

Optimasi Cluster Pada Fuzzy C-Means Menggunakan Algoritma Genetika Untuk Menentukan Nilai Akhir

Putri Elfa Mas`udia^{*1}, Retantyo Wardoyo²

¹Universitas Muhammadiyah Malang; Jl. Raya Tlogomas 246, Malang, (0341) 463513

²Jurusan Ilmu Komputer dan Elektronika, FMIPA UGM, Yogyakarta

e-mail: ^{*1}aishsalma@gmail.com, ²rw@ugm.ac.id

Abstrak

Nilai akhir mahasiswa dapat ditentukan dengan berbagai cara, beberapa diantaranya menggunakan range nilai, standart deviasi, dll. Dalam penelitian ini akan ditawarkan sebuah metode baru untuk menentukan nilai akhir mahasiswa menggunakan clustering dalam hal ini adalah Fuzzy C-Means.

Fuzzy C-Means digunakan untuk mengelompokkan sejumlah data dalam beberapa cluster. Tiap data memiliki derajat keanggotaan pada masing-masing cluster antara 0-1 yang diukur melalui fungsi objektif. Pada Fuzzy C-Means ini fungsi objektif diminimumkan menggunakan iterasi yang biasanya terjebak dalam optimum lokal. Algoritma genetika diharapkan dapat menangani masalah tersebut karena algoritma genetika berbasis evolusi yaitu dapat mencari individu terbaik melalui operasi genetika (seleksi, crossover, mutasi) dan dievaluasi berdasarkan nilai fitness.

Penelitian ini bertujuan untuk mengoptimasi titik pusat cluster pada Fuzzy C-Means menggunakan algoritma genetika. Hasilnya, bahwa dengan menggunakan GFS didapatkan fungsi objektif yang lebih kecil daripada menggunakan FCM, walaupun membutuhkan waktu yang relative besar. Meskipun selisih antara FCM dan GFS tidak terlalu besar namun hal tersebut berpengaruh pada anggota cluster

Kata kunci— clustering, Fuzzy C-Means, algoritma genetika

Abstract

The final grade of students could be determined in various ways, some of which use a range of values, deviation standard, etc. In this study will be offered a new method for determining final grades of students by using the clustering method. In this research the clustering method that will be used is the Fuzzy C-Means (FCM).

Fuzzy C-Means is used to group a number of data in multiple clusters. Each data has a degree of membership (the range value of membership degree is 0-1). Membership degree is measured through the objective function. In Fuzzy C-Means, objective function is minimized by using iteration and is usually trapped in a local optimum. Genetic algorithm is expected to handle these problems. The operation of genetic algorithm based on evolution that is able to find the best individuals through genetic operations (selection, crossover and mutation) and evaluated based on fitness values.

This research aims to optimize the cluster center point of FCM by using genetic algorithms. The result of this research shows that by combining the Genetic Algorithm with FCM could obtained a smaller objective function than using FCM, although it takes longer in execution time. Although the difference of objective function that produced by FCM and FCM-Genetic Algorithm combination is not too big each other, but it takes effect on the cluster members.

Keywords— clustering, fuzzy c-means, genetic algorithm

1. PENDAHULUAN

Terdapat berbagai cara untuk menentukan nilai akhir huruf mahasiswa, diantaranya adalah menggunakan skala pasti maupun standar deviasi, Tapi dengan cara yang seperti itu mungkin akan menyulitkan dosen jika terjadi kondisi-kondisi tertentu yang memerlukan pertimbangan dosen, pada sistem ini ditawarkan alternatif baru untuk penilaian dimana dosen diberi kebebasan (hak istimewa) untuk menentukan standart nilai menggunakan metode *clustering* dalam hal ini menggunakan *fuzzy clustering*. *Cluster* adalah sekelompok sesuatu yang mempunyai kesamaan sifat [1]. *Fuzzy clustering* memainkan peran yang paling penting dalam pencarian struktur dalam data [2].

Metode ini dipilih karena dengan metode ini, bisa ditentukan jumlah *cluster* yang akan dibentuk. Dengan penentuan jumlah *cluster* di awal, bisa diatur keragaman nilai akhir atau pelabelan nilai sesuai dengan *clusternya*.

Pada *fuzzy c-means* nilai awal titik pusat *cluster* dibangkitkan secara acak sehingga terjadi optimum lokal, dimana proses selanjutnya bergantung pada nilai awal yang dibangkitkan secara acak, disini akan digunakan algoritma genetika untuk mengoptimasi nilai awal titik pusat *cluster*.

