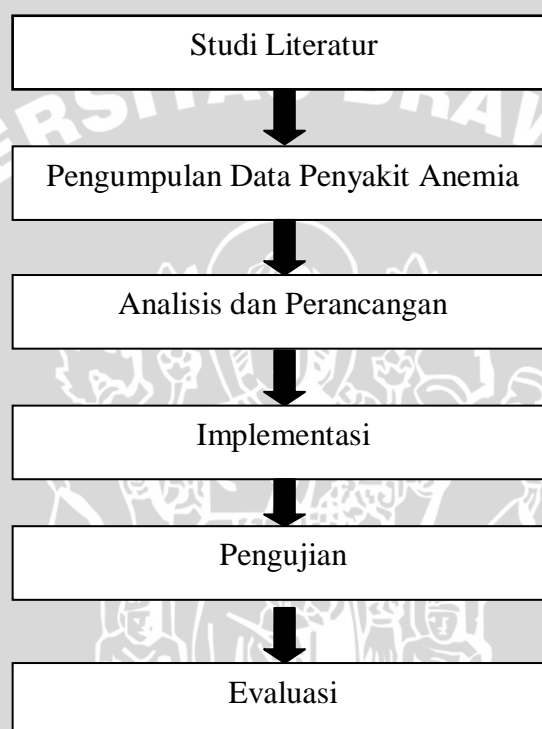


BAB III

METODOLOGI DAN PERANCANGAN

Pada bab metodologi dan perancangan ini akan dibahas mengenai langkah-langkah pengerjaan dalam penelitian yang berjudul “Implementasi algoritma fuzzy c-means untuk pengelompokan tingkat penyakit anemia”. Langkah-langkah yang dilakukan dalam pengerjaan penelitian ini adalah



Gambar 3.1 Diagram Alir Penelitian

Adapun penjelasan langkah-langkah penelitian pada Gambar (3.1) sebagai berikut :

1. Mempelajari literatur yang berkaitan dengan Fuzzy C-Means dan parameter yang mempengaruhi tingkat penyakit anemia.
2. Melakukan pengumpulan data set penderita penyakit anemia untuk pengujian.
3. Menganalisa dan melakukan perancangan sistem menggunakan algoritma Fuzzy C-Means untuk pengelompokan tingkat penyakit anemia.
4. Tahap dari proses implementasi program merupakan bagian dari pembuatan implementasi program.

5. Melakukan uji coba terhadap program yang dibangun, untuk memastikan bahwa implementasi program yang dibangun telah berjalan sesuai dengan yang diharapkan.
6. Melakukan Evaluasi output yang dihasilkan oleh uji coba implementasi program.

3.1 Studi Literatur

Dalam pengerjaan penelitian ini dibutuhkan studi literatur untuk mempelajari dasar teori terkait dengan penelitian yang akan dilakukan. Teori-teori tersebut dapat diperoleh dari berbagai sumber seperti buku, jurnal, *e-book*, penelitian sebelumnya, internet, dan sumber pustaka lain yang dapat dipertanggungjawabkan. Teori – teori yang dipelajari terkait dengan penelitian ini meliputi penyakit anemia dan *Fuzzy C-Means* (FCM).

3.2 Data Penelitian

Dataset yang digunakan dalam penelitian ini adalah data penyakit anemia pada hasil test darah yang diambil dari laboratorium klinik sejahtera Kota Probolinggo Tahun 2013. Jumlah data yang digunakan dalam penelitian ini adalah 300 jumlah data dan 6 atribut yaitu Jenis kelamin, Umur, Hemoglobin, Lekosit, Trombosit dan Eritrosit. Dari 6 atribut tersebut didapatkan hasil tingkat penyakit anemia.

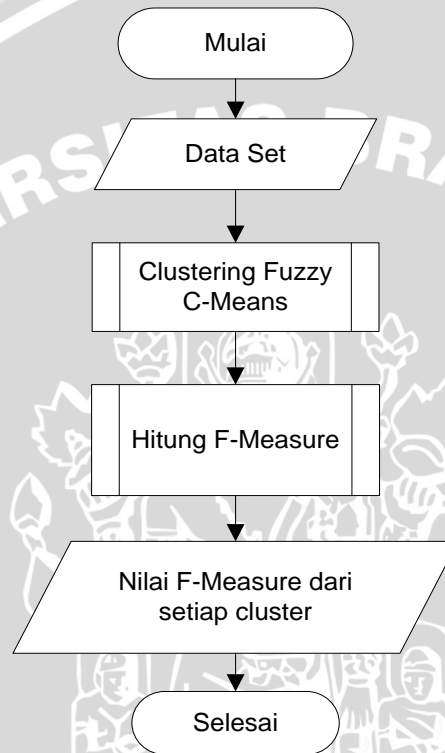
3.3 Deskripsi Umum

Merupakan implementasi program yang dapat mengelompokkan data menggunakan algoritma *Fuzzy C-Means*. Proses *cluster* data dilakukan dengan menentukan titik pusat suatu data serta menghitung jarak antar data berdasarkan kesamaan antar data, selain itu juga ditentukan seberapa besar nilai keanggotaan suatu data menjadi milik *cluster* tertentu. Adapun proses yang dilakukan pada saat implementasi program antara lain :

1. User memasukkan data.
2. User memasukkan jumlah *cluster*, pangkat atau pembobot *fuzzy*, *error* terkecil dan maksimum iterasi.
3. Program melihat hasil *clustering*.

3.4 Perancangan

Perancangan dilakukan untuk menjelaskan bagaimana proses yang dikerjakan dalam pengelompokan data tingkat penyakit anemia. Data yang digunakan berupa format Microsoft excel. Pada perancangan sistem ini terdiri dari 2 prosedur yaitu *clustering Fuzzy C-Means* dan hitung *F-Measure*. Rancangan alur pengelompokan secara umum dapat dilihat pada gambar (3.2).



Gambar 3.2 Alur perancangan

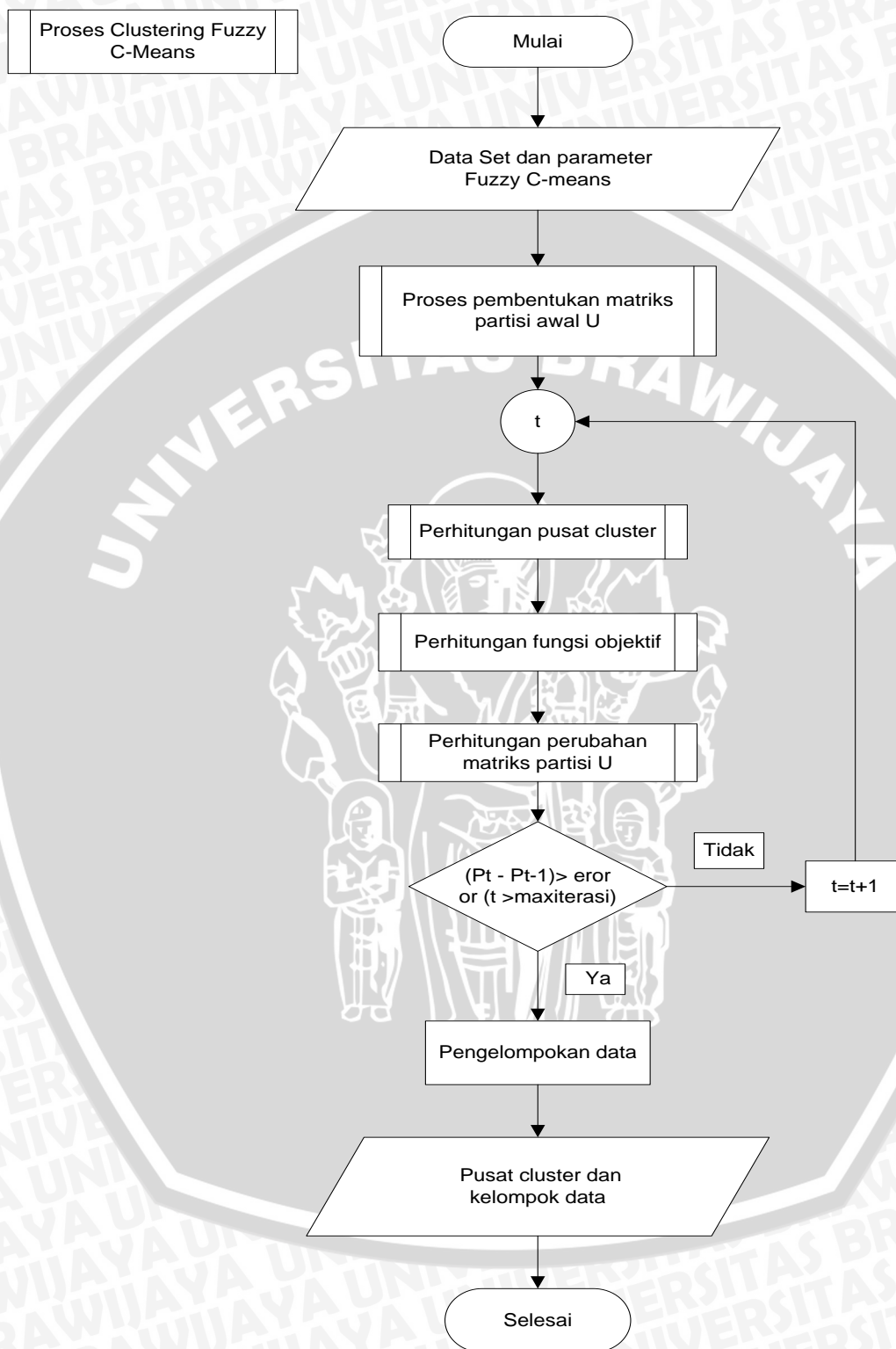
3.4.1 Proses Clustering *Fuzzy C-Means*

Proses *clustering* data terdiri atas beberapa *subproses*, diantaranya adalah pembentukan matriks partisi awal U , perhitungan pusat *cluster*, perhitungan fungsi objektif dan perhitungan perubahan matriks U . Input dari proses *clustering* berupa data tingkat penyakit anemia dan parameter *clustering*, sedangkan *output* proses ini berupa matriks pusat *cluster*, matriks partisi U dan kelompok data. Matriks pusat *cluster* berisi pusat data atribut pada setiap *cluster* (V_{kj}), matriks partisi U berisi derajat keanggotaan data pada semua *cluster* (μ_{ik}) dan kelompok data berisi hasil pengelompokan data berdasarkan derajat keanggotaan data

