## Problem Statement

The Fuzzy C-Means (FCM) algorithm is an unsupervised clustering algorithm and is considered one of the most effective and widely used algorithms for medical image segmentation. It is like the traditional K-means algorithm but with "soft" memberships. Instead of binary memberships (0 or 1) that indicate whether a data point belongs to a cluster, each point in FCM has a weighted membership value, a number between 0 and 1. This value represents the degree of membership or probability of the point belonging to each cluster (Pseudocode 1).

**Step 1**: Determine the number of clusters c and $\varepsilon$

**Step 2**: Initialize the center of the cluster $v_i^{(0)}$ and $u_{ij}^{(0)}$

**Step 3**: k=1

**Step 4**: While $\left\|v_i^{(k)} - v_i^{(k-1)}\right\| > \epsilon$

Calculation of $u_{ij}^{(k)}$ and $v_i^{(k)}$ using equations (3-4)

$$u_{ij}^{(k)} = \frac{\left(1/\left\|x_j - v_i^{(k-1)}\right\|^2\right)^{\frac{1}{m-1}}}{\Sigma_{j=1}^{c}\left(1/\left\|x_j - v_i^{(k-1)}\right\|^2\right)^{1/m}}, \forall i = 1,2,\ldots.,c, j = 1,2,\ldots..,n \quad (3)$$

$$v_i^{(k)} = \frac{\Sigma_{j=1}^{n}\left(u_{ij}^{(k)}\right)^m x_j}{\Sigma_{j=1}^{N}\left(u_{ij}^{(k)}\right)^m}, \forall i = 1,2,\ldots.,c$$

k=k+1

(4)

**Step 5**: return cluster centers $v_i$ and membership function $u_{ij}$

**Pseudocode 1**: FCM Algorithm

Although the FCM algorithm is simple to implement, it has a few shortcomings such as sensitivity to the cluster center initializations, getting stuck in the local minima and low convergence rate.

Boulanouar et al. propose enhancing the quality of segmentation and the speed of convergence by using the Bat Algorithm for determining the initial cluster centers and defining a fitness function This fitness function combines intra-cluster distance with fuzzy cluster validity indices. They refer to the combined algorithm as **MFBA**.

## BAT Algorithm

The Bat Algorithm (BA) is a metaheuristic optimization technique inspired by natural processes. Specifically, it is inspired from the echolocation behavior of bats, which they use to sense distances. Bats hunting at night emit brief, intense sound pulses and analyze the returning echoes to detect obstacles or prey. Their unique auditory system enables them to determine both the size and location of objects with precision.
In the BA, the location of a bat $x_i$ , $x_i$ represents a potential solution to an optimization problem, evaluated by a fitness function that measures how close the bat is to the optimal solution (or "prey"). The goal is to optimize this fitness value, guiding the bat toward the optimal solution.
Bats fly randomly with velocity $v_i$ at position $x_i$ and emit sounds with loudness A or at varying frequency (fmin, fmax) to search for a prey.

| Parameter | Description |
|---|---|
| nBats | Number of bats |
| IterMax | Maximum number of iterations |
| $f_{min}$, $f_{max}$ | Minimum and maximum frequency |
| Loudness Coefficient | Constant parameter in range [0, 1] used to update the loudness of each bat |
| Gamma | Constant parameter in range [0, 1] used to decay pulse rate |

The steps of the algorithm could be summarized as follows:
**Step 1** - Initialize the BAT algorithm parameters:
Initialize randomly bat positions, set initial velocities to 0. Compute fitness values for each bat with initial position and data. Set up initial best position using initial bat positions.
**Step 2** – Update using best position X*, pulse frequency, the velocity, and position of the $i^{th}$ bat as follows as:

$$\boxed{\begin{aligned} f_i &= f_{\min} + (f_{\max} - f_{\min})\,\beta, \qquad \beta \in [0,1], \\ V_i^{t+1} &= V_i^t + (X_i^t + X^*)\, f_i, \\ X_i^{t+1} &= X_i^t + V_i^t, \end{aligned}}$$

Where $V_i^t$, and $X_i^t$ are the velocity and position at time t (iteration t), $V_i^{t+1}$ and $X_i^{t+1}$ are the velocity and position at time t+1 (iteration t+1).

**Step 3** – If the random number is greater than $r_i$, a new solution for the bat is generated by the following equation:

$$X_{\text{new}} = X_{\text{old}} + \epsilon A^t,$$

where $\epsilon$ is a random number, $\epsilon \in [-1, 1]$, and $A^t$ represents the average loudness of all bats at time $t$.