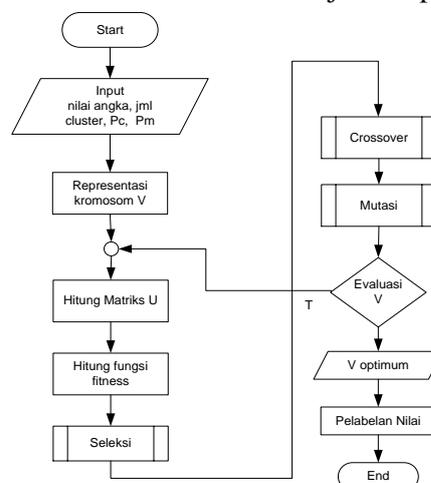
Algoritma genetika (GA) sebagai teknik optimasi dapat diterapkan pada *clustering* yang berbasis optimasi fungsi tujuan. Pada pendekatan GA untuk *fuzzy clustering* fungsi *fitness* diambil dari fungsi objektif yang diminimumkan, yaitu $J_m(U, V)$ [3].

Pada pendekatan algoritma genetika, Pada setiap generasi, kromosom dievaluasi berdasarkan nilai fungsi *fitness* [4], untuk pencarian titik pusat cluster dilakukan dengan cara meng-evolusikan matrik pusat *cluster* (seleksi, crossover dan mutasi) dengan fungsi *fitness* menggunakan fungsi objektif yang terdapat pada *fuzzy c-means*

2. METODE PENELITIAN

Secara umum sistem yang akan dibuat dalam penelitian ini adalah sistem untuk menentukan nilai akhir huruf mahasiswa dengan menggunakan perhitungan *Fuzzy clustering* dan algoritma genetika dimana algoritma genetika digunakan untuk optimasi titik pusat *cluster* pada *Fuzzy C-means*.

Dengan menggunakan algoritma genetika, nilai awal titik pusat *cluster* V dibangkitkan secara acak, kemudian nilai V tersebut digunakan untuk menghitung *matriks u*. Selanjutnya nilai V ini akan dievolusikan menggunakan seleksi, *crossover* dan mutasi untuk mendapatkan nilai V yang paling optimum. Deskripsi umum sistem tersebut ditunjukkan pada Gambar 1

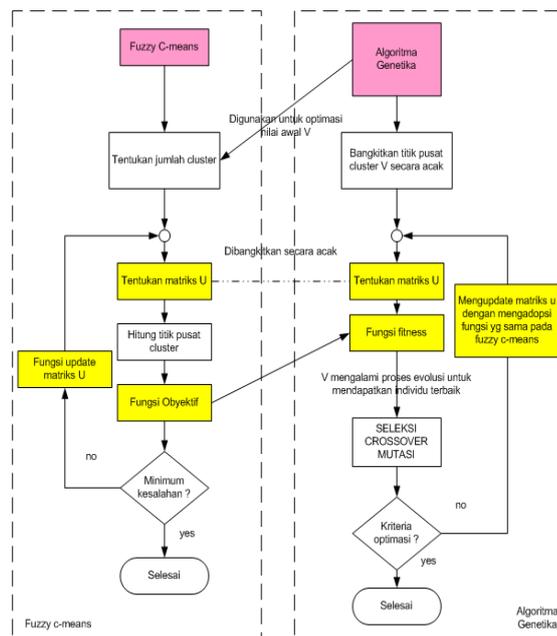


Gambar 1. Gambaran Umum Sistem

Langkah-langkah kegiatan yang akan dilakukan untuk permasalahan ini adalah:

1. Representasi solusi dan spesifikasi masukan dan keluaran.
2. Penentuan ukuran parameter genetika meliputi ukuran populasi, jumlah generasi, probabilitas *crossover*, probabilitas mutasi, maxIterasi.
3. Menentukan fungsi *fitness* yang digunakan, dalam hal ini fungsi *fitness* yang digunakan adalah fungsi objektif yang meminimumkan jarak antar data dengan titik pusat *cluster* (V) yang terdapat pada *Fuzzy C-means*.
4. Menentukan encoding kromosom yang digunakan yaitu *real encoding*.
5. Pemilihan fungsi seleksi yaitu menggunakan metode rank, *crossover* menggunakan *one cut point*, dan mutasi dengan mengubah gen secara acak.
6. Melakukan pengupdatetan *matrix* u yang merupakan derajat keanggotaan data pada masing-masing titik pusat *cluster* (V).