terhadap suatu *cluster*. Pengelompokan data dilakukan dengan memperhatikan nilai maksimum μ_{ik} data pada setiap *cluster*. Pada gambar (3.3) dijelaskan alur dari keseluruhan proses pengelompokan *Fuzzy C-Means*. Alur dari proses *clustering* data dengan menggunakan algoritma *Fuzzy C-Means* adalah sebagai berikut :

1. Menentukan *error* minimum (ξ), maksimum Iterasi ($maxIter$) dan nilai pangkat *cluster* (w).
2. Membentuk matriks partisi awal U .
3. Pembentukan matriks partisi awal dilakukan dengan cara membangkitkan bilangan random yang dihitung dengan persamaan (2.1) dan (2.2).
4. Selama memenuhi kondisi dimana nilai fungsi objektif lebih besar sama dengan *error* minimum yang diharapkan ($P_t \geq \xi$) atau iterasi lebih kecil sama dengan iterasi maksimum ($t \leq maxIter$), maka dilakukan proses berikut secara berulang :
 - a. Menghitung pusat *cluster* (V_{kj}) dengan menggunakan persamaan (2.3).
 - b. Menghitung nilai dari fungsi objektif (P_t) dengan menggunakan persamaan (2.4).
 - c. Menghitung perubahan matriks partisi U dengan menggunakan persamaan (2.5).
5. Pada tahap akhir dilakukan pengecekan kondisi berhenti. Alternatif kriteria penghentian adalah jika perubahan nilai *error* kurang dari *threshold* $|P_n - P_{n-1}| < \epsilon$ atau iterasi lebih besar dari iterasi maksimum ($t > maxIter$). Jika iterasi belum berhenti, kembali ke langkah 4.
6. Jika iterasi berhenti, ditentukan *cluster* dari tiap-tiap data. *Cluster* dipilih berdasarkan nilai matriks partisi terbesar.

Alur dari keseluruhan proses *clustering Fuzzy C-Means* ditunjukkan pada gambar (3.3).



Gambar 3.3 Flowchart Fuzzy C-Means Clustering

3.4.1.1 Proses Pembentukan Matriks Partisi Awal U

Input yang diperlukan dalam proses pembentukan matriks partisi awal adalah nilai jumlah data (n) dan jumlah *cluster* yang akan dibentuk (c). Nilai *cluster* yang akan dibentuk merupakan suatu nilai *inputan*, sehingga nantinya banyaknya *cluster* yang terbentuk sesuai nilai yang telah ditentukan oleh pengguna. *Output* dari proses ini adalah matriks derajat keanggotaan data terhadap *cluster* sesuai dengan banyaknya dimensi data.

Rincian alur proses pembentukan matriks partisi awal U adalah sebagai berikut:

1. Pembangkitkan bilangan random

Bilangan random yang dibangkitkan berguna untuk merepresentasikan derajat keanggotaan suatu data ke- i ke dalam *cluster* ke- k (μ_{ik}). Rentang bilangan random yang ditentukan yaitu bilangan antara 0 sampai 1.

2. Penjumlahan elemen setiap kolom

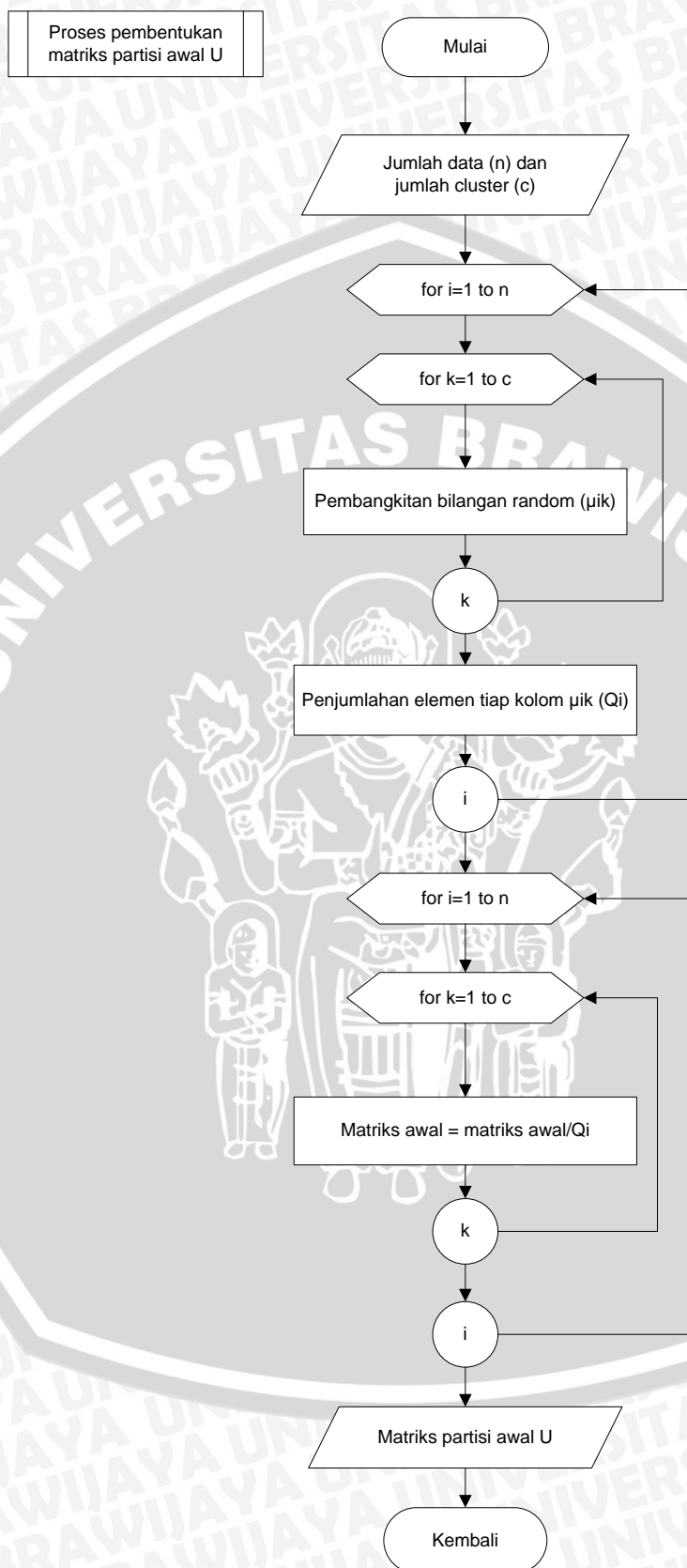
Proses ini menjumlah bilangan random μ_{ik} setiap iterasi ke- i ($Q_i = \mu_{i1} + \mu_{i2} + \mu_{i3} + \dots + \mu_{ik}$) berdasarkan Persamaan (2.1) sehingga didapatkan Q_i satu dimensi berukuran n .

3. Perhitungan nilai elemen matriks

Setelah perhitungan nilai Q_i , dilakukan perhitungan dengan Persamaan (2.2) untuk matriks partisi awal U sehingga menghasilkan derajat keanggotaan (μ_{ik}). Jumlah μ_{ik} pada setiap iterasi ke- i harus sama dengan 1 ($\mu_{i1} + \mu_{i2} + \mu_{i3} + \dots + \mu_{ik} \neq 1$) apabila tidak bernilai 1 maka terjadi kesalahan selama perhitungan derajat keanggotaan μ_{ik} .

