Prosiding
ANNUAL RESEARCH SEMINAR 2016
6 Desember 2016, Vol 2 No. 1

ISBN : 979-587-626-0 | UNSRI http://ars.ilkom.unsri.ac.id

Analisis Perbandingan Algoritma Fuzzy C-Means dan K-Means

Yohannes
Teknik Informatika
STMIK GI MDP Palembang, Indonesia
yohannesmasterous@mdp.ac.id

***Abstrak*— Klasterisasi merupakan teknik pengelompokkan data berdasarkan kemiripan data. Teknik klasterisasi banyak digunakan pada bidang ilmu komputer khususnya pengolahan citra, pengenalan pola, dan *data mining*. Algoritma yang sering digunakan untuk klasterisasi data pada umumnya adalah *Fuzzy C-Means* dan *K-Means*. *Fuzzy C-Means* merupakan algoritma klasterisasi dimana data dikelompokkan ke dalam suatu pusat *cluster* data dengan derajat keanggotaan masing-masing *cluster*. Sedangkan *K-Means* merupakan teknik mengelompokkan data dengan mempartisi data ke dalam beberapa *cluster* dengan menetapkan sejumlah objek data terdekatnya. Pada penelitian ini akan dilakukan perbandingan algoritma *Fuzzy C-Means* dan *K- Means* dalam hal klasterisasi data dengan jumlah *cluster* dan jumlah data yang berbeda. Hasil eksperimen menunjukkan bahwa *K-Means* lebih cepat dibandingkan dengan *Fuzzy C-Means* dari segi waktu klasterisasi. Sedangkan *Fuzzy C-Means* lebih baik dalam hal komputasi untuk mencari derajat keanggotaan masing- masing *cluster* dalam pengelompokan data. Hasil klasterisasi untuk setiap data dan jumlah *cluster* pada masing-masing algoritma bervariasi sesuai kompleksitasnya.**

Kata Kunci*—*fuzzy c-means; k-means; klasterisasi; cluster data

I. PENDAHULUAN

Klasterisasi adalah sebuah teknik untuk mengelompokkan data berdasarkan kemiripan data. Klasterisasi berbeda dengan klasifikasi, dimana pengelompokan data dilakukan tanpa berdasarkan kelas atau kelompok tertentu. Tidak adanya variabel acuan dalam proses klasterisasi, membuat klasterisasi dapat dipakai untuk memberikan label pada kelompok data yang belum diketahui sebelumnya.

Klasterisasi merupakan proses pengelompokkan yang tidak melakukan klasifikasi, estimasi, atau memprediksi nilai dari variabel acuan, tetapi membagi keseluruhan data menjadi kelompok yang memiliki kemiripan. Kemiripan data dalam satu kelompok akan bernilai maksimum sedangkan kemiripan data antar kelompok akan bernilai minimum. Klasterisasi dapat digunakan untuk mengetahui struktur dalam data yang dapat dipakai lebih lanjut dalam berbagai aplikasi secara luas.

Di bidang informatika, teknik klasterisasi digunakan untuk pengolahan citra, pengenalan pola, dan pengolahan pada *data mining*. Klasterisasi dapat diterapkan ke dalam data yang bersifat kuantitatif, kualitatif, atau gabungan dari keduanya Pada dasarnya, algoritma klasterisasi menghitung jarak setiap data dengan pusat data untuk mengukur kemiripan antar data.

Banyak algoritma yang digunakan untuk klasterisasi data. Salah satu jenis algoritma klasterisasi yang dikenal adalah metode partisi (*partitional clustering*) dimana harus menentukan jumlah partisi yang diinginkan kemudian setiap data diuji untuk dimasukkan pada salah satu partisi. *Partitional clustering* sering digunakan pada algoritma klasterisasi untuk mengklasterisasi data. Dua diantaranya adalah *Fuzzy C-Means* dan *K-Means*.