Langkah-langkah tersebut direpresentasikan pada Gambar 2.



Gambar 2. Penerapan Algoritma Genetika pada *Fuzzy C-means*

2.1 Perancangan Sistem

Pada penelitian ini digunakan metode GFS (*Genetic Fuzzy System*) untuk menentukan nilai akhir kuliah mahasiswa. Dari data nilai mahasiswa akan dikelompokkan menjadi beberapa *cluster*, dalam hal ini dosen berhak menentukan jumlah *cluster* yang diinginkan berdasarkan nilai terendah dan nilai tertinggi pada data nilai mahasiswa. Kemudian dari hasil *cluster* yang didapatkan, dosen dapat memberi pelabelan nilai yang sesuai

Dalam *clustering*, persoalan sebenarnya adalah persoalan untuk mencari titik pusat *cluster* yang paling optimum berdasarkan fungsi objektif. Pada penelitian ini, persoalan ini akan diselesaikan dengan menggunakan algoritma genetika. Dalam algoritma genetika, solusi dari persoalan direpresentasikan dengan kromosom.

2.1.1 Representasi Kromosom

Pada penelitian ini nilai awal kromosom akan dibangkitkan secara acak sebanyak n *cluster* (V) dan sebanyak jumlah populasi kromosom m . Jadi satu gen mewakili satu *cluster*. Banyaknya jumlah *cluster* ditentukan oleh dosen berdasarkan nilai minimal dan maksimal dari

data nilai mahasiswa, sedangkan jumlah awal kromosom yang digunakan pada penelitian ini adalah 50, hal ini dikarenakan algoritma genetika membutuhkan ruang solusi yang cukup besar dan ini berdasarkan rekomendasi dari De Jong 1975 yang terdapat pada dasar teori.

Tabel 1. Contoh Representasi Kromosom

	v1	v2	v3	v4	...	vn
Kromosom 1	1.4	5.0	7.1	1.5	...	0.4
Kromosom 2	0.5	4.4	6.0	1.3	...	2.6
...
Kromosom m	1.7	2.1	6.3	5.5	...	2.7

2.1.2 Hitung Matrik U

Matriks u adalah derajat keanggotaan data pada tiap-tiap *cluster*, Berdasarkan rumus perhitungan matriks u membutuhkan input berupa data nilai dan nilai titik pusat cluster. Rumus untuk menghitung matriks u ditunjukkan pada Rumus 1

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

2.1.3 Fungsi Fitness

Fungsi *fitness* merupakan ukuran kinerja suatu individu agar tetap bertahan hidup dalam lingkungannya. Dalam algoritma genetika fungsi *fitness* adalah fungsi objektif dari masalah yang akan dioptimasi.

Fungsi *fitness* yang digunakan dalam penelitian ini mengadopsi rumus fungsi objektif yang terdapat pada Fuzzy C-means yang meminimumkan jarak antara data dengan titik pusat cluster. Rumus fungsi objektif yang digunakan dalam fungsi *fitness* ditunjukkan pada Rumus 2

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

2.1.4 Seleksi

Pada penelitian ini, seleksi kromosom dilakukan dengan menggunakan metode *rank* yang bertujuan untuk mendapatkan prosentase yang lebih merata. Seleksi ini menggunakan *fitness* skala dalam memilih kromosom dari populasi. Untuk menghitung *fitness* skala dari suatu kromosom, maka perlu diketahui terlebih dahulu jumlah seluruh *fitness* dalam populasi. Prosedur seleksi adalah sebagai berikut:

1. Menghitung nilai *fitness* (f_k) tiap kromosom
2. Urutkan nilai *fitness* secara *descending* menggunakan library value comparator.
3. Beri prosentase pada tiap kromosom dengan membagi tiap kromosom dengan jumlah total kromosom dikalikan 100%
4. Hitung akumulasi prosentase
5. Pilih induk yang menjadi kandidat untuk di-*crossover* dengan cara:

- a. Bangkitkan suatu bilangan $[0,1]$
 - b. Memilih kromosom dengan nilai kumulatif q_k yang lebih besar dan yang paling mendekati dari nilai random sebagai parent.
6. Lakukan langkah 5 sebanyak ukuran populasi.