Alur proses pembentukan matriks partisi awal U ditunjukkan oleh Gambar

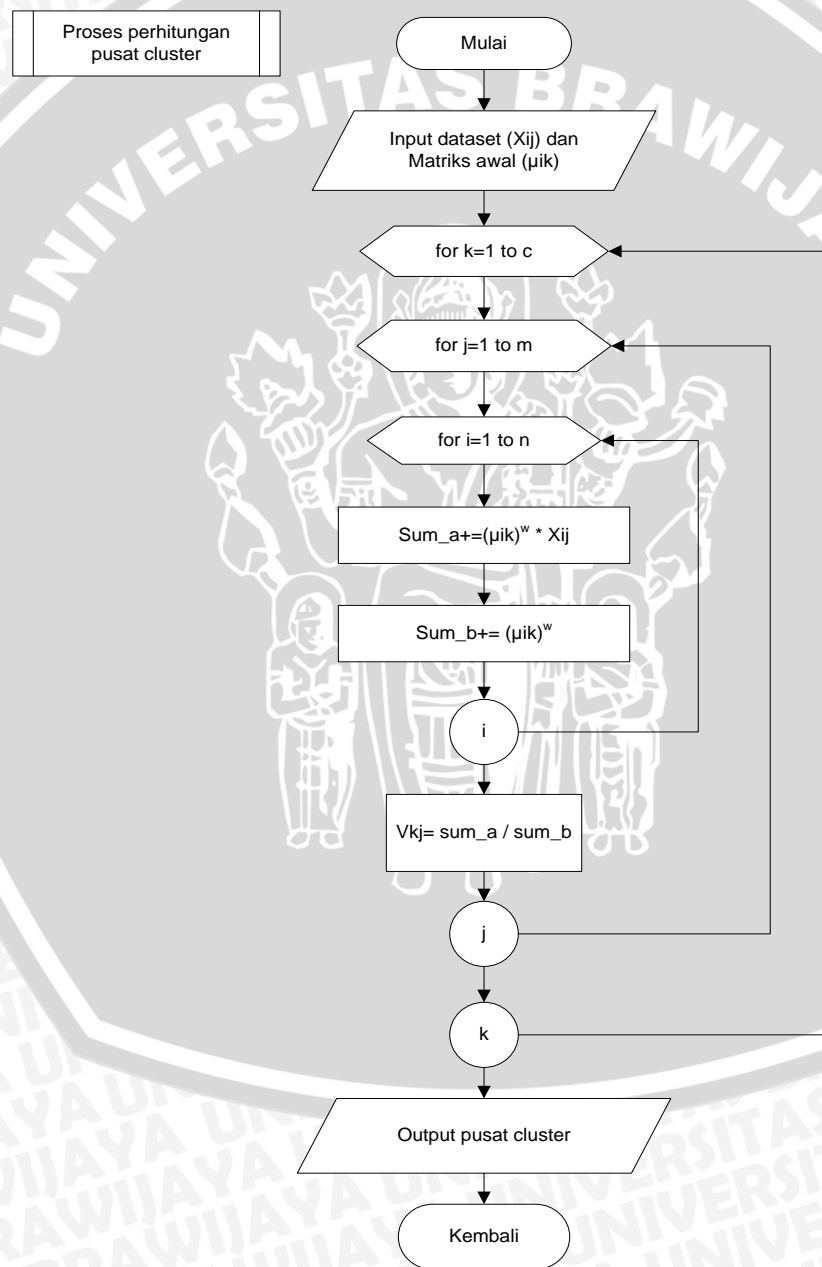
(3.4).



Gambar 3.4 Flowchart Pembentukan Matriks Awal U

3.4.1.2 Proses Perhitungan Pusat Cluster

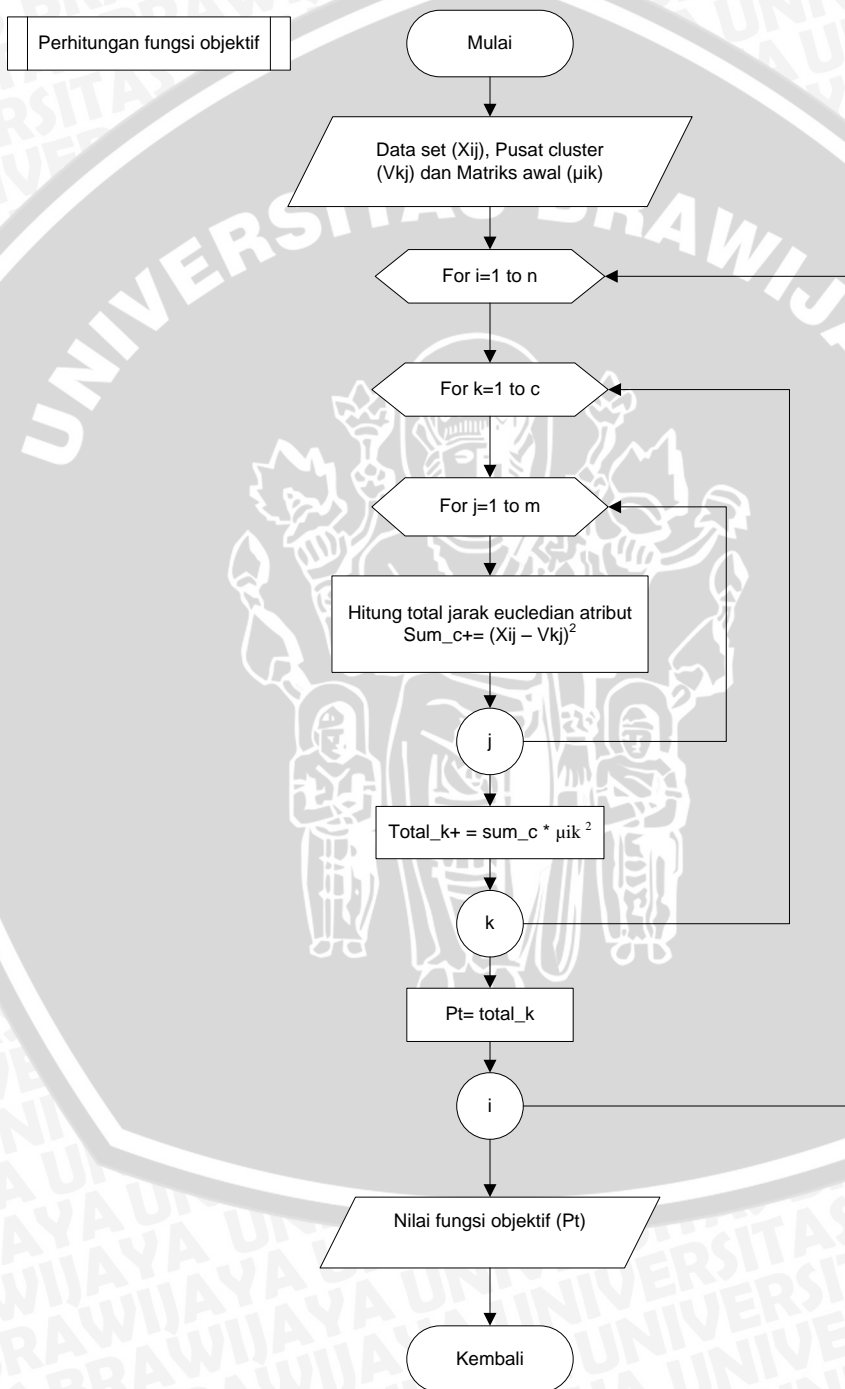
Pada proses perhitungan pusat *cluster* memerlukan *input* berupa dataset penyakit anemia (X_{ij}) dan derajat keanggotaan hasil dari proses pembentukan matriks partisi awal U (μ_{ik}). Perhitungan pusat *cluster* dilakukan menurut persamaan (2.3) Perhitungan ini menghasilkan matriks pusat *cluster* (V_{kj}) berukuran $c \times m$ (jumlah *cluster* x jumlah atribut). Alur proses perhitungan pusat cluster ditunjukkan pada gambar (3.5).



Gambar 3.5 Flowchart Perhitungan Pusat Cluster

3.4.1.3 Proses Perhitungan Fungsi Objektif

Proses perhitungan fungsi objektif menggunakan *inputan* data latih (X_{ij}), pusat *cluster* (V_{kj}) dan matriks awal (μ_{ik}). Perhitungan fungsi objektif dilakukan dengan menggunakan persamaan (2-4). Alur proses perhitungan fungsi objektif ditunjukkan pada Gambar (3.6).



