = Euclidean distance between i^{th} data and j^{th} cluster center

Mathematical Concept – Steps (1)

Step *1* is intuitive but the user can opt to manually define *c*. Step *2* repeats calculation of membership u_{ij} and move centroids v_j. The fuzzy membership algorithm is as follows:

$$u_{ij} = \frac{1}{\sum_{k=1}^{c} \left(\frac{d_{ij}}{d_{ik}}\right)^{\frac{2}{m-1}}}$$

Where…

$$d_{ij} = \left|\left|x_i - v_j\right|\right|, \; d_{ik} = \left|\left|x_i - v_k\right|\right|$$

If done correctly, $\sum_{j=1}^{c} u_{ij} = 1$ should be true because each u_{ij} acts as a percentage-value (eg. *0.1 + 0.3 + 0.6 = 1*)

Mathematical Concept Steps [5]

University of Windsor

Mathematical Concept – Steps (2)

The next step is to realign the centroids of each cluster which is the following:

$$v_j = \frac{\left(\sum_{i=1}^{n} (u_{ij})^m x_i\right)}{\left(\sum_{i=1}^{n} (u_{ij})^m\right)}, \forall_j = 1, 2, \ldots c$$

- Sum of all the weighted values over the sum of the data
- Finds the center of gravity for each cluster centroid

These two steps repeat until $\varepsilon > max_{ij}\left\{\left|u_{ij}^{(k+1)} - u_{ij}^{k}\right|\right\}$ **where…**

k is the iteration step.

β is the termination criterion between $[0, 1]$

$\left(u_{ij}\right)_{n\times c}$ is the fuzzy membership matrix

J is the objective function

which converges to a local minimum/saddle point of Jm. When the program terminates, we will have *c* number of clusters that are at their final values.

Pseudocode

1. Randomly select *c* cluster centers
2. **Repeat**
    a. Calculate the fuzzy membership u_{ij} for each data in a cluster
    b. Compute the centroids v_j for each cluster
3. **Until minimum *J* achieved**

Code

- Skfuzz library

```
from __future__ import division, print_function
import numpy as np
import matplotlib.pyplot as plt
import skfuzzy as fuzz

colors = ['b', 'orange', 'g', 'r', 'c', 'm', 'y', 'k', 'Brown', 'ForestGreen']

# Define three cluster centers
centers = [[4, 2],
           [1, 7],
           [5, 6]]

# Define three cluster sigmas in x and y, respectively
sigmas = [[0.8, 0.3],
          [0.3, 0.5],
          [1.1, 0.7]]

# Generate test data
np.random.seed(42)  # Set seed for reproducibility
xpts = np.zeros(1)
ypts = np.zeros(1)
labels = np.zeros(1)
for i, ((xmu, ymu), (xsigma, ysigma)) in enumerate(zip(centers, sigmas)):
    xpts = np.hstack((xpts, np.random.standard_normal(200) * xsigma + xmu))
    ypts = np.hstack((ypts, np.random.standard_normal(200) * ysigma + ymu))
    labels = np.hstack((labels, np.ones(200) * i))

# Visualize the test data
fig0, ax0 = plt.subplots()
for label in range(3):
    ax0.plot(xpts[labels == label], ypts[labels == label], '.',
             color=colors[label])
ax0.set_title('Test data: 200 points x3 clusters.')
```