2.1.5 Crossover

Metode *crossover* yang digunakan pada penelitian ini adalah metode *one cut point* karena mudah diterapkan pada kromosom yang diencodingkan secara real. Tidak semua kromosom mengalami *crossover*, kromosom yang di *crossover* sesuai dengan probabilitas *crossover* (Pc). Jika terdapat dua kromosom yang nilai randomnya lebih kecil dari Pc, maka langsung di *crossover*, jika lebih dari dua kromosom maka *dicrossover* secara berurutan dan hanya dipilih satu *offspring* terbaik yang dapat menggantikan induk.

Prosedur persilangan satu titik sebagai berikut:

1. Menentukan Pc dan nilai acak p, dimana Pc adalah peluang *crossover* dan p adalah nilai yang dibangkitkan secara acak. Hanya p yang lebih kecil dari Pc yang akan di *crossover*. Nilai awal Pc yang digunakan pada penelitian ini adalah 0,6 yang merupakan rekomendasi dari De Jong yang mengacu pada dasar teori pada Bab III
2. Tentukan posisi *crossover* dengan cara membangkitkan bilangan acak dengan batasan 1 sampai n-1, dimana n adalah panjang kromosom. Misalkan didapat posisi *crossover* adalah 20 maka kromosom induk akan dipotong mulai gen ke 21 sampai gen ke n, kemudian potongan gen tersebut saling ditukar antar induk.
3. Posisi *cut-point crossover* dipilih menggunakan bilangan acak $1-(n-1)$ sebanyak jumlah *crossover* yang terjadi.

2.1.6 Mutasi

Metode mutasi dilakukan dengan cara uniform, yaitu dengan mengganti satu gen yang dipilih secara acak dengan suatu nilai baru dalam hal ini nilai baru tersebut juga ditentukan secara acak. Jumlah kromosom yang mengalami mutasi dalam satu populasi ditentukan oleh parameter peluang mutasi (p_m).

Langkah-langkah mutasi sebagai berikut:

1. Menentukan Pm dan p, dimana Pm adalah peluang mutasi dan p adalah nilai yang dibangkitkan secara acak. Nilai awal Pm yang digunakan pada penelitian ini adalah 0,001 yang merupakan rekomendasi dari De Jong yang mengacu pada dasar teori yang terdapat pada Bab III
2. Untuk memilih posisi gen yang akan mengalami mutasi dilakukan dengan cara membangkitkan bilangan acak (p) antara 0 sampai 1. Jika $p < p_m$, maka gen akan dimutasi, misal p_m kita temukan 10% maka diharapkan ada 10% dari populasi yang mengalami mutasi.
3. Bangkitkan bilangan acak untuk mengubah nilai gen yang ditentukan secara acak dengan nilai yang acak.

3. HASIL DAN PEMBAHASAN

Pembahasan pada sistem ini meliputi perbandingan *clustering* menggunakan FCM dan GFS dilihat dari nilai objektif, hasil pengelompokan data, waktu dan analisis pengaruh parameter genetika terhadap nilai objektif dan lama proses sistem. Pengaruh parameter yang dipelajari dalam penelitian ini dikelompokkan menjadi 4 bagian yaitu:

1. Hasil *clustering* menggunakan FCM
2. Pengaruh parameter ukuran populasi.
3. Pengaruh parameter jumlah iterasi
4. Pengaruh kombinasi parameter Pc dan Pm

3.1 Hasil Percobaan

Hasil percobaan pertama adalah hasil clustering dengan menggunakan parameter popsize, P_c , P_m rekomendasi dari De Jong, yaitu populasi = 50, $P_c = 0,6$, $P_m = 0,001$, maksimal iterasi = 100 dan jumlah cluster = 4 Data tersebut ditampilkan dalam Tabel 2 berikut ini.

