

**PERBANDINGAN *FUZZY C-MEANS CLUSTERING* DAN
FUZZY SUBTRACTIVE CLUSTERING PADA KASUS RISIKO
PENYAKIT JANTUNG KORONER (PJK)**

SKRIPSI

Diajukan untuk memenuhi persyaratan memperoleh gelar Sarjana Komputer

UNIVERSITAS BRAWIJAYA



Disusun oleh :

ANGGUN K DIANITA

0910963068

**KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
PROGRAM STUDI TEKNIK INFORMATIKA/ILMU KOMPUTER
PROGRAM TEKNOLOGI INFORMASI DAN ILMU KOMPUTER**

UNIVERSITAS BRAWIJAYA

MALANG

2014

LEMBAR PERSETUJUAN

PERBANDINGAN *FUZZY C-MEANS CLUSTERING* DAN *FUZZY SUBTRACTIVE CLUSTERING* PADA KASUS RISIKO PENYAKIT JANTUNG KORONER (PJK)

SKRIPSI

KONSENTRASI KOMPUTASI CERDAS DAN VISUALISASI

Diajukan untuk memenuhi persyaratan memperoleh gelar Sarjana Komputer



Disusun oleh :

ANGGUN K DIANITA

0910963068

Skripsi ini telah disetujui oleh dosen pembimbing
pada tanggal 17 Desember 2013

Dosen Pembimbing I,

Dosen Pembimbing II,

Drs. Marji, MT
NIP.19670801 199203 1 001

Suprpto, ST, MT
NIP.19710727 199603 1 001

LEMBAR PENGESAHAN

PERBANDINGAN *FUZZY C-MEANS CLUSTERING* DAN *FUZZY SUBTRACTIVE CLUSTERING* PADA KASUS RISIKO PENYAKIT JANTUNG KORONER (PJK)

SKRIPSI

Untuk memenuhi sebagian persyaratan mencapai gelar Sarjana Komputer

Disusun Oleh:
ANGGUN K DIANITA

0910963068

Skripsi ini telah diuji dan dinyatakan lulus
tanggal 2 Januari 2014

Dosen Penguji I

Edy Santoso, S.Si., M.Kom

NIP.19740414 200312 1 004

Dosen Penguji II

Budi Darma Setiawan, S.Kom., M.Cs

NIP.841015 06 1 1 0090

Dosen Penguji III

Indriati, ST., M.Kom

NIK.831013 06 1 2 0035

Mengetahui

Ketua Program Studi Informatika/Ilmu Komputer

Drs. Marji, MT

NIP.19670801 199203 1 001

LEMBAR PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Anggun K. Dianita

NIM : 0910963068

Program Studi : Ilmu Komputer

Jurusan : Ilmu Komputer

Fakultas : Program Teknologi Informasi dan Ilmu Komputer

Penulis skripsi berjudul : PERBANDINGAN FUZZY C-MEANS
CLUSTERING DAN FUZZY SUBTRACTIVE
CLUSTERING PADA KASUS RISIKO
PENYAKIT JANTUNG KORONER (PJK)

Dengan ini menyatakan bahwa:

1. Isi dari skripsi yang saya buat adalah benar-benar karya sendiri dan tidak menjiplak karya orang lain, selain nama-nama yang termaktub di isi dan tertulis di daftar pustaka dalam skripsi ini.
2. Apabila di kemudian hari ternyata skripsi yang saya tulis terbukti hasil jiplakan, maka saya bersedia menanggung segala resiko yang akan saya terima.

Demikian pernyataan ini dibuat dengan segala kesadaran dan penuh tanggung jawab dan digunakan sebagaimana mestinya.

Malang, 17 Desember 2013

Yang menyatakan,

Anggun K Dianita

NIM. 0910963068

KATA PENGANTAR

Puji syukur kehadiran Tuhan Yang Maha Esa yang telah memberikan kelancaran pada penulis dalam menyelesaikan skripsi ini sebagai salah satu syarat untuk menyelesaikan studi di Program Teknologi Informasi dan Ilmu Komputer, Universitas Brawijaya.

Dalam penulisan skripsi ini, tentunya banyak pihak yang telah memberikan bantuan baik moril maupun materiil, petunjuk, dan bimbingan. Oleh karena itu, penulis ingin menyampaikan ucapan terima kasih yang sebesar-besarnya kepada :