Algoritma *Fuzzy C-Means* merupakan algoritma klasterisasi dimana data dikelompokkan ke dalam suatu pusat *cluster* data dengan derajat keanggotaan masing-masing *cluster*. Algoritma *K-Means* merupakan teknik mengelompokkan data dengan mempartisi data ke dalam beberapa *cluster* dengan menetapkan sejumlah objek data terdekatnya. Kedua algoritma klasterisasi ini akan dibandingkan untuk proses klasterisasi dengan jumlah *cluster* yang berbeda dan dengan jumlah data yang berbeda. Selain itu juga dibandingkan bagaimana waktu klasterisasi menggunakan algoritma *Fuzzy C-Means* dan *K-Means*. Tujuan penelitian ini adalah membandingkan algoritma *Fuzzy C-Means* dan *K-Means* dalam klasterisasi data dengan jumlah *cluster* dan jumlah data yang berbeda.

2.1. *Fuzzy C-Means*

Fuzzy C-Means ditemukan oleh Bezdek pada tahun 1981. *Fuzzy C-Means* adalah teknik pengklasteran data dimana keberadaan pada setiap titik data dikelompokkan dalam suatu *cluster* dengan derajat keanggotaan tertentu [1]. Pengelompokkan data dengan FCM menghasilkan keluaran berupa daftar pusat *cluster* dan beberapa fungsi keanggotaan untuk tiap data. Informasi ini digunakan dalam mendefinisikan fungsi-fungsi keanggotaan untuk mempresentasikan nilai *fuzzy* dari tiap *cluster*.

ISBN : 979-587-626-0 | UNSRI http://ars.ilkom.unsri.ac.id

Tetapkan matriks partisi f(c) awal sebarang, sebagai berikut :

$$\mu_f(c) = \begin{bmatrix} \mu_{11}[u_1] & \mu_{21}[u_1] & \dots & \mu_{c1}[u_1] \\ \mu_{12}[u_1] & \mu_{22}[u_2] & \dots & \mu_{c2}[u_2] \\ \vdots & \vdots & \vdots & \vdots \\ \mu_{1N}[u_1] & \mu_{2N}[u_N] & \dots & \mu_{cN}[u_{N1}] \end{bmatrix} \text{.....(1)}$$

1. Tetapkan nilai w > 1, ε sangat kecil, MaxIter, jumlah
2. cluster c > 1, dan $t=0$
3. Tetapkan fungsi obyektif awal: Pt(c) secara random
4. Naikkan nomor iterasi: $t=t+1$
5. Hitung pusat vektor tiap-tiap cluster untuk matriks partisi tersebut sebagai berikut:

$$v_{fi} = \frac{\sum_{k=1}^{N}(\mu_{ik})^w u_k}{\sum_{k=1}^{N}(\mu_{ik})^w} \text{..... (2)}$$

6. Modifikasi tiap nilai keanggotaan sebagai berikut:
   a. Jika $y_k \neq v_{fi}$,

$$\mu_{ik}(y_k) = \left[\sum_{g=1}^{c}\left(\frac{|u_k - v_{fi}|^2}{|u_k - v_{gi}|^2}\right)^{1/(w-1)}\right]^{-1} \text{..... (3)}$$

   b. Jika $y_k = v_{fi}$,
   $\mu_{ik}(y_k) = 1$, jika i=g;
   $\mu_{ik}(y_k) = 0$, jika i≠g;

7. Modifikasi tiap nilai keanggotaan sebagai berikut:

$$\mu_f(c) = \begin{bmatrix} \mu_{11}[u_1] & \mu_{21}[u_1] & \dots & \mu_{c1}[u_1] \\ \mu_{12}[u_1] & \mu_{22}[u_2] & \dots & \mu_{c2}[u_2] \\ \vdots & \vdots & \vdots & \vdots \\ \mu_{1N}[u_1] & \mu_{2N}[u_N] & \dots & \mu_{cN}[u_N] \end{bmatrix} \text{..... (4)}$$

8. Menghitung fungsi objektif :

$$P_t(c) = \sum_{k=1}^{N}\sum_{i=1}^{c}(\mu_{ik})^w |y_k - v_f i|^2 \text{..... (5)}$$

9. Cek kondisi untuk berhenti, yaitu :

$$\left(|P_t(c) - P_{t-1}(c)| < \varepsilon\right) \text{ atau } \left(t > MaxIter\right) \text{..... (6)}$$

   jika iya berhenti, dan jika tidak, ulangi kembali ke langkah-4.