Tabel 2 Data Nilai Mahasiswa Percobaan 1

Nim	Nama	Nilai	Cluster	Pusat Cluster	Nilai Huruf
1031140052	Aditya Dani Rosanto	90.0	4	94.11973	A
1031140065	Adhi Susanto	72.0	2	73.054245	B
1031140003	Arynda Putri Mertabuan	35.0	3	24.373365	D
1031140099	Avinda Ika Nadiasari	25.0	3	24.373365	D
1031140059	Bagus Rakhmadian	63.0	1	73.054245	C
1031140009	Deka Cahya Endradewi	83.0	2	73.054245	B
1031140056	Dian Munika Ayu	44.0	1	48.716026	C
1031140083	Diana Mayangsari	10.0	3	24.373365	D
1031140076	Elviera Savitrie P	53.0	1	48.716026	C
1031140024	Farah Azizah Ansi	95.0	4	94.11973	A
1031140092	Firman Adiyansyah	80.0	2	73.054245	B
1031140041	Ira Reza Permata	75.0	2	73.054245	B
1031140072	M. Hamzah Mashuri	65.0	1	73.054245	C
1031140066	Madarina Imanisari	95.0	4	94.11973	A
1031140097	Mokh Zafril	51.0	1	48.716026	C
1031140033	Nadia Putri	50.0	1	48.716026	C
1031140046	Putri Ardiani	25.0	3	24.373365	D
1031140017	Raj Nur	45.0	1	48.716026	C
1031140045	Revidya Dewantari	75.0	2	73.054245	B
1031140022	RR Eka Regina	100.0	4	94.11973	A

Dari hasil yang ditunjukkan pada Tabel 6.1, didapatkan fungsi objektif sebesar 800,557 dan membutuhkan waktu 869 ms

Percobaan selanjutnya sistem dijalankan dengan menggunakan data parameter genetika dengan jumlah *cluster* yang sama, tetapi parameter berbeda. Jumlah *cluster* = 4, ukuran populasi 50, probabilitas *crossover* (P_c) 0.3, probabilitas mutasi (P_m) 0.9, pembobot 2 dan max iterasi 300.

Tabel 3. Data Nilai Mahasiswa Percobaan 2

Nim	Nama	Nilai	Cluster	Pusat Cluster	Nilai Huruf
1031140052	Aditya Dani Rosanto	90.0	2	94.11973	A
1031140065	Adhi Susanto	72.0	3	73.054245	B
1031140003	Arynda Putri Mertabuan	35.0	4	24.373365	D
1031140099	Avinda Ika Nadiasari	25.0	4	24.373365	D
1031140059	Bagus Rakhmadian	63.0	3	73.054245	B
1031140009	Deka Cahya Endradewi	83.0	3	73.054245	B
1031140056	Dian Munika Ayu	44.0	1	48.716026	C
1031140083	Diana Mayangsari	10.0	4	24.373365	D
1031140076	Elviera Savitrie P	53.0	1	48.716026	C
1031140024	Farah Azizah Ansi	95.0	2	94.11973	A
1031140092	Firman Adiyansyah	80.0	3	73.054245	B
1031140041	Ira Reza Permata	75.0	3	73.054245	B
1031140072	M. Hamzah Mashuri	65.0	3	73.054245	B
1031140066	Madarina Imanisari	95.0	2	94.11973	A
1031140097	Mokh Zafril	51.0	1	48.716026	C
1031140033	Nadia Putri	50.0	1	48.716026	C
1031140046	Putri Ardiani	25.0	4	24.373365	D
1031140017	Raj Nur	45.0	1	48.716026	C
1031140045	Revidya Dewantari	75.0	3	73.054245	B
1031140022	RR Eka Regina	100.0	2	94.11973	A

Dari hasil percobaan tersebut terlihat sedikit perbedaan pengelompokan nilai. Fungsi objektif terakhir yang dihasilkan pada percobaan kedua adalah 662.817 dan membutuhkan waktu 2795 ms, nilai ini jauh lebih baik dibandingkan percobaan sebelumnya, data inilah yang digunakan sebagai acuan untuk membandingkan hasilnya dengan metode FCM.

3.2 Hasil Perbandingan dengan FCM

Dilakukan perbandingan menggunakan FCM dan GFS dengan data yang sama dan jumlah cluster yang sama. Metode GFS dijalankan dengan menggunakan data parameter genetika dengan jumlah *cluster* = 4, ukuran populasi 50, probabilitas *crossover* (Pc) 0.3, probabilitas mutasi (Pm) 0.9, pembobot 2 dan max iterasi 300. Data ini ditentukan secara acak. Fungsi objektif terakhir yang dihasilkan adalah 662.817 dan membutuhkan waktu 2795 ms.

Pengelompokan data nilai mahasiswa menggunakan FCM (*Fuzzy C-Means*) dilakukan dengan nilai parameter jumlah *cluster* = 4, pembobot = 2, maxIterasi = 100. Dari percobaan tersebut diketahui bahwa hasil *cluster* data nilai mahasiswa hasilnya sama baik menggunakan FCM maupun GFS, sedangkan dilihat dari waktu yang digunakan untuk mengklaster data, FCM relative lebih singkat dibanding GFS.

