Fuzzy Clustering Algorithm

Dr.Omkaresh Kulkarni

- **Clustering** is a fundamental technique in machine learning used to group similar data points together. Traditional clustering methods, such as **K-Means**, assign each data point to a single cluster, creating well-defined boundaries.

- However, in many real-world scenarios, data points don’t belong strictly to one cluster but rather exhibit characteristics of multiple clusters simultaneously.
- This is where **Fuzzy Clustering**, based on **Fuzzy Logic**, making it more suitable for handling uncertainty and overlapping data distributions.

Fuzzy Clustering in Machine learning

- **Fuzzy Clustering** is a type of clustering algorithm in machine learning that allows a data point to belong to more than one cluster with different degrees of membership. Unlike traditional clustering (like **K-Means**), where each data point belongs to only **one** cluster, fuzzy clustering allows a data point to belong to multiple clusters with different membership levels.

- Guest are casually forming groups based on shared interests like **music lovers, food enthusiasts, and sports fans.**
- Some people clearly fit into one group
- But others might belong to multiple groups!

How Does Fuzzy Clustering Work?

- Fuzzy clustering follows an iterative optimization process where data points are assigned membership values instead of hard cluster labels.

Step 01: Initialize Membership Values Randomly

- Each data point is assigned a membership degree for all clusters. These values indicate the probability of the data point belonging to each cluster. Unlike hard clustering (where a point strictly belongs to one cluster), fuzzy clustering allows partial membership.

- Let us assume there are 2 clusters in which the data is to be divided, initializing the data point randomly. Each data point lies in both clusters with some membership value which can be assumed anything in the initial state.
- The table below represents the values of the data points along with their membership (gamma) in each cluster.

The table below represents the values of the data points along with their membership (gamma) in each cluster.

Cluster	(1, 3)	(2, 5)	(4, 8)	(7, 9)
1)	0.8	0.7	0.2	0.1
2)	0.2	0.3	0.8	0.9

Step 02: Compute Cluster Centroids

- The centroids of the clusters are calculated based on the weighted sum of all data points, where weights are determined by membership values. This ensures that points with higher membership contribute more to the centroid.

The formula for finding out the centroid (V) is:

$$V_{ij} = (\sum_{1}^{n}(\gamma_{ik}^{m} * x_k) / \sum_{1}^{n} \gamma_{ik}^{m}$$

Where, **μ is fuzzy membership value** of the data point, **m is the fuzziness parameter** (generally taken as 2), and **xk is the data point.**

```
V11  = (0.8^2 *1 + 0.7^2 * 2 + 0.2^2 * 4 + 0.1^2 * 7) / ( (0.8^2 +
0.7^2  + 0.2^2  + 0.1^2 ) = 1.568
V12  = (0.8^2 *3 + 0.7^2 * 5 + 0.2^2 * 8 + 0.1^2 * 9) / ( (0.8^2 +
0.7^2  + 0.2^2  + 0.1^2 ) = 4.051
V21  = (0.2^2 *1 + 0.3^2 * 2 + 0.8^2 * 4 + 0.9^2 * 7) / ( (0.2^2 +
0.3^2  + 0.8^2  + 0.9^2 ) = 5.35
V22  = (0.2^2 *3 + 0.3^2 * 5 + 0.8^2 * 8 + 0.9^2 * 9) / ( (0.2^2 +
0.3^2  + 0.8^2  + 0.9^2 ) = 8.215
```

Centroids are: (1.568, 4.051) and (5.35, 8.215)

Step 03: Calculate Distance Between Data Points and Centroids:

- The Euclidean distance (or another distance metric) between each data point and the centroids is computed. This helps in updating the membership values.

$$D11 = ((1 - 1.568)^2 + (3 - 4.051)^2)^{0.5} = 1.2$$

$$D12 = ((1 - 5.35)^2 + (3 - 8.215)^2)^{0.5} = 6.79$$

Step 04: Update Membership Values:

$$\gamma = \sum_{1}^{n} \left(d_{ki}^2 / d_{kj}^2\right)^{1/m-1}]^{-1}$$

For point 1 new membership values are:

$$\gamma_{11} = [\{ [(1.2)^2 / (1.2)^2] + [(1.2)^2 / (6.79)^2]\} \wedge \{(1 / (2 - 1))\}]^{-1} = 0.96$$

$$\gamma_{12} = [\{ [(6.79)^2 / (6.79)^2] + [(6.79)^2 / (1.2)^2]\} \wedge \{(1 / (2 - 1))\}]^{-1} = 0.04$$

Alternatively,

$$\gamma_{12} = 1 - \gamma_{11} = 0.04$$

Similarly, compute all other membership values, and update the matrix.

