FUZZY IMAGE PROCESSING

Fuzzy c-means clustering

Farah Al-Tufaili

Introduction to fuzzy image

What Is Fuzzy Image Processing (FIP)?

- Under fuzzy image processing (FIP) we understand the collection of all methodologies in digital image processing, with which the images, their segments or features which represent these images or their segments, are understood, represented and processed as fuzzy sets. In which manner this takes place, depends on the problem definition and on the respective fuzzy method. This definition refers naturally to other methodologies and not to image processing as A separate branch of science. Thus the misunderstanding has to be avoided that fuzzy image processing may replace the classical image processing.

Introduction to fuzzy image

- Fuzzy image processing is the collection of all approaches that understand, represent and process the images, their segments and features as fuzzy sets. The representation and processing depend on the selected fuzzy technique and on the problem to be solved.

- Fuzzy image processing has three main stages: image fuzzification, modification of membership values, and, if necessary, image defuzzification.

Fuzzy c-means clustering

- In fuzzy clustering, every point has a degree of belonging to clusters, as in fuzzy logic, rather than belonging completely to just one cluster. Thus, points on the edge of a cluster, may be *in the cluster* to a lesser degree than points in the center of cluster. An overview and comparison of different fuzzy clustering algorithms is available.

Fuzzy c-means clustering

- The FCM algorithm is one of the most widely used fuzzy clustering algorithms. This technique was originally introduced by Professor Jim Bezdek in 1981.
- The FCM algorithm attempts to partition a finite collection of elements X={,, ...,} into a collection of c fuzzy clusters with respect to some given criterion.
- The algorithm is based on minimization of the following objective function:

$$J_m = \sum_{i=1}^{N} \sum_{j=1}^{C} u_{ij}^m \left\| x_i - c_j \right\|^2$$

- where *m* (the Fuzziness Exponent) is any real number greater than 1, *N* is the number of data, *C* is the number of clusters, u_{ij} is the degree of membership of x_i in the cluster *j*, x_i is the *i*th of d-dimensional measured data, c_j is the d-dimension center of the cluster, and ||*|| is any norm expressing the similarity between any measured data and the center.

Fuzzy c-means clustering

- Given a finite set of data, the algorithm returns a list of c cluster centers V, such that

  $V = v_{i,}\ i = 1, 2, \ldots, c$

- and a membership matrix U such that

  $U = u_{ij}\ ,\ i = 1, \ldots, c\ ,\ j = 1, \ldots, n$

- Where u_{ij} is a numerical value in [0, 1] that tells the degree to which the element x_j belongs to the i-th cluster.
- Summation of membership of each data point should be equal to one.

Parameters of the FCM algorithm

Before using the FCM algorithm, the following parameters must be specified:

- the number of clusters, c,
- the fuzziness exponent, m, where m is any real number greater than 1

Steps:

- Step1: choose random centroid at least 2
- Step2: compute membership matrix.

$$u_{ij} = \frac{1}{\sum_{k=1}^{c} \left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|} \right)^{\frac{2}{m-1}}}$$

$$= \frac{1}{\left(\frac{\|x_i - c_j\|}{\|x_i - c_1\|}\right)^{2/(m-1)} + \left(\frac{\|x_i - c_j\|}{\|x_i - c_2\|}\right)^{2/(m-1)} + \cdots + \left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|}\right)^{2/(m-1)}}$$

where $||x_i - c_j||$ is the Distance from point *i* to current cluster centre *j*,