Fungsi objektif yang dihasilkan FCM lebih besar dari GFS yaitu 662.82, hal ini dikarenakan proses penyelesaian optimasi secara klasik sering menghantar pada penyelesaian yang merupakan optimum lokal sedangkan algoritma genetika akan selalu berevolusi untuk menemukan individu yang lebih baik tergantung dari parameter masukan.

3.3 Pengaruh Parameter Ukuran Populasi

Percobaan pertama adalah dengan mengambil jumlah *cluster* = 4, ukuran populasi = 10, Pc = 0.4, dan Pm = 0.8, jumlah iterasi = 100. Satu nilai parameter yang sama dihitung sebanyak 5 kali. Percobaan selanjutnya adalah dengan menambah ukuran populasi dan tetap menggunakan nilai parameter yang sama seperti pada percobaan pertama. Setelah dilakukan 10 kali percobaan dengan ukuran populasi yang berbeda diperoleh hasil seperti Tabel 2

Tabel 4 Pengaruh Parameter Ukuran Populasi

Ukuran Populasi	Nilai Objektif (B)	Waktu Proses (ms)	Ukuran Populasi	Nilai Objektif (B)	Waktu Proses (ms)
10	682,817	188	50	663,330	1058
	753,889	188		670,899	1035
	668,92	186		663,210	1162
	678,316	191		665,880	933
	674,107	190		664,535	1079
20	674,643	374	80	662,889	1822
	667,502	458		663,540	1740
	666,024	382		663,730	1701
	670,111	379		663,880	1620
	690,206	360		663,117	1729
30	669,843	566	100	663,09	2610
	663,751	650		663,089	2256
	668,497	575		663,477	2238
	666,702	398		662,997	2265
	667,801	450		663,137	2647
40	667,162	779	150	662,879	3406
	671,35	772		662,861	3505
	664,501	820		662,863	3762
	676,153	726		663,070	3065
	665,313	650		662,910	3889

Dari tabel diatas dapat dilihat bahwa semakin besar jumlah populasi maka nilai objektifnya semakin kecil dan stabil dan membutuhkan waktu yang relative lebih besar. Hal ini dikarenakan semakin banyak jumlah populasi kemungkinan ruang solusi juga semakin besar

dan peluang untuk memperoleh individu yang lebih baik semakin besar walaupun harus dibayar dengan waktu yang lebih banyak.

3.4 Pengaruh Parameter Jumlah Iterasi

Percobaan pertama adalah dengan mengambil jumlah iterasi awal = 100, ukuran populasi = 50, $P_c = 0.8$, dan $P_m = 0.2$. Percobaan selanjutnya adalah dengan menambah jumlah iterasi dan tetap menggunakan nilai parameter lain yang sama seperti pada percobaan pertama. Setelah dilakukan 8 kali percobaan diperoleh nilai objektif seperti Tabel 5

Tabel 5 Pengaruh Parameter Jumlah Iterasi

Jumlah Iterasi	Nilai Objektif (B)	Waktu Proses (ms)	Jumlah Iterasi	Nilai Objektif (B)	Waktu Proses (ms)
50	664,692	535	400	662,911	3264
	667,569	552		662,830	3113
	675,553	449		662,901	3963
	681,370	412		662,860	3888
	670,824	474		662,849	3246
100	667,04	1107	500	662,839	4934
	663,46	1111		662,852	4068
	664,091	1064		662,817	4065
	665,651	1079		662,85	4474
	663,920	1095		662,833	4808
200	663,213	1735	800	662,830	7331
	663,240	1943		662,827	6230
	663,490	1909		662,818	6505
	663,831	1928		662,817	7200
	663,197	1934		662,819	7730
300	662,970	2936	1000	662,830	7819
	662,890	2840		662,818	8941
	662,860	2808		662,820	7556
	662,88	2559		662,818	7467
	662,95	2728		662,816	7556

Dari Tabel 6.5 diketahui bahwa nilai objektif terbaik (yang lebih kecil daripada FCM) diperoleh mulai dari jumlah iterasi 500 dan seterusnya, dimana nilai objektif yang diperoleh yaitu 662,817. Perbedaan nilai yang terjadi pada iterasi yang sama dikarenakan setiap kali perhitungan yang berbeda, proses seleksi, *crossover*, dan mutasi yang terjadi juga berbeda tergantung bilangan acak yang didapatkan. Hasil ini bisa lebih baik maupun lebih buruk.