2.1. *K-Means*

K-Means ditemukan oleh MacQueen pada tahun 1967. *K- Means* adalah salah satu teknik pengelompokkan yang paling banyak digunakan karena kesederhanaan dan kecepatan [3]. Algoritma ini mempartisi data ke dalam *k cluster* dengan menetapkan setiap objek *cluster centroid* terdekatnya (nilai rata- rata dari variabel untuk semua objek dalam *cluster* tertentu) berdasarkan ukuran jarak yang digunakan. Hal ini lebih kuat untuk berbagai jenis variabel.

Langkah-langkah algoritma K-Means [4] :

1. Tentukan jumlah cluster k
2. Pilih pusat cluster k (pemilihan objek k dari data set dilakukan secara random)
3. Hitung jarak setiap objek terhadap pusat cluster.
4. Hitung kembali jarak terdekat setiap objek terhadap pusat cluster.
5. Ulangi langkah 3 dan 4 sampai (selisih perhitungan jarak) atau iterasi maksimum tercapai.

Partitioning pada algoritma K-Means digunakan untuk menentukan jumlah cluster akhir terlebih dahulu. Beberapa masalah pada algoritma K-Means seperti kerentanan terhadap optimalisasi, kepekaan, ruang memori, dan jumlah iterasi yang diperlukan untuk klasterisasi data tidak diketahui. Kompleksitas waktu algoritma K-Means adalah ( ) sedangkan kompleksitas waktu algoritma Fuzzy C-Means adalah (2)dimana *c* adalah jumlah *cluster*, n adalah banyaknya data yang akan diklasterisasi, *d* merupakan bobot *cluster*, dan *i* adalah jumlah iterasi yang diperlukan dalam proses klasterisasi.

Klasterisasi data dengan menggunakan *cluster* tertentu dapat diasumsikan bahwa algoritma *K-Means* lebih baik dari algoritma *Fuzzy C-Means* [5]. *Fuzyy C-Means* menghasilkan hasil klasterisasi yang dekat dengan pengelompokan data tetapi masih membutuhkan waktu yang lebih lama jika dibandingkan dengan *K-Means* karena keterlibatan perhitungan derajat keanggotaan *fuzzy* dalam klasterisasi data [6].

Fuzzy C-Means merupakan teknik pengelompokan yang cocok untuk kemampuan dalam pengenalan pola, data yang tidak lengkap, informasi campuran, dan dapat memberikan solusi perkiraan lebih cepat. Algoritma *K-Means* tampaknya unggul dibandingkan algoritma *Fuzzy C-Means* [6], [5]. Namun klasterisasi yang dilakukan [6] hanya untuk satu jenis *cluster* (*cluster* = 3) pada satu data. Di sisi lain, pada penelitian [7] perbandingan klasterisasi yang dilakukan hanya terbatas, yaitu *n* = 100. Pada penelitian [6], [7] tidak mengemukakan secara jelas *processo*r dan memori yang digunakan dalam percobaannya.

Algoritma *K-Means* merupakan algoritma klasterisasi yang populer karena kesederhanaannya. Perhitungan jarak sangat berperan penting dalam performa algoritma ini. Terdapat beberapa teknik perhitungan jarak pada setiap data dalam proses klasterisasi *K-Means* diantaranya adalah *city block distance*, *euclidean*, *cosine*, dan *correlation*. Keempat teknik ini mempengaruhi dalam hal klasterisasi khususnya dalam waktu klasterisasi. Dari keempat teknik tersebut, *city block distance* menunjukkan performa klasterisasi yang

ISBN : 979-587-626-0 | UNSRI http://ars.ilkom.unsri.ac.id

lebih baik dari segi waktu. Sedangkan *cosine* membutuhkan waktu yang lebih lama dari ketiga teknik lainnya [8].

II. SKENARIO EKSPERIMEN

Data yang digunakan dalam uji coba adalah data yang dibuat secara *random* dengan interval nilai [0,1]. Data uji coba yang akan digunakan ada tiga, yaitu data yang memiliki jumlah elemen sebanyak 100, 1.000, dan 10.000. Uji coba dilakukan dengan menggunakan *processor Core i7* 2.00 GHz dan RAM 4.00 GB dengan sistem operasi *Windows 8.1*. Eksperimen dilakukan dengan menggunakan MATLAB R2014a.