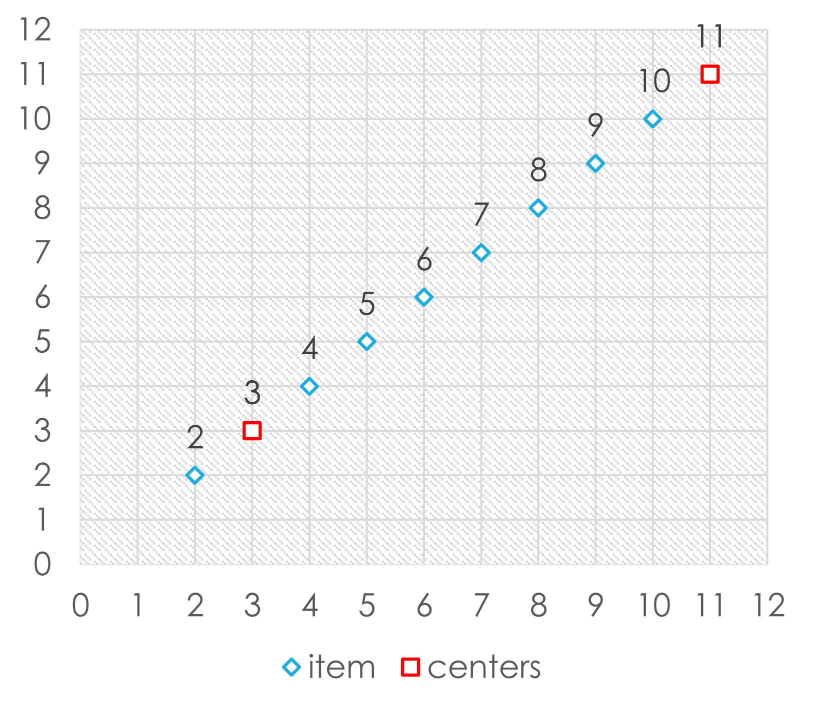
$||x_i - c_k||$ is the Distance from point *i* to other cluster centers *k*.(note if we have 2D data we use euclidean distance).

- Step3: calculate the c cluster centers.

$$c_j = \frac{\sum_{i=1}^{N} u_{ij}^{m} \cdot x_i}{\sum_{i=1}^{N} u_{ij}^{m}}$$

Example:

- Let x=[2 3 4 5 6 7 8 9 10 11] , m=2, number of cluster C=2, c_1=3 , c_2=11.
- Step 1: for first iteration calculate membership matrix.
- For node 2 (1st element):



$$U11 = \frac{1}{\left(\frac{2-3}{2-3}\right)^{\frac{2}{2-1}} + \left(\frac{2-3}{2-11}\right)^{\frac{2}{2-1}}} = \frac{1}{1+\frac{1}{81}} = \frac{81}{82} = 98.78\%$$

The membership of first node to first cluster

$$U12 = \frac{1}{\left(\frac{2-11}{2-3}\right)^{\frac{2}{2-1}} + \left(\frac{2-11}{2-11}\right)^{\frac{2}{2-1}}} = \frac{1}{81+1} = \frac{1}{82} = 1.22\%$$

The membership of first node to second cluster

- For node 3 (2^{nd} element):

  U21 = 100%
  The membership of second node to first cluster

  U22 = 0%

  The membership of second node to second cluster

- For node 4 (3rd element):

$$U31 = \frac{1}{\left(\frac{4-3}{4-3}\right)^{\frac{2}{2-1}} + \left(\frac{4-3}{4-11}\right)^{\frac{2}{2-1}}} = \frac{1}{1+\frac{1}{49}} = \frac{1}{\frac{50}{49}} = 98\%$$