3.5 Pengaruh Kombinasi P_c dan P_m

Percobaan pertama adalah dengan mengambil nilai $P_c = 0.6$, dan $P_m = 0,001$ ukuran populasi = 50, jumlah iterasi = 500. Percobaan selanjutnya adalah dengan menambah nilai P_m dan tetap menggunakan nilai parameter lain yang sama seperti pada percobaan pertama kemudian dilanjutkan mengubah $P_c = 0,1$ dan seterusnya. Satu parameter dihitung sebanyak 3 kali

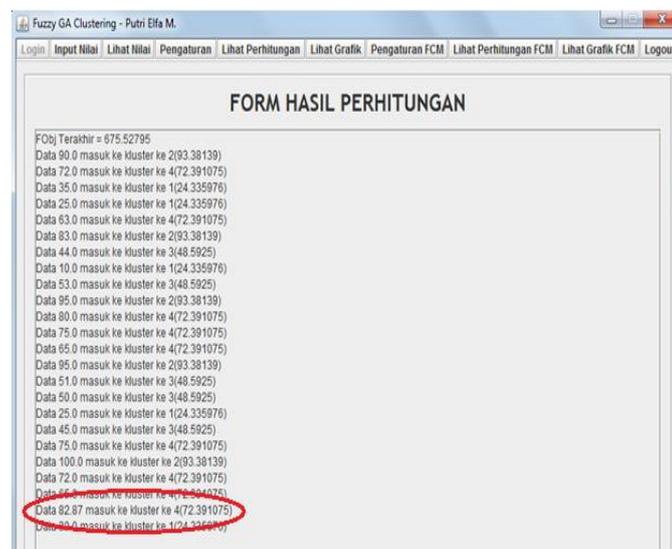
Hasil pengujian menunjukkan bahwa perubahan nilai P_m sangat berpengaruh pada hasil fungsi objektif dimana nilai P_m yang dapat menghasilkan fungsi objektif terbaik adalah antara 0,7- 0,9. Nilai parameter P_m yang cenderung besar ini menyebabkan jumlah kromosom baru hasil mutasi lebih banyak untuk generasi berikutnya sehingga nilai objektif terbaik akan mudah dicapai.

Untuk perubahan nilai P_c tidak terlalu berpengaruh secara signifikan pada hasil fungsi objektif, namun ketika kombinasi $P_c = 0,3$ dan $P_m = 0,9$ setelah 10 kali perhitungan semua hasil proses menghasilkan nilai objektif yang lebih rendah dari FCM. Hal ini dikarenakan parameter $P_c=0,3$ tidak terlalu kecil tapi juga tidak besar.

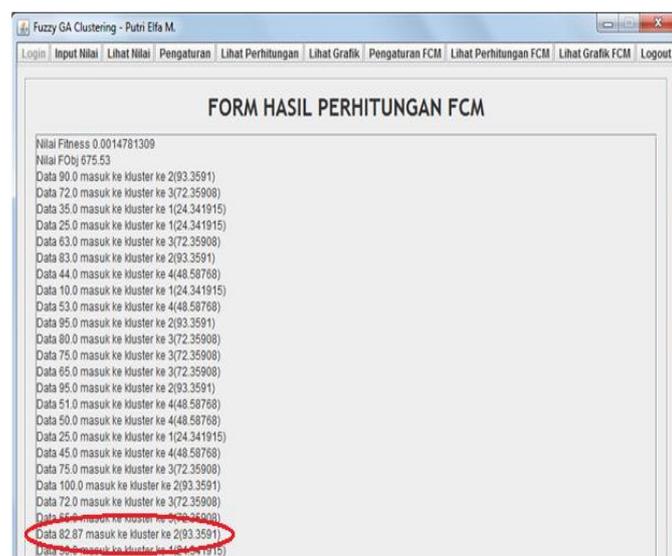
3.6 Perbandingan FCM dan GFS dengan Data Terperinci