III. EKSPERIMEN

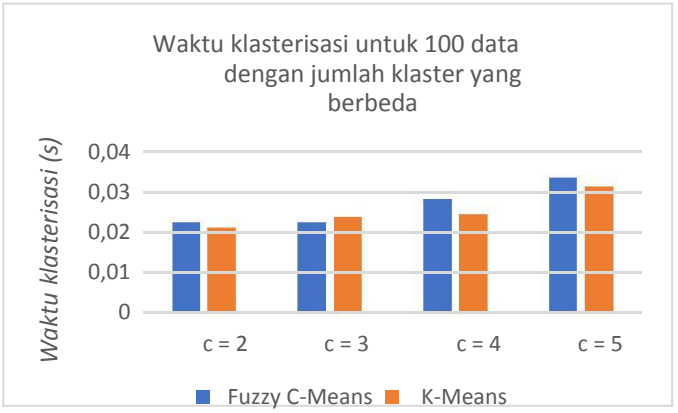
Uji coba pertama adalah melakukan klasterisasi untuk masing-masing algoritma *Fuzzy C-Means* dan *K-Means* terhadap data yang memiliki 100 elemen dengan = 2, = 10^{-5}, = 2 . . 5. Uji coba kedua adalah melakukan klasterisasi untuk masing-masing algoritma *Fuzzy C-Means* dan *K-Means* terhadap 1.000 data dan 10.000 data dengan maksimum iterasi sebanyak 100, nilai pembobot (w = 2), dan = 10^{-5}. Uji coba pertama dapat dilihat pada Tabel 1.

TABEL 1. PERBANDINGAN WAKTU DAN ITERASI DALAM KLASTERISASI FUZZY C-MEANS DAN K-MEANS UNTUK 100 DATA DENGAN JUMLAH *CLUSTER* (C) YANG BERBEDA

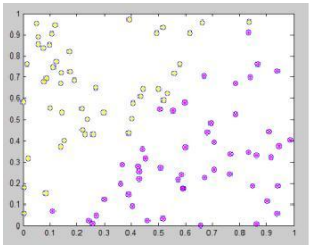
	c = 2		c = 3		c = 4		c =5	
	waktu	iterasi	waktu	iterasi	waktu	iterasi	waktu	iterasi
Fuzzy C-Means	0.022446	22	0.022528	36	0.028109	31	0.033485	35
K-Means	0.021028	9	0.023828	6	0.024455	7	0.031242	7

Pada Tabel 1 dapat dilihat untuk jumlah *cluster* sebanyak 2, waktu klasterisasi algoritma *Fuzzy C-Means* adalah 0.022446 detik dengan 22 iterasi sedangkan pada algoritma *K-Means* membutuhkan waktu 0.021028 detik dengan 9 iterasi. Untuk jumlah *cluster* sebanyak 3, waktu klasterisasi algoritma *Fuzzy C-Means* adalah 0.022528 detik dengan 36 iterasi sedangkan pada algoritma *K-Means* membutuhkan waktu 0.023828 detik dengan 6 iterasi. Hal ini menunjukkan bahwa algoritma *K-Means* lebih cepat dalam proses klasterisasi dengan jumlah iterasi yang lebih sedikit dibandingkan dengan algoritma *Fuzzy C-Means*.

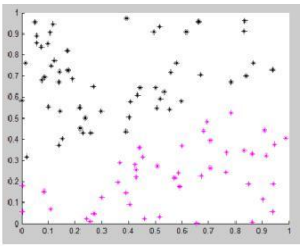
Berdasarkan pada Tabel 1, semakin banyak jumlah *cluster* yang digunakan maka waktu yang diperlukan semakin bertambah untuk masing-masing algoritma. Pada algoritma *Fuzzy C-Means* dan *K-Means*, jumlah iterasi berubah-ubah untuk jumlah *cluster* yang berbeda. Namun secara umum untuk algoritma *Fuzzy C-Means*, jumlah iterasi semakin bertambah ketika jumlah *cluster* yang digunakan semakin besar. Grafik waktu klasterisasi *Fuzzy C-Means* dan *K-Means* dapat dilihat pada Gambar 1.