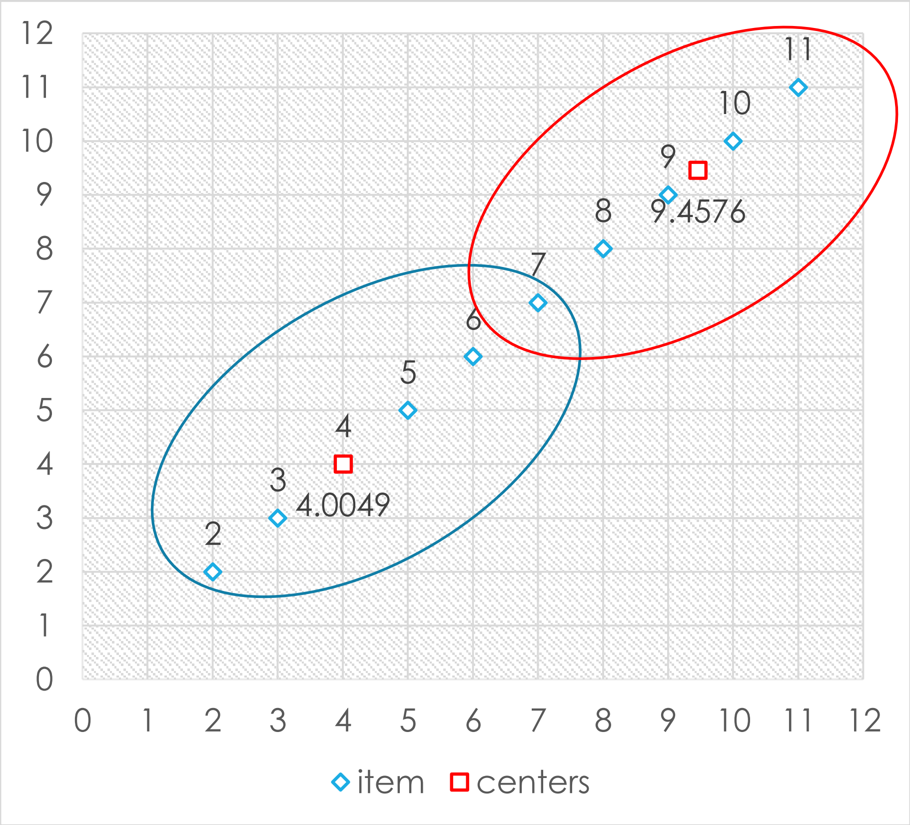
The membership of first node to first cluster

$$U32 = \frac{1}{\left(\frac{4-11}{4-3}\right)^{\frac{2}{2-1}} + \left(\frac{4-11}{4-11}\right)^{\frac{2}{2-1}}} = \frac{1}{49+1} = \frac{1}{50} = 2\%$$

The membership of first node to second cluster

- And so on until we complete the set and get U matrix

X	cluster1	cluster2
2	0.9878	0.0122
3	1.0000	0
4	0.9800	0.0200
5	0.9000	0.1000
6	0.7353	0.2647
7	0.5000	0.5000
8	0.2647	0.7353
9	0.1000	0.9000
10	0.0200	0.9800
11	0	1.0000