Step 05: Repeat Until Convergence

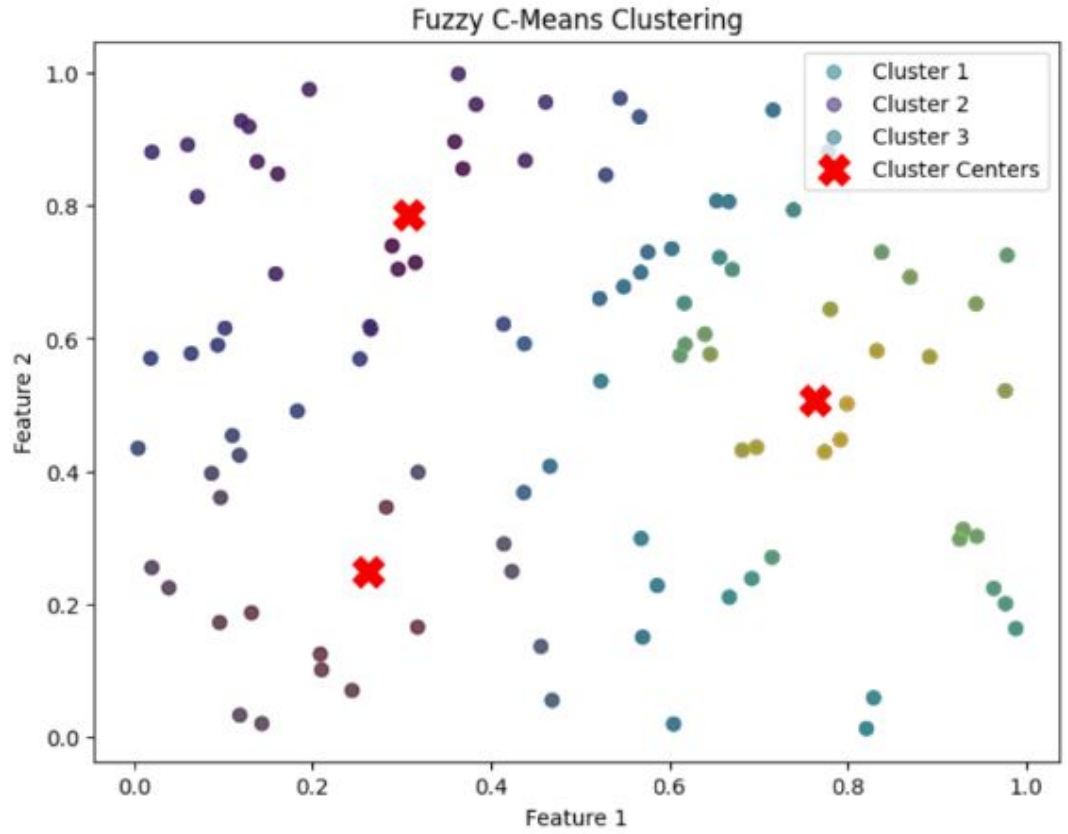
- Steps 2–4 are repeated until the membership values stabilize, meaning there are no significant changes from one iteration to the next. This indicates that the clustering has reached an optimal state.

Step 06: Defuzzification (Optional):

- In some cases, we may want to convert fuzzy memberships into crisp cluster assignments by assigning each data point to the cluster where it has the highest membership

Implementation of Fuzzy Clustering in Python

- The **fuzzy scikit learn** library has a pre-defined function for **fuzzy c-means** which can be used in Python. For using fuzzy c-means you need to install the **skfuzzy** library.



Fuzzy C means Clustering

- The plot demonstrates that **FCM allows soft clustering**, meaning a point can belong to multiple clusters with different probabilities rather than being assigned to just one cluster. This makes it useful when boundaries between clusters are not well-defined and all the **Red “X” markers** indicate the **cluster centers** computed by the algorithm.

Advantages of Fuzzy Clustering

- **Flexibility:** Fuzzy clustering allows for overlapping clusters, which can be useful when the data has a complex structure or when there are ambiguous or overlapping class boundaries.
- **Robustness:** Fuzzy clustering can be more robust to outliers and noise in the data, as it allows for a more gradual transition from one cluster to another.

- **Interpretability:** Fuzzy clustering provides a more nuanced understanding of the structure of the data, as it allows for a more detailed representation of the relationships between data points and clusters.

Disadvantages of Fuzzy Clustering

- **Complexity**: Fuzzy clustering algorithms can be computationally more expensive than traditional clustering algorithms, as they require optimization over multiple membership degrees.
- **Model selection:** Choosing the right number of clusters and membership functions can be challenging, and may require expert knowledge or trial and error.

Conclusion

- Fuzzy Clustering, especially Fuzzy C-Means (FCM), provides a more flexible approach by allowing data points to belong to multiple clusters with varying degrees of membership.
- This is useful when data lacks clear boundaries between clusters.