Gambar 3.6 Flowchart Perhitungan Fungsi Objektif

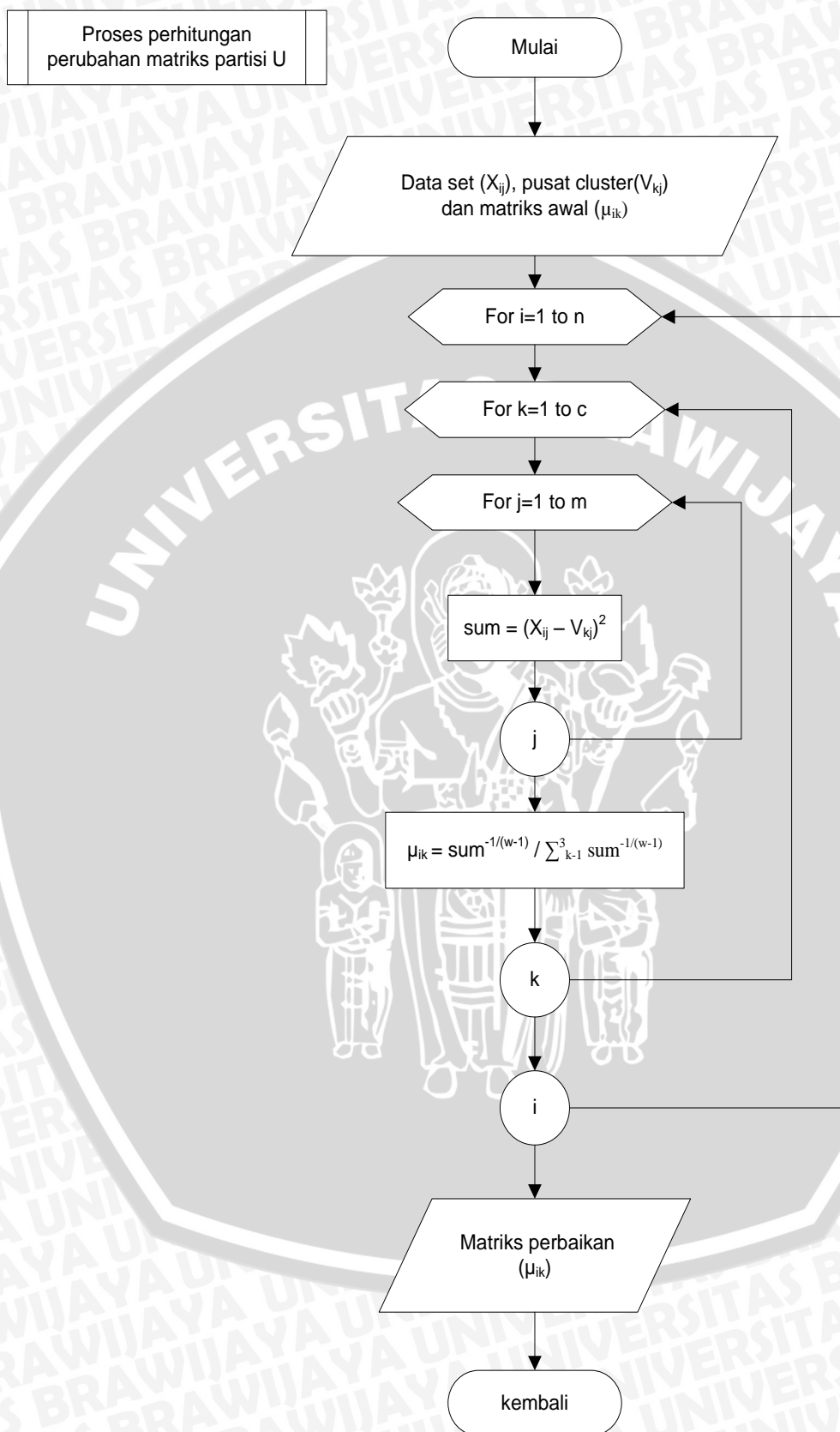
Rincian alur proses perhitungan fungsi objektif adalah sebagai berikut:

- Pada setiap iterasi ke- i dilakukan iterasi k sebanyak c (jumlah *cluster* yang akan dibentuk) untuk melakukan perhitungan total jarak atribut (A_{ik}). Total jarak (A_{ik}) didapatkan dari penjumlahan jarak semua atribut ($A_{ik} = (X_{i1} - V_{k1})^2 + (X_{i2} - V_{k2})^2 + (X_{i3} - V_{k3})^2 + \dots + (X_{ij} - V_{kj})^2$)
- Nilai A_{ik} dikalikan dengan derajat keanggotaan (μ_{ik}) kemudian ditotal untuk setiap iterasi i . Pada akhirnya, semua total perhitungan akan dijumlah sehingga menghasilkan satu nilai fungsi objektif (P_t).
- Selisih Nilai P_t dengan nilai P_{t-1} dihitung kemudian dibandingkan dengan nilai kesalahan minimum (ξ) yang telah ditentukan untuk pemeriksaan kondisi berhenti pada proses *clustering* data. Jika $|P_t - (P_{t-1})| < \xi$ atau $t > \maxIter$ maka perulangan pada proses *clustering* dihentikan.

3.4.1.4 Proses Perubahan Matriks Partisi U

Seperti pada perhitungan fungsi objektif, proses perubahan matriks partisi U dilakukan untuk memperbaiki nilai derajat keanggotaan (μ_{ik}) berdasarkan nilai pusat *cluster*. Seperti pada perhitungan fungsi objektif, proses perubahan matriks partisi U memiliki *input* berupa data latih mamografi (X_{ij}), pusat *cluster* (V_{kj}), dan matriks awal (μ_{ik}). *Output* proses ini adalah perbaikan derajat keanggotaan μ_{ik} yang merupakan hasil perubahan matriks partisi U. Perubahan matriks partisi U dapat dihitung dengan menggunakan Persamaan (2.5).

Sama seperti perhitungan nilai elemen matriks pada matriks partisi awal, jumlah μ_{ik} pada setiap iterasi ke- i adalah 1 ($\mu_{i1} + \mu_{i2} + \mu_{i3} + \dots + \mu_{ik} = 1$). Apabila jumlah μ_{ik} pada setiap iterasi ke- i tidak sama dengan 1 ($\mu_{i1} + \mu_{i2} + \mu_{i3} + \dots + \mu_{ik} \neq 1$), maka terjadi kesalahan selama proses *clustering* data. Alur proses perubahan matriks partisi U ditunjukkan oleh Gambar (3.7).



Gambar 3.7 Flowchart Perubahan Matriks Partisi U

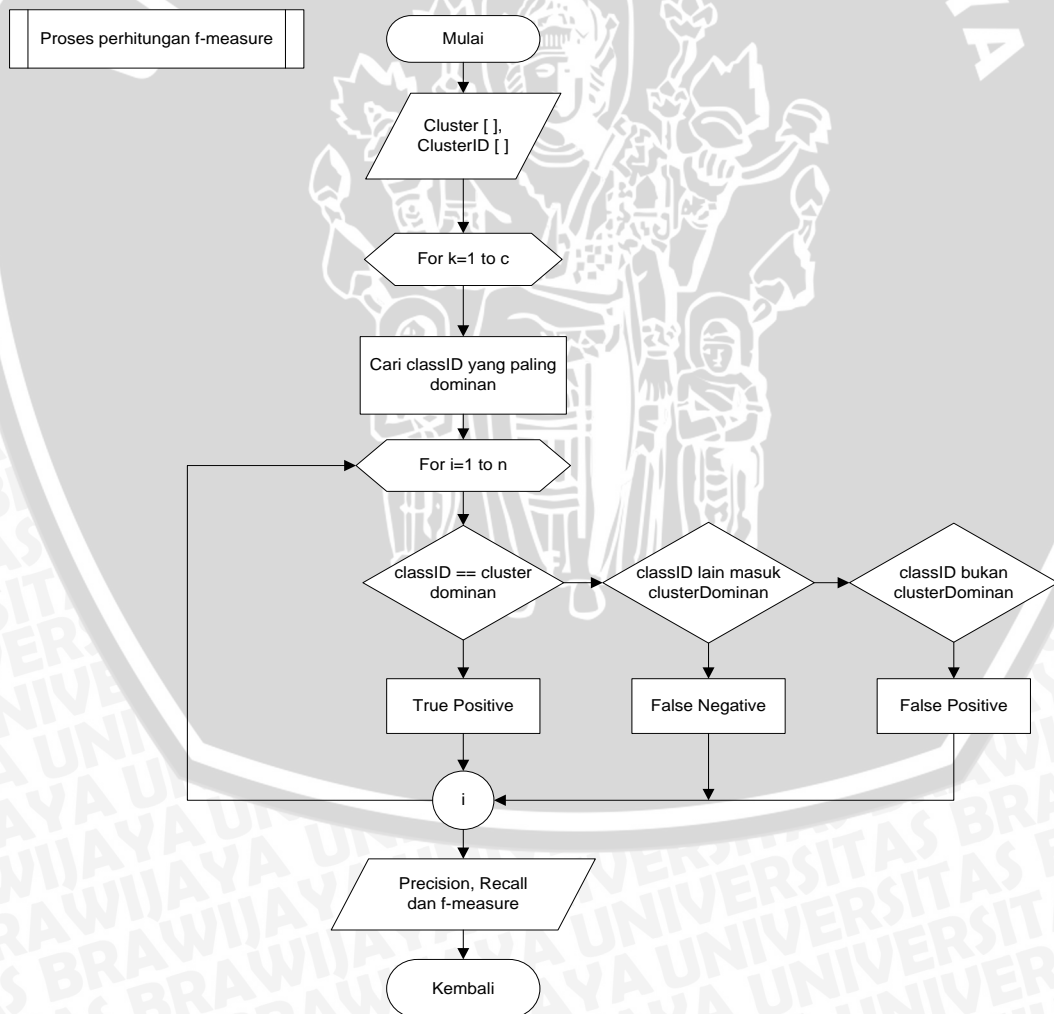
3.4.2 Pengujian F-Measure

Untuk menguji metode *Fuzzy C-Means* digunakan metode *f-Measure*.

Langkah-langkah metode *f-Measure* adalah sebagai berikut :

1. Ambil Cluster[], classID[], classID adalah nilai kelas sebenarnya.
2. Cari classID paling dominan pada suatu *cluster*.
3. Jika :
 - 3.1 ClassID = ClusterDominan → true positive
 - 3.2 ClassID cluster lain masuk ClusterDominan → false negative
 - 3.3 ClassID bukan ClusterDominan → false positive
4. Lakukan perhitungan *precision, recall, f-Measure*.
5. Ulangi langkah 4 untuk semua *cluster*.

Alur proses *f-Measure* dapat dilihat pada gambar (3.8).



Gambar 3.8 Flowchart Perhitungan f-Measure

3.5 Perhitungan Manual

Perhitungan manual dilakukan untuk mengimplementasikan sistem secara matematis lewat perhitungan langkah demi langkah pada data set. Proses perhitungan manual *Fuzzy C-Means Clustering* sebagai berikut :

1. Proses *Fuzzy C-Means Clustering*

Langkah pertama pada proses *fuzzy c-means clustering* adalah menyiapkan data set yang akan diklaster dan parameter *fuzzy c-means*. Jumlah atribut yang digunakan sebanyak enam atribut yang terdiri dari Jenis Kelamin, Umur, Hemoglobin, Lekosit, Trombosit, dan eritrosit. Data Penyakit Anemia (X) yang digunakan untuk data set pada proses *clustering* disajikan pada Tabel (3.1).