- Step2: now we compute new centers

$$c_j = \frac{\sum_{i=1}^{N} u_{ij}^{m} \cdot x_i}{\sum_{i=1}^{N} u_{ij}^{m}}$$

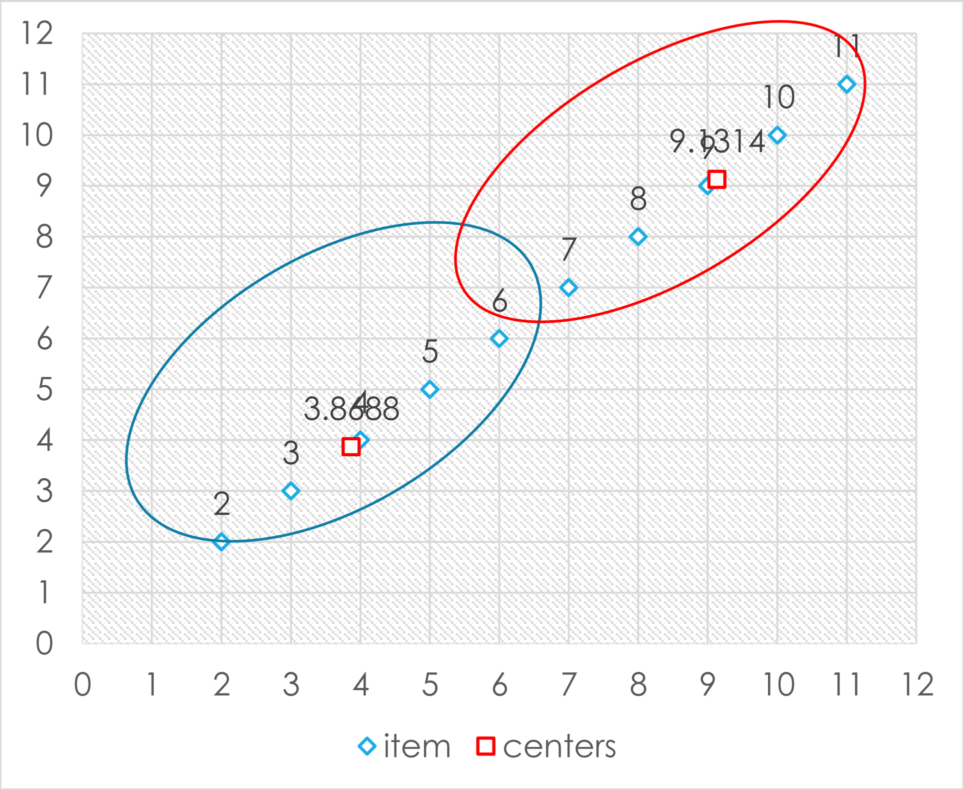
$$c1= \frac{(98.78\%)^2 * 2 + (100\%)^2 * 3 + (98\%)^2 * 4 + (50\%)^2 * 7 + \ldots}{(98.78\%)^2 + (100\%)^2 + (98\%)^2 + (50\%)^2 + \ldots} = 4.0049$$

And c2=9.4576

- **Repeat step until there is visible change.**
- Final iteration :
- U =

X	cluster1	cluster2
2	0.9357	0.0643
3	0.9803	0.0197
4	0.9993	0.0007
5	0.9303	0.0697
6	0.6835	0.3165
7	0.3167	0.6833
8	0.0698	0.9302
9	0.0007	0.9993
10	0.0197	0.9803
11	0.0642	0.9358

- c1 = 3.8688
- c2 = 9.1314



Matlab implementation:

```
x=[2 3 4 5 6 7 8 9 10 11];
c1=3;c2=11;
for j=1:1
k=1;
for i=1:length(x)
    u1(k)=1/(((x(i)-c1)/(x(i)-c1))^2+((x(i)-c1)/(x(i)-c2))^2);
    if(isnan(u1(k)))
        u1(k)=1;
    end
    u2(k)=1/(((x(i)-c2)/(x(i)-c1))^2+((x(i)-c2)/(x(i)-c2))^2);
     if(isnan(u2(k)))
        u2(k)=1;
    end
    k=k+1;
end
u=[u1;u2;u1+u2]'
c11=sum((u1.^2).*x)/sum(u1.^2)
c22=sum((u2.^2).*x)/sum(u2.^2)
c1=c11;c2=c22;
end
```

Output for first iteration:

u =

cluster1	cluster2	summation
0.9878	0.0122	1.0000
1.0000	0	1.0000
0.9800	0.0200	1.0000
0.9000	0.1000	1.0000
0.7353	0.2647	1.0000
0.5000	0.5000	1.0000
0.2647	0.7353	1.0000
0.1000	0.9000	1.0000
0.0200	0.9800	1.0000
0	1.0000	1.0000

c11 =

4.0049

c22 =

9.4576

Advantages:

1. Gives best result for overlapped data set and comparatively better than k-means algorithm.
2. Unlike k-means where data point must exclusively belong to one cluster center here data point is assigned membership to each cluster center as a result of which data point may belong to more than one cluster center.

- Demo on FCM check this link : http://aydos.com/fcm