Advantages

- Flexibility
- Robustness
- No preset number of clusters

Disadvantages

- Complexity
- Model selection
- Deciding on the number of clusters

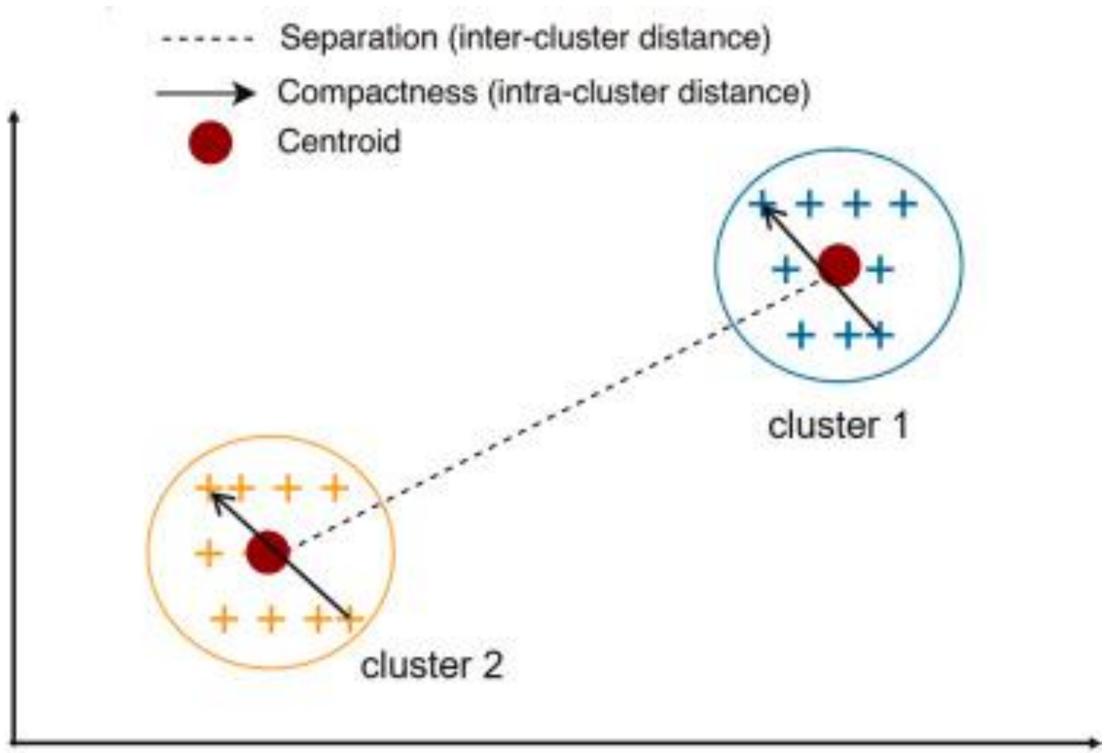
The Dunn Index (DI)

- Cluster validation metric

- $$\boldsymbol{DI = \frac{min\_inter{-}cluster\_distance}{max\_intra{-}cluster\_distance}}$$

- Higher Dunn Index, Better Clustering

- DI of 0 typically indicates poor clustering



References

[1] Rajesh Khadka, "Machine Learning Types #2 - Towards Data Science," *Medium*, Sep. 07, 2017. https://towardsdatascience.com/machine-learning-types-2-c1291d4f04b1

[2] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*. Springer Science & Business Media, 2013.

[3] Witold Pedrycz, *An Introduction to Computing with Fuzzy Sets*. Springer Nature, 2020.

[4] "Day 71 - Fuzzy C-Means Clustering Implementation," *www.youtube.com*. https://www.youtube.com/watch?v=W-3ZYGmLJ-4&list=PPSV (accessed Oct. 17, 2023)

[5] "Dunn index and DB index - Cluster Validity indices | Set 1," *GeeksforGeeks*, May 09, 2019. https://www.geeksforgeeks.org/dunn-index-and-db-index-cluster-validity-indices-set-1/

[6] "Fuzzy c-means," *www.youtube.com*. https://www.youtube.com/watch?v=zr50h_91gOw&ab_channel=jeff (accessed Oct. 20, 2023).

[7]"Day 70 - Fuzzy C-Means Clustering Algorithm," *www.youtube.com*. https://www.youtube.com/watch?v=VhYt7nxOKKs&ab_channel=DataSciencewithSharan (accessed Oct. 20, 2023).

[8]"Fuzzy C Means Clustering Algorithm Solved Example | Clustering Algorithm in ML & DL by Mahesh Huddar," *www.youtube.com*. https://www.youtube.com/watch?v=X7co6-U4BJY&t=230s&ab_channel=MaheshHuddar (accessed Oct. 20, 2023).