Tabel 3.1 Sampel Data Set

Data (i)	Atribut (j)						Class
	Jenis Kelamin	Umur	Hemoglobin	Lekosit	Trombosit	Eritrosit	
1	1	59	13,7	4500	315000	5,2	Normal
2	1	79	13,2	11000	317000	5,8	Normal
3	1	48	13,7	11000	317000	5,2	Normal
4	1	59	14,2	8000	315000	5,5	Normal
5	2	49	14,6	5000	260000	4	Normal
6	2	26	13,1	11000	315000	4	Normal
7	2	50	12,5	5600	245000	5,3	Ringan
8	2	41	11,8	9000	280000	5,1	Ringan
9	2	91	11,9	14800	224000	5	Ringan
10	1	52	10,9	8400	415000	5,1	Ringan
11	1	56	10	6600	216000	5,1	Ringan
12	1	46	10,8	4800	158000	5,3	Ringan
13	2	58	12,8	9000	248000	5	Ringan
14	2	62	7,1	14200	488000	4,9	Berat
15	1	57	9,8	10400	393000	4,9	Berat
16	2	50	9,4	6000	256000	5,2	Berat
17	2	44	5,3	13200	212000	4,3	Berat
18	2	54	8	16000	200000	4,5	Berat
19	1	23	6,6	6200	280000	4,8	Berat
20	2	80	11	8300	300000	4,4	Berat

Berdasarkan tabel data set, terdapat data sejumlah 20 ($n=20$) dengan atribut sejumlah 6 ($m=6$). Pada contoh perhitungan manual ini data akan dikelompokkan ke dalam 3 kelompok dengan menggunakan FCM *clustering*.

Nilai awal parameter pada algoritma FCM dapat diinisialisasi sebagai berikut:

1. Menentukan Atribut

Banyak <i>cluster</i> / c	3
Pembobot / w	2
Max iterasi / i	300
<i>Error</i> terkecil yang diharapkan	0.00001
Fungsi Objektif awal / P_0	0
Iterasi Awal / t	1

2. Membangkitkan bilangan random μ_{ik} dalam matriks berukuran ($n \times m$).

Setelah menentukan atribut, langkah selanjutnya adalah pembentukan matriks partisi awal U . Langkah awal pembentukan matriks partisi awal U yaitu membangkitkan bilangan random μ_{ik} dalam matriks berukuran jumlah data \times jumlah cluster (20×3). Pembangkitan bilangan random dilakukan dengan fungsi RAND() pada Microsoft Excel. Hasil pembangkitan bilangan random μ_{i1} , μ_{i2} dan μ_{i3} ditunjukkan pada tabel (3.2).

Setelah proses pembangkitan bilangan random langkah selanjutnya dilakukan penjumlahan kolom setiap baris bilangan random yang terbentuk untuk masing-masing elemen I (Q_i) berdasarkan persamaan (2.1).

Tabel 3.2 Bilangan Random dan Hasil Penjumlahan (Q_i)

Matriks Bilangan Random			Hasil Penjumlahan
U_{i1}	U_{i2}	U_{i3}	Q_i
0,670568651	0,136947374	0,587714353	1,395230378
0,877724787	0,062086654	0,768513199	1,708324641
0,152396045	0,352121689	0,353928482	0,858446216
0,902627771	0,330443597	0,318733147	1,551804515
0,190690599	0,760466817	0,953871766	1,905029183
0,899809188	0,814130238	0,162344047	1,876283474
0,752771054	0,971996335	0,003835239	1,728602628

0,995490233	0,033088723	0,456309601	1,484888557
0,15324577	0,964758975	0,078448408	1,196453154
0,868092086	0,560372768	0,852627036	2,28109189
0,691532372	0,576323543	0,486534166	1,754390081
0,014742302	0,636861823	0,090017166	0,741621291
0,279951357	0,061754037	0,156348856	0,498054249
0,086001959	0,120813948	0,47317792	0,679993826
0,023698677	0,508337863	0,116690575	0,648727115
0,829944995	0,625972956	0,073165626	1,529083576
0,078055702	0,848955886	0,754578733	1,68159032
0,372773793	0,341447608	0,706489018	1,420710419
0,740418183	0,424538446	0,178244857	1,343201485
0,209201576	0,428219648	0,778510334	1,415931557

Setiap bilangan random yang telah terbentuk kemudian dibagi dengan Q_i berdasarkan persamaan (2.2). Contoh perhitungan nilai matriks partisi awal U untuk μ_{i1} :

$$\mu_{i1} = \frac{\mu_i 1}{Q_i} = \frac{0,670568651}{1,395230378} = 0,480615002$$

Hasil matriks partisi awal U :

0,480615002	0,098153951	0,421231047
0,513792734	0,036343592	0,449863674
0,177525443	0,410184916	0,412289641
0,581663323	0,21294151	0,205395167
0,100098519	0,399189064	0,500712417
0,479569959	0,433905777	0,086524264
0,435479526	0,56230178	0,002218693
0,670414105	0,022283641	0,307302254
0,128083385	0,806349143	0,065567472
0,380559893	0,245659884	0,373780223
0,394172527	0,328503649	0,277323824
0,019878477	0,858742637	0,121378886
0,562090088	0,123990583	0,313919329
0,126474617	0,177669184	0,695856199
0,036531041	0,783592748	0,179876211
0,542772814	0,409377856	0,047849331
0,046417787	0,504852981	0,448729232
0,262385485	0,240335823	0,497278692
0,551233892	0,316064604	0,132701504
0,147748367	0,302429624	0,549822009



3. Menghitung Pusat Cluster

Setelah matriks partisi awal U terbentuk, langkah selanjutnya adalah melakukan perhitungan pusat *cluster*. Perhitungan nilai pusat cluster sesuai dengan diagram alir proses perhitungan pusat cluster pada Gambar (3.3). Input dari proses perhitungan pusat *cluster* adalah data set dan matriks partisi awal U . Output proses ini menghasilkan matriks pusat *cluster* berukuran jumlah *cluster* \times jumlah atribut.

Perhitungan nilai Pusat Cluster didasarkan pada persamaan (2.3) sehingga dapat dihitung 2 pusat *cluster* V_{kj} pada iterasi 1 dengan $k=1,2$; dan $j=1,2,3,4,5,6$ sebagai berikut :

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

$$V_{11} = \frac{((0,480615002)^2 * 1) + ((0,513792734)^2 * 1) + ((0,177525443)^2 * 1) + \dots + ((0,147748367)^2 * 2)}{(0,480615002)^2 + (0,513792734)^2 + (0,177525443)^2 + \dots + (0,147748367)^2}$$

$$V_{11} = 1,52338$$

Hasil dari perhitungan pusat cluster dapat ditunjukkan pada tabel (3.3).

Tabel 3.3 Perhitungan Pusat Cluster Iterasi 1

Cluster (k)	ATRIBUT (j)					
	1	2	3	4	5	6
1	1,5233	50,1544	11,6020	8172,950	287062,669	5,088
2	1,5117	56,2317	10,7960	9250,269	260596,970	4,928
3	1,6596	58,6833	10,4940	10357,225	319882,507	4,816

Pada *cluster* ke-1 terdapat 6 pusat *cluster*. Secara berturut-turut dari kiri ke kanan adalah pusat *cluster* Jenis Kelamin, Umur, Hemoglobin, Lekosit, Trombosit, dan eritrosit. Pada *cluster* ke-2 juga sama dengan *cluster* 1 yaitu pusat *cluster* dari atribut Jenis Kelamin, Umur, Hemoglobin, Lekosit, Trombosit, dan eritrosit. Pada *cluster* ke-3 juga sama dengan *cluster* 1 yaitu pusat *cluster* dari atribut Jenis Kelamin, Umur, Hemoglobin, Lekosit, Trombosit, dan eritrosit.

4. Menghitung fungsi objektif.

Langkah selanjutnya setelah perhitungan pusat *cluster* adalah perhitungan nilai fungsi objektif. Perhitungan nilai fungsi objektif sesuai dengan diagram alir perhitungan fungsi objektif pada Gambar (3.6).

Proses perhitungan fungsi objektif menggunakan input data set (X_{ij}), pusat *cluster* (V_{kj}) dan matriks awal (μ_{ik}). Proses perhitungan fungsi objektif akan menghasilkan *output* berupa nilai fungsi objektif iterasi ke- t (P_t). Perhitungan fungsi objektif pada iterasi pertama P_1 dapat dihitung dengan menggunakan persamaan (2.4) sebagai berikut :

$$L_1 = \left(\left[\sum_{j=1}^6 (X_{ij} - V_{1j})^2 \right] (\mu_{i1})^2 \right)$$

$$L_1 = \left(\left((1 - 1,52338)^2 + (59 - 50,15443)^2 + (13,7 - 11,60206)^2 + (4500 - 8172,95088)^2 + (315000 - 287062,66985)^2 + (5,2 - 5,08822)^2 \right) + (0,480615002)^2 \right)$$

$$L_1 = 183403230,2$$

$$L_2 = \left(\left[\sum_{j=1}^6 (X_{ij} - V_{2j})^2 \right] (\mu_{i2})^2 \right)$$

$$L_2 = \left(\left((1 - 1,51178)^2 + (79 - 56,23174)^2 + (13,2 - 10,79608)^2 + (11000 - 9250,26961)^2 + (317000 - 260596,97089)^2 + (5,8 - 4,92890)^2 \right) + (0,098153951)^2 \right)$$

$$L_2 = 28731632,08$$

$$L_3 = \left(\left[\sum_{j=1}^6 (X_{ij} - V_{3j})^2 \right] (\mu_{i3})^2 \right)$$

$$L_3 = \left(\left((1 - 1,65969603)^2 + (48 - 58,6833688)^2 + (13,7 - 10,4940117)^2 + (11000 - 10357,225)^2 + (317000 - 319882,507)^2 + (5,2 - 4,81650543)^2 \right) + (0,42123104)^2 \right)$$

$$L_3 = 10317167,49$$

$$P_1 = \sum_{i=1}^{20} \sum_{k=1}^3 \left(\left[\sum_{j=1}^6 (X_{ij} - V_{1j})^2 \right] (\mu_{1k})^2 \right) = 55586749870,6301$$

Detail perhitungan fungsi objektif dapat ditunjukkan pada Tabel (3.4).