**Step 4** - If the random number is lower than $A_i$ and $f(X_i) < f(X^*)$, the new solution is accepted. Next, update $A_i$ and $r_i$, respectively, as follows:

$$A_i^{t+1} = \alpha A_i^t,$$
$$r_i^t = r_i^0 \left[1 - e^{-\gamma t}\right],$$

where $A_i^{t+1}$ and $A_i^t$ denote the loudness at times $t$ and $t+1$, respectively; $r_i^0$ and $r_i^t$ are the initial pulse rate and pulse rate at time $t$, respectively, $\alpha$ is a constant parameter in range $[0, 1]$, $\gamma$ is a constant parameter, and $\gamma > 0$. As $t \to \infty$, $A_i^t \to 0$ and $r_i^t \to r_i^0$.

**Step 5.** Sort the bats based on their fitness and find the current optimal solution $X^*$.
**Step 6.** Return to Step 2 until the maximum number of iterations is reached; output the globally optimal solution.

## MBFA

When the BAT algorithm and FCM algorithm are combined, the process involves two main steps. In the first step, the BAT algorithm selects the optimal initial clusters for the FCM algorithm. In the second step, the FCM algorithm refines these clusters to find the optimal cluster assignments and their centers. Both algorithms operate using the same cost function (Pseudocode 2).
Boulanouar et al. define as fitness function, which is minimized when the value of PC is high and the value of (Intra_cluster + SC) is low.

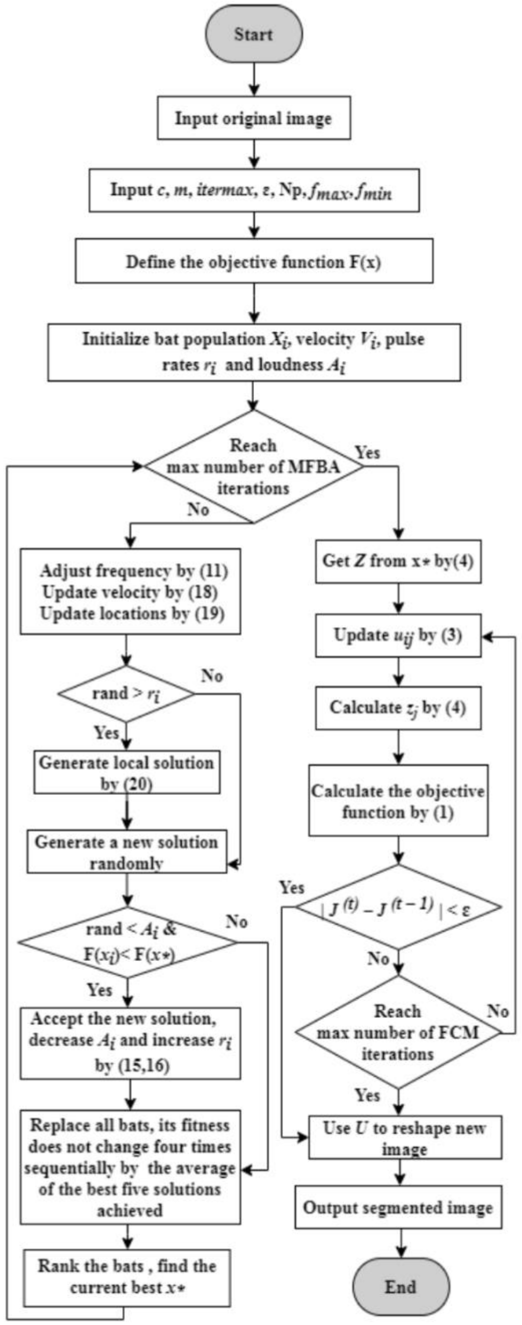
Fitness = (Intra_Cluster + SC) / PC

- **Intra Cluster Distance**: this metrics measures the compactness of the clusters. The goal is to minimize the distance between data points and their assigned cluster centers.
- **Partition Coefficient (PC)**: This metric quantifies the overlap between clusters, with higher values indicating better-defined clusters.

- **Classification Entropy (CE)**: Like PC, CE measures the fuzziness in cluster assignments. Lower values are preferred as they suggest more distinct cluster boundaries.
- **Partition Index (SC)**: This index measures cluster validity based on individual cluster characteristics, normalized by the fuzzy cardinality of each cluster. A higher SC value indicates better separation between clusters.

To have more control and flexibility, we weight each term of the fitness function with weight alpha, beta and zeta.

$$\boxed{\text{fitness} = \alpha \cdot \text{intraCluster} + \beta \cdot SC + \zeta \cdot \left(\frac{1}{PC} + CE\right)}$$




**Pseudocode 2:** MBFA flow chart


[1] Gandomi A. H. and Yang X.-S., Chaotic bat algorithm, *Journal of Computational Science*. (2014) **5**, no. 2, 224–232, https://doi.org/10.1016/j.jocs.2013.10.002, 2-s2.0-84897588368.