

# Fuzzy C-Means

## Clustering

# Fuzzy C-Means

- ▶ *Fuzzy C-Means* merupakan salah satu metode dari *fuzzy clustering*.
- ▶ Metode ini dikembangkan Dunn (1973) dan diperbaiki Bezdek (1981) sebagai metode yang sering digunakan dalam pengenalan pola (*pattern recognition*).

# Clustering

- ▶ Pengelompokan data pada FCM ditentukan berdasarkan derajat keanggotaan yang bernilai antara 0 dan 1.

# Algoritma

1. Masukkan input data, dan tentukan parameter yang terlibat. ( matriks ukuran  $n \times m$  ( $n =$  jml sampel data,  $m =$  atribut data), jml cluster ( $c$ ), tingkat kekaburan/fuzzy ( $w$ ), MaxIterasi, akurasi ( $e$ ), fungsi objektif ( $P_0 = 0$ ), iterasi awal  $t = 1$ ).
2. Bangkitkan bilangan random sebagai derajat keanggotaan awal ( $\mu$ ).
3. Hitung pusat cluster tiap cluster.

$$V_{kj} = \frac{\sum_{i=1}^n ((\mu_{ik})^w \times x_{ij})}{\sum_{i=1}^n (\mu_{ik})^w}$$

# Algoritma (cont'd)

5. Hitung fungsi objektif pada iterasi ke -t.

$$P_t = \sum_{i=1}^n \sum_{k=1}^c \left( \left[ \sum_{j=1}^m (x_{ij} - v_{kj})^2 \right] (\mu_{ik})^w \right)$$

6. Update derajat keanggotan  $\mu$ .

$$\mu_{ik} = \frac{\left[ \sum_{j=1}^m (x_{ij} - v_{kj})^2 \right]^{\frac{-1}{w-1}}}{\sum_{k=1}^c \left[ \sum_{j=1}^m (x_{ij} - v_{kj})^2 \right]^{\frac{-1}{w-1}}}$$

Dengan  $i = 1 \dots n$ ; dan  $k = 1, 2, \dots, c$ .

# Algoritma (cont'd)

7. Cek kondisi berhenti,

a) Jika  $(P_t - P_{t-1} < \varepsilon)$  atau  $(t < \text{MaxIterasi})$   
maka berhenti.

b) Jika tidak:  $t=t+1$ , ulangi langkah ke 3.

c) Jika  $\text{maks}(|U_t - U_{t-1}|) < \epsilon$

# Contoh

- ▶ Misalkan terdapat 5 buah sampel data dalam vektor dua dimensi.

<b>Data(i)</b>	<b><math>x_{i1}</math></b>	<b><math>x_{i2}</math></b>
<b>1</b>	<b>10</b>	<b>8</b>
<b>2</b>	<b>4</b>	<b>5</b>
<b>3</b>	<b>2</b>	<b>3</b>
<b>4</b>	<b>9</b>	<b>7</b>
<b>5</b>	<b>0</b>	<b>1</b>

- ▶ Parameter lain: jml cluster=2,  $w=2$ ,  
MaxIterasi=10, Akurasi( $\epsilon$ ) = 0.1,  $P_0=0$ ,  $t = 1$

# Langkah algoritma

- ▶ (2) Bangkitkan bil. Random untuk U. Matriks ini menunjukkan derajat keanggotaan data ke-i pada cluster ke k .

$$U = \begin{pmatrix} 0.3 & 0.7 \\ 0.6 & 0.4 \\ 0.7 & 0.3 \\ 0.4 & 0.6 \\ 0.8 & 0.2 \end{pmatrix}$$



# Langkah algoritma (cont'd)

- ▶ (3) Menghitung pusat cluster.

Data	$\mu_{ij}$		$x_{ij}$		$(\mu_{i1})^2$	$(\mu_{i2})^2$	$(\mu_{i1})^2 * x_{i1}$	$(\mu_{i1})^2 * x_{i2}$	$(\mu_{i2})^2 * x_{i1}$	$(\mu_{i2})^2 * x_{i2}$
	1	2	1	2						
1	0.3	0.7	10	8	0.09	0.49	0.9	0.72	4.9	3.92
2	0.6	0.4	4	5	0.36	0.16	1.44	1.8	0.64	0.8
3	0.7	0.3	2	3	0.49	0.09	0.98	1.47	0.18	0.27
4	0.4	0.6	9	7	0.16	0.36	1.44	1.12	3.24	2.52
5	0.8	0.2	0	1	0.64	0.04	0	0.64	0	0.04
$\Sigma$					1.74	1.14	4.76	5.75	8.96	7.52

$$V_{kj} = \begin{pmatrix} 2.736 & 3.305 \\ 7.860 & 6.596 \end{pmatrix}$$

Dimana k = cluster, j = byk data

# Langkah algoritma (cont'd)

- ▶ (5) menghitung fungsi objektif pd iterasi 1.