Percobaan selanjutnya dilakukan dengan mengubah-ubah data yang lebih presisi untuk menguji apakah fungsi objektif yang berbeda dengan selisih yang sangat kecil berpengaruh terhadap anggota *cluster*. Hal ini terbukti pada hasil yang diperoleh pada data 82,87 dimana fungsi objektif yang dihasilkan dari proses GA dan GFS berbeda dengan selisih 0,01. Fungsi objektif hasil proses FCM adalah 675,53 dan fungsi objektif hasil proses GFS adalah 675,527. Hasil percobaan diatas menunjukkan bahwa dengan selisih fungsi objektif yang kecil yaitu 0,01 dapat berpengaruh terhadap anggota *cluster* jika datanya adalah data yang lebih presisi (dengan beberapa angka dibelakang koma). Hal ini dikarenakan walaupun fungsi objektif hanya selisih 0,01 tapi hal tersebut berpengaruh terhadap titik pusat *cluster*. Data yang lebih terinci, akan membutuhkan ketelitian lebih untuk memilih anggota pada *cluster*.

Pada hasil proses FCM_GA data nilai 82,87 masuk ke *cluster* 4 dengan titik pusat *cluster* adalah 72,391 sedangkan hasil proses FCM, data nilai 82,87 masuk ke *cluster* 2 dengan titik pusat *cluster* adalah 93,359.



Gambar 3. Hasil Perhitungan GFS



Gambar 4. Hasil Perhitungan FCM

4. KESIMPULAN

1. Algoritma genetika (GFS) dapat digunakan untuk mengelompokkan data nilai mahasiswa dengan nilai fungsi objektif yang lebih kecil daripada menggunakan FCM.
2. Meskipun selisih nilai objektif hasil proses GFS dan FCM tidak terlalu besar tetapi hal tersebut berpengaruh terhadap keanggotaan pada tiap-tiap *cluster* karena data yang lebih terperinci (beberapa dibelakang koma) membutuhkan ketelitian tinggi untuk memilih anggota *cluster*.
3. Setiap dilakukan beberapa kali perhitungan pada parameter genetika yang sama, hasilnya bisa berubah-ubah karena setiap kali perhitungan, proses seleksi, crossover dan mutasi yang terjadi juga berbeda. Namun pada kombinasi Pc 0,3 dan Pm 0,9 perubahan yang terjadi sangat kecil dan relative stabil.
4. Jika ditinjau dari waktu komputasi, metode GFS masih relative membutuhkan waktu yang lebih lama dibanding menggunakan FCM karena untuk menemukan individu terbaik menggunakan proses evolusi (seleksi, crossover, mutasi) yang membutuhkan waktu lebih besar daripada FCM yang hanya meminimumkan jarak dengan menggunakan iterasi. Namun dengan perkembangan prosesor yang semakin baik kemungkinan hal tersebut dapat diatasi.
5. Semakin besar ukuran populasi dan jumlah iterasi maka hasil yang didapatkan akan semakin optimal dan stabil, stabil dalam artian nilai objektif relatif sama meskipun dilakukan beberapa kali perhitungan. karena semakin banyak jumlah generasi maka memberikan eksplorasi terhadap ruang pencarian yang lebih besar walaupun harus dibayar dengan waktu yang lebih lama.

5. SARAN

Berdasarkan pada pengujian yang telah dilakukan pada sistem yang dibuat, masih banyak kekurangan dan kelemahan sehingga perlu dikembangkan lagi agar kinerjanya lebih baik, oleh karena itu saran yang diberikan adalah sebagai berikut:

1. Metode GFS ini lebih baik digunakan untuk studi kasus yang membutuhkan data yang lebih presisi, misalnya pengukuran yang membutuhkan tingkat ketelitian yang tinggi.
2. Pada penelitian lebih lanjut diharapkan melakukan riset algoritma untuk mengoptimalkan nilai parameter Pc dan Pm sehingga didapatkan nilai Pc dan Pm yang terbaik yang dapat digunakan pada berbagai jenis data yang berbeda.

DAFTAR PUSTAKA

- [1] Cox, E. 2005. *Fuzzy Modeling and Genetic Algorithm for Data Mining and Exploration*, Morgan Kaufmann Publisher, San Francisco.
- [2] Klir G. J., Yuan B., 1995. *Fuzzy Sets and Fuzzy Logic, Theory and Application*. Prentice Hall International, Inc.
- [3] Widyastuti N., Hamzah A., 2007, Penggunaan Algoritma Genetika Dalam Peningkatan Kinerja *Fuzzy Clustering* untuk Pengenalan Pola, *Seminar Penggunaan Algoritma Genetika*, Yogyakarta
- [4] Gen, M. and Cheng, R., 2000, *Genetic Algorithms and Engineering Optimization*, John Wiley & Sons, New York.