Tabel 3.4 Perhitungan Fungsi Objektif

$\left[\sum_{j=1}^6 (X_{ij} - V_{1j})^2 \right] (\mu_{i1})^2$	$\left[\sum_{j=1}^6 (X_{ij} - V_{2j})^2 \right] (\mu_{i2})^2$	$\left[\sum_{j=1}^6 (X_{ij} - V_{3j})^2 \right] (\mu_{i3})^2$	L1+L2+L3
L1	L2	L3	
183403230,2	28731632,08	10317167,49	222452029,8
238703113,2	4206088,226	1765222,444	244674423,9
28497251,9	535774432,8	1482612,548	565754297,3
264076558,6	134275302,5	1240109,293	399591970,4
7439193,623	2935458,788	906230628,2	916605280,6
181342080,3	557809841,4	181569,7106	739333491,4
336783746,1	81129474,46	27714,34493	417940934,9
22726875,39	186975,1445	150383142	173296992,6
65962970,11	890862923,8	39608383,54	996434277,4
2370510959	1438776336	1264553676	5073840972
784997864,2	215388327,4	831049614,7	1831435806
6586641,539	7776989098	386542851,7	8170118591
482314540,7	2440512,644	509374682	994129735,4
646425605	1633136238	13692780787	15972342630
14983526,19	10764885245	172977726,8	10952846498
285649709,8	5312012,795	9387098,721	300348821,3
12194399,31	605909348	2345158773	2963262520
526065165	214730651,2	3561824031	4302619847
16339886,65	38538480,87	28314588,15	83192955,67
3654087,12	142089031,2	120784677,4	266527795,7
		Pt	55586749871

5. Menghitung perbaikan Matriks partisi U

Setelah perhitungan fungsi objektif terselesaikan langkah selanjutnya adalah melakukan perubahan matriks partisi U. Perhitungan nilai perubahan matriks partisi U sesuai dengan diagram alir proses perubahan matriks partisi U pada Gambar (3.5). Perubahan matriks partisi U dapat dihitung dengan menggunakan persamaan (2.5) sehingga dapat memperbaiki nilai derajat keanggotaan setiap elemen matriks partisi U (μ_{ik}). Contoh perhitungan perubahan matriks partisi U sebagai berikut:

$$L_1 = \left[\sum_{j=1}^m (X_{ij} - V_{ij})^2 \right]^{\frac{-1}{w-1}}$$

$$L_1 = \left[\sum_{j=1}^6 (X_{1j} - V_{1j})^2 \right]^{\frac{-1}{3-1}}$$

$$L_1 = \left[\left((1 - 1,52338)^2 + (59 - 50,15443)^2 + (13,7 - 11,60206)^2 + (4500 - 8172,95088)^2 + (315000 - 287062,66985)^2 + (5,2 - 5,08822)^2 \right) \right]^{-1}$$

$$L_1 = 1,25947E - 09$$

$$L_2 = \left[\sum_{j=1}^m (X_{ij} - V_{ij})^2 \right]^{\frac{-1}{w-1}}$$

$$L_2 = \left[\sum_{j=1}^6 (X_{2j} - V_{2j})^2 \right]^{\frac{-1}{3-1}}$$

$$L_2 = \left[\left((1 - 1,51178)^2 + (79 - 56,23174)^2 + (13,2 - 10,79608)^2 + (11000 - 9250,26961)^2 + (317000 - 260596,97089)^2 + (5,8 - 4,92890)^2 \right) \right]^{-1}$$

$$L_2 = 3,35317E - 10$$

$$L_3 = \left[\sum_{j=1}^m (X_{ij} - V_{ij})^2 \right]^{\frac{-1}{w-1}}$$

$$L_3 = \left[\sum_{j=1}^6 (X_{3j} - V_{3j})^2 \right]^{\frac{-1}{3-1}}$$

$$L_3 = \left[\left((1 - 1,659696039)^2 + (48 - 58,68336883)^2 + (13,7 - 10,49401172)^2 + (11000 - 10357,22558)^2 + (317000 - 319882,5078)^2 + (5,2 - 4,816505432)^2 \right) \right]^{-1}$$

$$L_3 = 1,71981E - 08$$

Hasil Proses perhitungan perbaikan matriks partisi U ditunjukkan pada Tabel (3.5).

Tabel 3.5 Perhitungan perbaikan matriks partisi U

$\left[\sum_{j=1}^6 (X_{i1} - V_{i1})^2 \right]^{-1}$	$\left[\sum_{j=1}^6 (X_{i2} - V_{i2})^2 \right]^{-1}$	$\left[\sum_{j=1}^6 (X_{i3} - V_{i3})^2 \right]^{-1}$	$\sum_{k=1}^3 \left[\sum_{j=1}^6 (X_{ij} - V_{ij})^2 \right]^{-1}$
L1	L2	L3	LT=L1+L2+L3
1,25947E-09	3,35317E-10	1,71981E-08	1,87929E-08
1,10591E-09	3,14034E-10	1,14647E-07	1,16067E-07

1,10591E-09	3,14035E-10	1,14651E-07	1,16071E-07
1,28119E-09	3,37695E-10	3,40189E-08	3,56378E-08
1,34688E-09	5,42852E-08	2,76655E-10	5,59087E-08
1,26825E-09	3,37524E-10	4,12318E-08	4,28376E-08
5,63098E-10	3,89727E-09	1,77619E-10	4,63799E-09
1,97764E-08	2,65576E-09	6,27961E-10	2,30601E-08
2,48706E-10	7,29853E-10	1,0854E-10	1,0871E-09
6,10948E-11	4,19445E-11	1,10483E-10	2,13522E-10
1,97927E-10	5,01024E-10	9,25438E-11	7,91494E-10
5,99932E-11	9,48232E-11	3,81144E-11	1,92931E-10
6,55061E-10	6,29936E-09	1,93463E-10	7,14788E-09
2,4745E-11	1,93287E-11	3,53629E-11	7,94366E-11
8,90656E-11	5,70389E-11	1,8705E-10	3,33154E-10
1,03134E-09	3,15493E-08	2,43905E-10	3,28245E-08
1,76689E-10	4,20651E-10	8,58611E-11	6,83201E-10
1,3087E-10	2,68994E-10	6,94268E-11	4,69291E-10
1,85961E-08	2,59213E-09	6,2193E-10	2,18102E-08
5,97402E-09	6,43707E-10	2,50284E-09	9,12056E-09

Lanjutan perhitungan perbaikan matrik partisi U

μ	μ^2	μ^3
L1/LT	L2/LT	L3/LT
0,067018448	0,017842758	0,915138794
0,009528172	0,002705634	0,987766194
0,009527861	0,002705544	0,987766596
0,035950307	0,009475751	0,954573942
0,024090728	0,970960945	0,004948328
0,029606041	0,007879156	0,962514803
0,121410143	0,840293226	0,038296631
0,857601665	0,115166846	0,02723149
0,22877919	0,671377061	0,099843749
0,28612835	0,196440948	0,517430702
0,250067093	0,633009968	0,116922939
0,310957247	0,491488154	0,197554599
0,091644009	0,881290161	0,02706583
0,311506962	0,243321993	0,445171046
0,267340379	0,171208731	0,56145089
0,031419828	0,961149609	0,007430564
0,258618745	0,615706508	0,125674747
0,278867437	0,573192781	0,147939783
0,85263492	0,118849529	0,028515551
0,655005531	0,070577552	0,274416917

Hasil Matriks Perbaikan

0,067018448	0,017842758	0,915138794
0,009528172	0,002705634	0,987766194
0,009527861	0,002705544	0,987766596
0,035950307	0,009475751	0,954573942
0,024090728	0,970960945	0,004948328
0,029606041	0,007879156	0,962514803
0,121410143	0,840293226	0,038296631
0,857601665	0,115166846	0,02723149
0,22877919	0,671377061	0,099843749
0,28612835	0,196440948	0,517430702
0,250067093	0,633009968	0,116922939
0,310957247	0,491488154	0,197554599
0,091644009	0,881290161	0,02706583
0,311506962	0,243321993	0,445171046
0,267340379	0,171208731	0,56145089
0,031419828	0,961149609	0,007430564
0,258618745	0,615706508	0,125674747
0,278867437	0,573192781	0,147939783
0,85263492	0,118849529	0,028515551
0,655005531	0,070577552	0,274416917

Setelah dihitung perubahan matriks partisi kemudian dilakukan pemeriksaan kondisi berhenti. Kondisi berhenti diperiksa dengan menghitung selisih fungsi objektif iterasi ke satu dengan fungsi objektif iterasi sebelumnya. $|P1-P0| = |55586749870,63010-0| = 55586749870,63010 \gg \xi = 0,00001$ dan iterasi = 1 < maxIter (=300). Pada iterasi pertama ini syarat kondisi berhenti belum terpenuhi yaitu nilai fungsi objektif masih lebih besar dari nilai eror terkecil, dan iterasi masih kurang dari maksimum iterasi sehingga proses harus dilanjutkan ke iterasi 2 (t=2) dan seterusnya sampai didapatkan syarat kondisi berhenti.