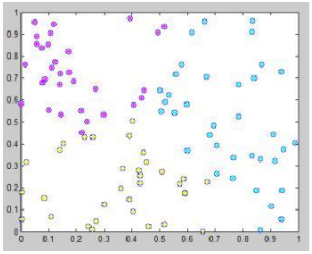
Gambar 1. Grafik waktu klasterisasi untuk 100 data dengan jumlah *cluster* yang berbeda



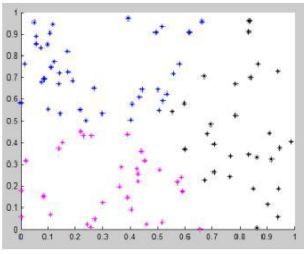
Gambar 2.1 *FCM* 100 data dengan c = 2



Gambar 2.2 *K-Means* 100 data dengan c = 2

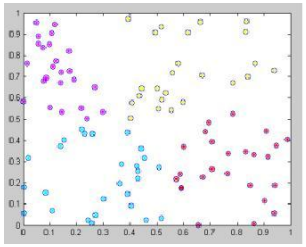


Gambar 2.3 *FCM* 100 data dengan c = 3

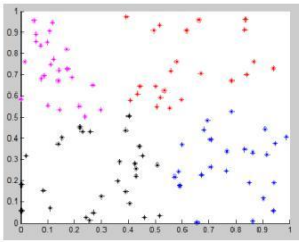


Gambar 2.4 *K-Means* 100 data dengan c = 3

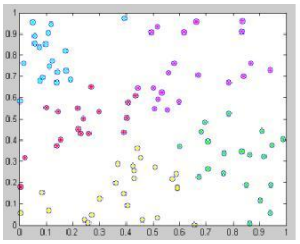
ISBN : 979-587-626-0 | UNSRI http://ars.ilkom.unsri.ac.id



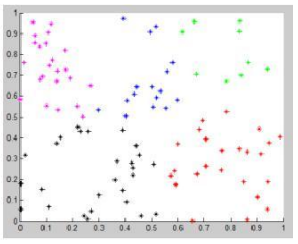
Gambar 2.5 *FCM* 100 data dengan c = 4



Gambar 2.6 *K-Means* 100 data dengan c = 4



Gambar 2.7 *FCM* 100 data dengan c = 5

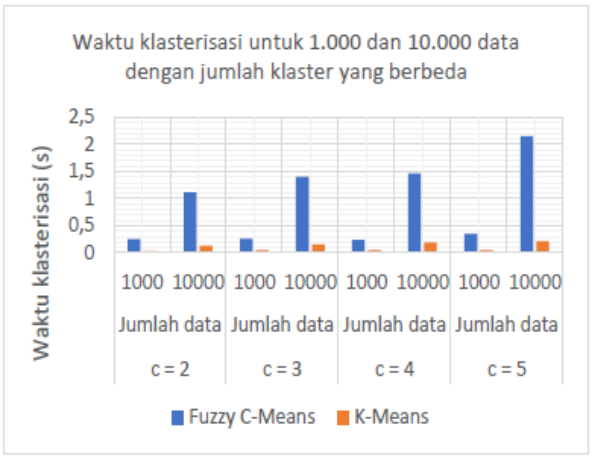


Gambar 2.8 *K-Means* 100 data dengan c = 5

Pada Gambar 2, dapat dilihat hasil klasterisasi untuk masing-masing algoritma *Fuzzy C-Means* dan *K-Means*. Perbandingan waktu dan iterasi dalam klasterisasi *Fuzzy C-Means* dan *K-Means* dapat dilihat pada Tabel 2.

TABEL 2. Perbandingan Waktu Dan Iterasi Dalam Klasterisasi Fuzzy C-Means Dan K-Means Untuk 1.000 Dan 10.000 Data Dengan Jumlah *Cluster* (C) Yang Berbeda Maxiter = 100

	c = 2		c = 3		c = 4		c =5	
	Jumlah data		Jumlah data		Jumlah data		Jumlah data	
	waktu	iterasi	waktu	iterasi	waktu	iterasi	waktu	iterasi
Fuzzy C-Means	0.2454 29	1.1154 11	0.2612 69	1.3988 05	0.2292 08	1.4685 75	0.3467 89	2.1598 15
K-Means	0.0304 46	0.1232 48	0.0404 03	0.1530 83	0.0416 49	0.1844 90	0.0455 32	0.2138 55