$$P_t = \sum_{i=1}^n \sum_{k=1}^c \left( \left[ \sum_{j=1}^m (x_{ij} - v_{kj})^2 \right] (\mu_{ik})^w \right)$$

i	$(\mu_{i1})^2$	$(\mu_{i2})^2$			C1 + C2
			C1	C2	
1	0.09	0.49	11.31	3.21	14.52
2	0.36	0.16	1.984	2.792	5.406
3	0.49	0.09	0.311	4.254	4.565
4	0.16	0.36	9.204	1.359	10.563
5	0.64	0.04	7.013	3.724	10.737
fungsi obyektif = $\Sigma$					45.431

$$C1 = \left[ \sum_{j=1}^2 (x_{ij} - v_{1j})^2 \right] \mu_{i1}^w$$

$$C2 = \left[ \sum_{j=1}^2 (x_{ij} - v_{2j})^2 \right] \mu_{i2}^w$$

Jadi  $P_1 = 45.431$ .

# Langkah algoritma (cont'd)

- ▶ (6) update derajat keanggotaan ( $\mu$ ).

$$\mu_{ik} = \frac{\left[ \sum_{j=1}^m (x_{ij} - v_{kj})^2 \right]^{\frac{-1}{w-1}}}{\sum_{k=1}^c \left[ \sum_{j=1}^m (x_{ij} - v_{kj})^2 \right]^{\frac{-1}{w-1}}}$$

i	$(x_{i1} - v_{11})^2$	$(x_{i2} - v_{12})^2$	$(x_{i1} - v_{21})^2$	$(x_{i2} - v_{22})^2$	LT	$\mu_{i1}$	$\mu_{i2}$
	L1	L2	L3	L4	L1+L2+L3+L4	(L1+L2)/LT	(L3+L4)/LT
1	52.766	22.043	4.580	1.971	81.36	0.919	0.081
2	1.598	2.873	14.900	2.547	21.918	0.204	0.796
3	0.542	0.093	34.340	12.931	47.906	0.013	0.987
4	39.238	13.653	1.300	0.163	54.354	0.973	0.027
5	7.486	5.313	61.780	31.315	105.894	0.121	0.879

$$U = \begin{pmatrix} 0.919 & 0.081 \\ 0.204 & 0.796 \\ 0.013 & 0.987 \\ 0.973 & 0.027 \\ 0.121 & 0.879 \end{pmatrix}$$

- ▶ Berikutnya kita cek kondisi berhenti. Karena
$$[P_1 - P_0] = 45.431 - 0 = 0$$
berarti  $[P_1 - P_0] > \varepsilon$  dan Iterasi = 1 < MaxIterasi (= 10) maka dilanjutkan ke iterasi ke-2 (t=2). Kembali ke algoritma FCM langkah 3 dan seterusnya.
- ▶ Setelah dilakukan 5 kali iterasi maka akan didapat hasil seperti berikut.

- ▶ Setelah dilakukan 5 kali iterasi maka akan didapat hasil seperti berikut:

$$V_{kj} = \begin{pmatrix} 2.736 & 3.305 \\ 7.860 & 6.596 \end{pmatrix}$$

$$[P_5 - P_4] = 0.079 (< \varepsilon)$$

$$U = \begin{pmatrix} 0.992 & 0.008 \\ 0.225 & 0.775 \\ 0.002 & 0.998 \\ 0.996 & 0.004 \\ 0.046 & 0.954 \end{pmatrix}$$

- ▶ Dari matriks U tersebut dapat diperoleh informasi mengenai kecenderungan suatu data untuk masuk ke dalam *cluster* yang mana.

Data (i)	Cluster	
	1	2
1	X	
2		X
3		X
4	X	
5		X

# Pengenalan pola

- ▶ Setelah melakukan clustering dari data set → fase learning. Tahap berikutnya adalah menguji dengan data uji/test.
- ▶ Untuk mengetahui data uji masuk dalam cluster mana, kita dapat mengetahuinya dari jarak euclidean antara data uji dengan titik pusat cluster.