Setelah iterasi ke-33, salah satu syarat kondisi berhenti terpenuhi, yaitu $|P33-P32| = |12503271179,4796 - 12503271179,4796| = 9,53674E-06 < \xi = 0,00001$ dengan pusat *cluster* (V_{kj}) akhir yang ditunjukkan pada Tabel (3.6) dan fungsi objektif iterasi ke-33 ditunjukkan pada tabel (3.7)

Tabel 3.6 Perhitungan Pusat *Cluster* Iterasi 33 (Pusat *Cluster* Akhir)

Cluster (k)	ATRIBUT (j)					
	1	2	3	4	5	6
1	1,4057	53,078	12,332	8519,982	303612,680	4,9749
2	1,7397	57,003	10,029	10032,195	218681,905	4,8700
3	1,3315	56,649	9,3474	10854,342	432744,268	4,9794

Tabel 3.7 Perhitungan Fungsi Objektif Iterasi 33 (P33)

$\sum_{j=1}^6 (X_{ij} - V_{1j})^2$	$(\mu_{i1})^2$	$\sum_{j=1}^6 (X_{ij} - V_{2j})^2$	$(\mu_{i2})^2$	$\sum_{j=1}^6 (X_{ij} - V_{3j})^2$	$(\mu_{i3})^2$	L1+L2+L3
L1		L2		L3		
138491799,3		2169845,223		1452554,013		142114198,5
173712960,8		3330934,569		2403673,084		179447568,4
173712394		3330912,236		2403656,944		179446963,2
124075043,8		1737093,12		1162244,334		126974381,3
406897674,1		449630157,1		26075031,28		882602862,5
129419992,9		1894578,678		1267920,543		132582492,2
97847826,34		473103547,8		9552949,353		580504323,5
405934772,9		60204098,67		9703592,747		475842464,3
400694,2814		50092397,55		58625,62003		50551717,45
7759293,785		2497718,702		300018208,3		310275220,8
46603,4151		18863990,56		7615,486751		18918209,46
432962175,7		2476261208		121636406,6		3030859790
140943139,1		506540921,2		12771392,84		660255453,2
215229510,4		100958010		2390187162		2706374682
199944266,4		52597606,27		1011689088		1264230961
315038734,7		508328334,5		22908989,36		846276058,6
350000,2098		53859040,37		60434,29663		54269474,87
12608059,64		353745016		2510506,759		368863582,4
408964345,1		60992813,31		9858620,463		479815778,8
13029512,2		25801,64762		9683,369665		13064997,21
				Pt		12503271179

Nilai matriks partisi U akhir pada iterasi ke-33 berisi informasi mengenai kecenderungan suatu data untuk masuk ke dalam kelompok kelas tingkat resiko penyakit anemia. Selain itu FCM juga memberikan hasil *Clustering* yang ditunjukkan pada tabel (3.8).

Tabel 3.8 Tabel Hasil *Clustering*

Data ke	Cluster ke-1	Cluster ke-2	Cluster ke-3	Hasil max	Hasil cluster
1	0,974511	0,015268	0,010221	0,974510642	1
2	0,968043	0,018562	0,013395	0,968042987	1
3	0,968043	0,018562	0,013395	0,968043093	1
4	0,977166	0,013681	0,009153	0,977165962	1
5	0,46102	0,509437	0,029543	0,509436524	2
6	0,976147	0,01429	0,009563	0,976146929	1
7	0,168557	0,814987	0,016456	0,814987102	2
8	0,853086	0,126521	0,020392	0,853086494	1
9	0,007926	0,990914	0,00116	0,990913859	2
10	0,025008	0,00805	0,966942	0,966942213	3
11	0,002463	0,997134	0,000403	0,997134037	2
12	0,142851	0,817016	0,040133	0,817016092	2
13	0,213468	0,767189	0,019343	0,767189282	2
14	0,079527	0,037304	0,883169	0,883169347	3
15	0,158155	0,041604	0,800241	0,80024072	3
16	0,372265	0,600665	0,02707	0,600664887	2
17	0,006449	0,992437	0,001114	0,992437104	2
18	0,034181	0,959013	0,006806	0,959013127	2
19	0,852336	0,127117	0,020547	0,852336188	1
20	0,997284	0,001975	0,000741	0,997283964	1

6. Perhitungan f-Measure

Pada perhitungan F-measure ini ambil hasil cluster[] dan classID[], classID adalah nilai kelas sebenarnya dari Dataset awal. Hasil dari pencocokan hasil *cluster* dan *class ID* ditunjukkan pada Tabel (3.9).

Tabel 3.9 Tabel Pencocokan Cluster dan Class

Record	Hasil Cluster	Class ID
1	1	normal
2	1	normal
3	1	normal
4	1	normal
5	2	normal

6	1	normal
7	2	ringan
8	1	ringan
9	2	ringan
10	3	ringan
11	2	ringan
12	2	ringan
13	2	ringan
14	3	berat
15	3	berat
16	2	berat
17	2	berat
18	2	berat
19	1	berat
20	1	berat

7. Mencari classID paling dominan pada suatu cluster

- Cluster 1 : 5 Normal, 1 Ringan, 2 Berat = Normal Class
- Cluster 2 : 1 Normal, 5 Ringan, 3 Berat = Ringan Class
- Cluster 3 : 0 Normal, 1 Ringan, 2 Berat = Berat Class

Jika :

- a. ClassID = ClusterDominan → Trus Positive
- b. ClassID cluster lain, masuk ClusterDominan → False Negative
- c. ClassID, bukan ClusterDominan → False Positive

Hasil dari pencarian classID paling dominan pada suatu cluster ditunjukkan pada Tabel (3.9).

Tabel 3.10 Tabel classID paling Dominan

Cluster 1	ClassID	Cluster 2	ClassID	Cluster 3	ClassID
1	Normal	5	Normal	10	Ringan
2	Normal	7	Ringan	14	Berat
3	Normal	9	Ringan	15	Berat
4	Normal	11	Ringan		
6	Normal	12	Ringan		

8	Ringan		13	Ringan		
19	Berat		16	Berat		
20	Berat		17	Berat		
			18	Berat		

- Untuk *cluster* 1 yang merupakan *cluster* ‘Normal’ mendapatkan nilai *true positive* 5 didapat dari record 1, 2, 3, 4 dan 6. *False negative* 1 didapat dari record 5 sedangkan *false positive* 3 didapat dari record 8,19, dan 20.
- Untuk *cluster* 2 yang merupakan *cluster* ‘Ringan’ mendapatkan nilai *true positive* 5 didapat dari record 7,9,11,12, dan 13. *False negative* 2 didapat dari record 8 dan 10, sedangkan *false positive* 4 dari record 5, 16, 17 dan 18.
- Untuk *cluster* 3 yang merupakan *cluster* ‘Berat’ mendapatkan nilai *true positive* 2 didapat dari record 14 dan 15. *False negative* 5 didapat dari record 16,17,18, 19 dan 20, sedangkan *false positive* 1 didapat dari record 10.

Hasil dari perincian True Positive, False Negative, False Positif pada setiap cluster ditunjukkan pada Tabel (3.11).

Tabel 3.11 Rincian True Positive, False Negative, False Positif tiap Cluster

Cluster	True Positif	False Negatif	False Positif
Cluster 1	5	1	3
Cluster 2	5	3	4
Cluster 3	2	5	1

Setelah didapat nilai true positive, false negative dan false positif, maka langkah selanjutnya yaitu menghitung *precision* dengan persamaan (2.7), *recall* dengan persamaan (2.6) dan *f-Measure* dengan persamaan (2.8). Perhitungannya didapat dari tabel (3.11).

Cluster 1 :

$$➤ \textit{Precision} = \frac{TP}{TP + FP} = \frac{5}{5 + 3} = \frac{5}{8} = 0.63$$

$$\begin{aligned} \text{➤ recall} &= \frac{TP}{TP + FN} = \frac{5}{5 + 1} = 0.83 \\ \text{➤ f-measure} &= \frac{2 \times \text{recall} \times \text{precision}}{\text{recall} + \text{precision}} = \frac{2 \times 0.83 \times 0.63}{0.83 + 0.63} = \frac{1.05}{1.46} = 0.72 \end{aligned}$$

Cluster 2 :

$$\begin{aligned} \text{➤ Precision} &= \frac{TP}{TP + FP} = \frac{5}{5 + 4} = \frac{5}{9} = 0.56 \\ \text{➤ Recall} &= \frac{TP}{TP + FN} = \frac{5}{5 + 3} = \frac{5}{8} = 0.63 \\ \text{➤ f-measure} &= \frac{2 \times \text{recall} \times \text{precision}}{\text{recall} + \text{precision}} = \frac{2 \times 0.63 \times 0.56}{0.63 + 0.56} = \frac{0.71}{1.19} = 0.60 \end{aligned}$$

Cluster 3 :

$$\begin{aligned} \text{➤ precision} &= \frac{TP}{TP + FP} = \frac{2}{2 + 1} = \frac{2}{3} = 0.67 \\ \text{➤ recall} &= \frac{TP}{TP + FN} = \frac{2}{2 + 5} = \frac{2}{7} = 0.29 \\ \text{➤ f-measure} &= \frac{2 \times \text{recall} \times \text{precision}}{\text{recall} + \text{precision}} = \frac{2 \times 0.29 \times 0.67}{0.29 + 0.67} = \frac{0.39}{0.96} = 0.41 \end{aligned}$$

Hasil dari perhitungan True Positive, False Negative, False Positif pada setiap cluster ditunjukkan pada Tabel (3.12).