Gambar 3. Grafik waktu klasterisasi untuk 1.000 dan 10.000 data dengan *cluster* berbeda

Kompleksitas waktu untuk algoritma *fuzzy c-means* adalah (^{2}) sedangkan *K-Means* adalah ( ). Berdasarkan uji coba untuk n = 1.000 dan 10.000, = 2, 2 ≤ ≤ 5, = 100, dapat dihitung kompleksitas waktu dalam proses klasterisasi dimana n adalah jumlah data, d adalah nilai pembobot, c adalah jumlah *cluster*, dan i adalah iterasi. Perbandingan kompleksitas waktu *Fuzzy C-Means* dan *K-Means* dapat dilihat pada Tabel 3.

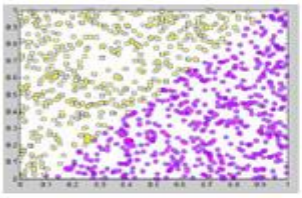
Tabel 3. Perbandingan Kompleksitas Waktu Fuzzy C-Means Dan K-Means

	c = 2		c = 3		c = 4		c =5	
	Jumlah data		Jumlah data		Jumlah data		Jumlah data	
	1.000	10.000	1.000	10.000	1.000	10.000	1.000	10.000
Fuzzy C-Means $O(ndic^2)$	800.000	8.000.000	1.800.000	18.000.000	3.200.000	32.000.000	5.000.000	50.000.000
K-Means $O(ndic)$	400.000	4.000.000	600.000	6.000.000	800.000	8.000.000	1.000.000	10.000.000

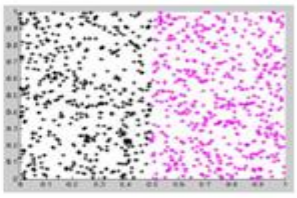
Berdasarkan kompleksitas waktu masing-masing algoritma, kompleksitas waktu algoritma *k-means* lebih cepat dibandingkan dengan algoritma *fuzzy c-means*. Kompleksitas waktu *fuzzy c-means* adalah *c* kali lebih lama dari kompleksitas waktu algoritma *k-means* dimana *c* merupakan jumlah *cluster* data. Algoritma *fuzzy c-means* membutuhkan kompleksitas waktu berbentuk kuadrat dikarenakan untuk setiap *cluster* dihitung pusat *cluster* dan derajat keanggotaan masing-masing *cluster*. Pusat *cluster* dan derajat keanggotaan akan di-*update* sesuai dengan jumlah iterasi atau fungsi objektif yang lebih kecil dari . Untuk menghasilkan *update* dari pusat *cluster* dan derajat keanggotaan masing-masing *cluster* dibutuhkan waktu sebanyak ^{2} untuk jumlah data, nilai pembobot, dan iterasi tertentu dimana *c* merupakan jumlah *cluster*.

Sedangkan algoritma *k-means* membutuhkan kompleksitas waktu berbentuk linear dikarenakan untuk setiap *cluster* dihitung jarak terdekat setiap objek terhadap pusat *cluster*. Untuk menghitung jarak terdekat setiap objek terhadap pusat *cluster* dibutuhkan waktu sebanyak *c* untuk jumlah data, nilai pembobot, dan iterasi tertentu dimana *c* merupakan jumlah *cluster*. Algoritma *fuzzy c-means* baik dalam komputasi klasterisasi data namun tidak lebih baik untuk jumlah data yang besar. Algoritma *k-means* lebih sederhana dan baik dalam jumlah data yang besar. Hasil klasterisasi *Fuzzy C-Means* dan *K-Means* untuk 1.000 data dapat dilihat pada Gambar 4.

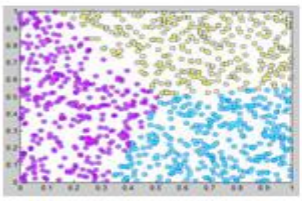
ISBN : 979-587-626-0 | UNSRI http://ars.ilkom.unsri.ac.id



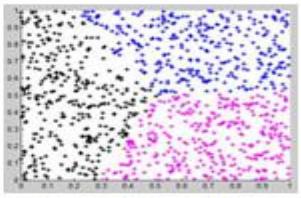
Gambar 4.1 *FCM* 1.000 data dengan c = 2



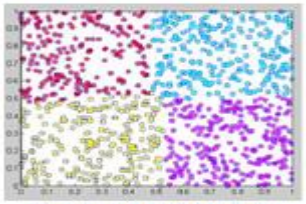
Gambar 4.2 *K-Means* 1.000 data dengan c = 2



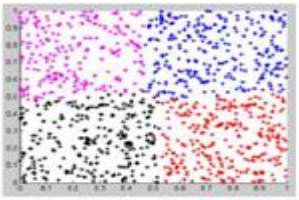
Gambar 4.3 *FCM* 1.000 data dengan c = 3



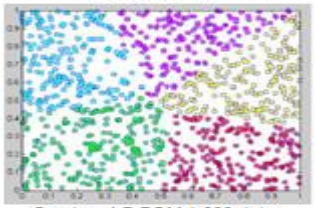
Gambar 4.4 *K-Means* 1.000 data dengan c = 3



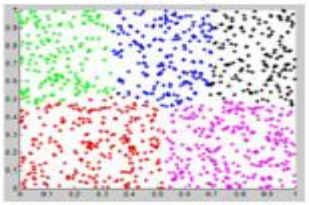
Gambar 4.5 *FCM* 1.000 data dengan c = 4



Gambar 4.6 *K-Means* 1.000 data dengan c = 4



Gambar 4.7 *FCM* 1.000 data dengan c = 5



Gambar 4.8 *K-Means* 1.000 data dengan c = 5

IV. KESIMPULAN

Berdasarkan kompleksitas waktu, algoritma *K-Means* lebih cepat dibandingkan dengan algoritma *Fuzzy C-Means* untuk jumlah data, *cluster*, nilai pembobot, dan iterasi yang sama. Hasil eksperimen menunjukkan untuk jumlah data 100, 1.000, dan 10.000 dengan jumlah *cluster* mulai dari dua sampai dengan lima, algoritma *K-Means* membutuhkan waktu *cluster* yang lebih sedikit dibandingkan dengan algoritma *Fuzzy C-Means*. Berdasarkan hasil eksperimen juga didapatkan bahwa jumlah iterasi yang dibutuhkan algoritma *K-Means* lebih sedikit dibandingkan algoritma *Fuzzy C-Means*. Dari eksperimen yang dilakukan dapat disimpulkan bahwa algoritma *K-Means* lebih cepat dibandingkan dengan algoritma *Fuzzy C-Means* dari segi waktu klasterisasi. Sedangkan algoritma *Fuzzy C-Means* lebih baik dalam hal komputasi untuk mencari derajat keanggotaan masing-masing *cluster* dalam pengelompokan data.

REFERENSI

[1] K. Miyamoto, Sadaaki, Ichihashi, Hidetomo, Honda, *Algorithms For Fuzzy Clustering*. Springer, 2008.

[2] S. Kusumadewi, *Analisis & Desain Sistem Fuzzy Menggunakan Toolbox Matlab*. Graha Ilmu, 2003.

[3] J. Dean, *Big Data, Data Mining, and Machine Learning*. Wiley, 2014.

[4] C. D. Manning, P. Raghavan, and H. Schutze, *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[5] T. VELMURUGAN, "Performance Comparison between k-Means and Fuzzy C-Means Algorithms using Arbitrary Data Points," *Wulfenia J.*, vol. 19, no. 8, pp. 234–241, 2012.

[6] T. Singh and M. Mahajan, "Performance Comparison of Fuzzy C Means with Respect to Other Clustering Algorithm," *Int. J. Adv. Res. Comput. Sci. Softw. Eng.*, vol. 4, no. 5, pp. 89–93, 2014.

[7] S. Ghosh and S. K. Dubey, "Comparative Analysis of K-Means and Fuzzy C-Means Algorithms," *Int. J. Adv. Comput. Sci. Appl.*, vol. 4, no. 4, pp. 35–38, 2013.

[8] D. J. Bora and A. K. Gupta, "Effect of Different Distance Measures on the Performance of K-Means Algorithm : An Experimental Study in Matlab," *Int. J. Comput. Sci. Inf. Technol.*, vol. 5, no. 2, pp. 2501–2506, 2014.