Tabel 3.12 Tabel Hasil Perhitungan *Precision*, *Recall*, *f-Measure*

Cluster	Precision	Recall	F-Measure
Cluster 1	0.63	0,83	0,72
Cluster 2	0,56	0.63	0,60
Cluster 3	0.67	0,29	0,41

Setelah melakukan proses perhitungan dan mendapatkan hasil *Precision*, *Recall* dan *f-measure*. Kemudian menghitung rata-rata *f-Measure* untuk semua cluster : (Cluster 1+cluster 2 + cluster 3)/ 3 (0.72 + 0.60 + 0.41) / 3 = 0.58 , Jadi hasil evaluasi clustering *Fuzzy C-Means* yang berupa *f-Measure* adalah 0.58.

3.7 Rancangan Antarmuka

Pada perancangan antarmuka ini bertujuan untuk mewakili keadaan sebenarnya dari implementasi program yang akan dibangun. Implementasi program ini dibagi menjadi 4 halaman yaitu halaman input data, halaman *Clustering*, halaman *fuzzy c-means* dan halaman evaluasi.

1. Perancangan Antarmuka halaman input data

Antarmuka halaman input merupakan antarmuka untuk menampilkan data yang diinputkan untuk *cluster*. Rancangan antarmuka input data ditunjukkan pada Gambar (3.9).

Implementasi Algoritma fuzzy C-Means Untuk Pengelompokan Tingkat Penyakit Anemia

Input Data
Clustering
Fuzzy C-Means
Evaluasi

INPUT DATA

Jenis Kelamin	Umur	Hemoglobin	Lekosit	Trombosit	Eritrosit	Class

Browse Data

Gambar 3.9 Rancangan Antarmuka nilai input

Berikut ini keterangan rancangan antarmuka halaman input data pada gambar (3.9)

- a. Input data digunakan untuk memasukkan data yang akan digunakan untuk perhitungan.
 - b. Input data terdiri dari 6 atribut dan class yaitu jenis kelamin, umur, hemoglobin, lekosit, trombosit, dan eritrosit
2. Perancangan antarmuka halaman *Clustering*

Antarmuka *clustering* merupakan antarmuka untuk menampilkan hasil *clustering* yang terbentuk terhadap data uji yang berisi jumlah cluster, pangkat, Max iterasi dan eror terkecil yang diharapkan. Rancangan antarmuka *clustering* ditunjukkan pada Gambar (3.10).

Implementasi Algoritma fuzzy C-Means Untuk Pengelompokan Tingkat Penyakit Anemia

CLUSTERING

Jumlah cluster :
 Pangkat (w) : 2
 Max iterasi :
 Eror Terkecil :

Gambar 3.10 Rancangan Antarmuka *Clustering*

Berikut ini keterangan rancangan antarmuka *clustering* pada gambar (3.10)

- a. Jumlah *cluster* digunakan untuk memasukkan banyaknya jumlah *cluster*.
 - b. Max iterasi digunakan memasukkan nilai maksimum iterasi yang diharapkan.
 - c. Eror Terkecil digunakan untuk memasukkan nilai maksimum iterasi yang diharapkan.
3. Perancangan Antarmuka halaman *Fuzzy C-Means*

Antarmuka halaman *Fuzzy C-Means* merupakan antarmuka untuk menampilkan hasil dari perhitungan *fuzzy c-means* yang berisi bilangan random, derajat awal, pusat cluster, fungsi objektif dan derajat baru yang terbentuk terhadap data uji. Rancangan antarmuka *Fuzzy C-Means* ditunjukkan pada Gambar (3.11).

Implementasi Algoritma fuzzy C-Means Untuk Pengelompokan Tingkat Penyakit Anemia

Fuzzy C-Means

Bilangan Random	Derajat Awal	Pusat Cluster	Fungsi Objektif	Derajat Baru

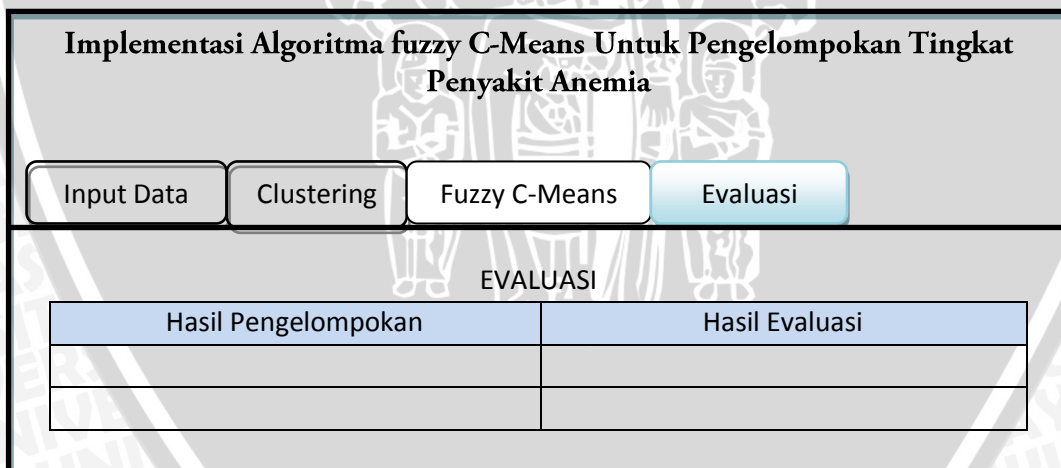
Gambar 3.11 Rancangan Antarmuka *Fuzzy C-Means*

Berikut ini keterangan rancangan antarmuka *Fuzzy C-Means* pada gambar (3.11) Pada tab derajat keanggotaan terdapat submenu diantaranya :

- a. Membangkitkan bilangan random
digunakan untuk menampilkan nilai bilangan random
- b. Derajat Awal
Digunakan untuk menampilkan nilai matriks derajat keanggotaan awal.
- c. Pusat *Cluster*
Digunakan untuk menampilkan hasil perhitungan pusat *cluster*.
- d. Fungsi Objektif
Digunakan untuk menampilkan hasil perhitungan fungsi objektif.
- e. Derajat Baru
Digunakan untuk menampilkan nilai dari derajat keanggotaan baru.

4. Perancangan Antarmuka Halaman Evaluasi

Antarmuka halaman evaluasi merupakan antarmuka untuk menampilkan hasil dari pengelompokan setiap cluster, hasil True Positive, False Negative, False Positif dan rata – rata *Precision, Recall, f-Measure*. Rancangan antarmuka evaluasi ditunjukkan pada Gambar (3.12).



Gambar 3.12 Rancangan Antarmuka Evaluasi

Berikut ini keterangan rancangan antarmuka evaluasi pada gambar (3.12).

Pada tab Fuzzy C-means terdapat submenu diantaranya :

- a. Tabel Pengelompokan
Digunakan untuk menampilkan data pada setiap *cluster*.

b. Tabel Hasil Evaluasi

Digunakan untuk menampilkan hasil perhitungan *precision*, *recall* dan *f-measure*

3.8 Rancangan Pengujian Evaluasi

Setelah program selesai diimplementasikan, langkah selanjutnya adalah melakukan pengujian dan evaluasi terhadap sistem tersebut. Uji coba ini bertujuan mengevaluasi ketepatan hasil *clustering* yang dilakukan oleh program. Hal ini bertujuan mengetahui keefektifan program yang dibuat menggunakan standart *precision*, *recall* dan *f-measure*.

Pada pengujian program ini digunakan input dataset penyakit anemia yang berupa format excel. Setelah data tersebut dikelompokkan dengan menggunakan *Fuzzy C-Means* maka dihitung nilai *precision*, *recall* dan *f-measure*. Dari hasil perhitungannya inilah yang nantinya digunakan untuk mencari nilai *cluster* yang paling optimal. Penentuan saat mendapatkan nilai *f-measure*, maka terlebih dahulu dilakukan pengujian terhadap *input* parameter *error* minimum yang diharapkan (ξ) pada masing-masing *cluster* (c) dengan tujuan mendapatkan nilai *error* minimum (ξ) pada saat nilai fungsi objektif telah cenderung konvergen. Nilai *error* minimum (ξ) pada saat nilai telah konvergen lah yang nantinya digunakan untuk penentuan dalam menghitung nilai *f-measure*.

Perhitungan dalam akurasi yang didapat dihitung berdasarkan rata-rata *precision*, *recall* dan *f-measure* yang dilakukan pada masing-masing *cluster*. Rancangan tabel uji coba dalam penelitian ini ditunjukkan pada tabel (3.13).

Tabel 3.13 Tabel Pengujian Evaluasi

Cluster	Rata-Rata Precision	Rata-Rata Recall	Rata-Rata f-Measure