1. Drs. Marji, MT selaku Dosen Pembimbing I yang telah memberikan bimbingan, arahan, dan nasehat untuk kesempurnaan tugas akhir ini.
2. Suprpto, ST MT selaku Dosen Pembimbing II yang telah memberikan bimbingan, arahan, dan nasehat untuk kesempurnaan tugas akhir ini.
3. Kedua orang tua penulis Bapak Soeharto dan Ibu Tuty Herawati yang tidak pernah lelah untuk mendidik, mengajarkan, membantu, menasehati, merawat, mendorong penulis untuk bekerja keras dan yang selalu mendoakan agar menjadi orang yang baik dan sukses.
4. Kedua kakak penulis Handono Kusuma Prasetyo dan Ratih Kusuma Dianata, SH yang selalu mendukung serta memberikan motivasi dalam menyelesaikan penulisan tugas akhir ini.
5. Penulis ucapkan juga kepada Khoirul Sholeh yang telah meluangkan waktunya dalam membantu, mengarahkan dalam menyelesaikan penulisan tugas akhir ini.
6. Sahabatku Silvi Agustina, Alief Ramadanie, Inas Ulvy Maulina, Mufaroha, dan teman-teman badminton yang selalu menemani dan menghibur selama masa kuliah serta teman-teman di Ilmu Komputer angkatan 2009, khususnya teman-teman Ilkom B'09 terima kasih atas kerjasamanya selama ini.
7. Dan semua pihak dan teman-teman yang tidak bisa penulis sebutkan satu-persatu terima kasih atas bantuannya, dorongan, masukan, dan doa sehingga tugas akhir ini diselesaikan oleh penulis.

Penulis menyadari bahwa skripsi ini masih jauh dari kesempurnaan, maka saran dan kritik yang konstruktif dari semua pihak sangat diharapkan demi penyempurnaan selanjutnya dan semoga skripsi ini bisa memberi manfaat untuk penelitian selanjutnya.

Akhirnya hanya kepada Tuhan Yang Maha Esa kita kembalikan semua urusan dan semoga skripsi ini dapat bermanfaat bagi semua pihak, khususnya bagi penulis dan para pembaca pada umumnya.

Malang, 17 Desember 2013

Penulis

UNIVERSITAS BRAWIJAYA



**PERBANDINGAN FUZZY C-MEANS CLUSTERING DAN FUZZY
SUBTRACTIVE CLUSTERING PADA KASUS RISIKO PENYAKIT
JANTUNG KORONER (PJK)**

ABSTRAK

Pada teknik pemodelan *fuzzy* telah banyak digunakan dalam menyelesaikan permasalahan ketidakpastian. Hal ini dikarenakan logika *fuzzy* dapat dipakai untuk memodelkan proses berpikir manusia yang menyangkut unsur ketidakpastian, keraguan dan linguistik dimana metode klasik yang biasanya digunakan seperti metode matematis sulit untuk diterapkan karena kurang cukupnya pengetahuan. Penyakit jantung koroner (PJK) merupakan problema kesehatan utama di negara maju. Pada proses pengujian terdapat 3 skenario, dimana masing-masing percobaan dibagi menjadi percobaan 10 data, 20 data, dan 30 data, sehingga pada penelitian kali ini, dimana akan membahas mengenai perbandingan dua metode untuk mengetahui serta menguji seberapa bagus proses *clustering* data PJK dengan menggunakan algoritma *Fuzzy C-Means Clustering* dan *Fuzzy Subtractive Clustering* yang mungkin nantinya akan digunakan sebagai acuan untuk para pakar dalam proses mendiagnosa penyakit tersebut. Pada kasus PJK, didapatkan hasil akhir yang menunjukkan bahwa *Fuzzy C-Means Clustering* lebih bagus daripada *Fuzzy Subtractive Clustering* karena pada tingkat nilai validitas clusternya, FCM lebih unggul, sehingga FCM lebih efektif dari FSC.

Kata kunci : penyakit jantung koroner, *fuzzy c-means*, *fuzzy subtractive*

**COMPARISON OF FUZZY C-MEANS CLUSTERING AND FUZZY
SUBTRACTIVE CLUSTERING IN THE CASE OF THE RISK CORONARY
HEART DISEASE (CHD)**

ABSTRACT

Fuzzy modeling technique have been widely used in solving problems of uncertainty. This is because fuzzy logic can be used to model human thinking process which involving the element of uncertainty, doubt, and linguistic where classic method like mathematical method is hard to implement for the lack of knowledge. Coronary heart disease (CHD) is a major health problem in most of developed country. System testing process consists of 3 scenario where each testing were divided into 10 data, 20 data, and 30 data. This study will do a comparison between the two methods to find out and test the accuracy of CHD data clustering with Fuzzy C-means Clustering methods and Fuzzy Subtractive Clustering which which may later be used as a reference for specialists in the process of diagnosing the disease. In the case of CHD, the final result show that by the level of cluster validity value Fuzzy C-Means Clustering is more effective than Fuzzy Subtractive Clustering.

Key phrases : *coronary heart disease, fuzzy c-means, fuzzy subtractive*

DAFTAR ISI

	Halaman
LEMBAR PERSETUJUAN	ii
LEMBAR PENGESAHAN	iii
LEMBAR PERNYATAAN	iv
KATA PENGANTAR	v
ABSTRAK	vii
ABSTRACT	viii
DAFTAR ISI	ix
DAFTAR GAMBAR	xiii
DAFTAR TABEL	xv
KODE PROGRAM	xvi
BAB I PENDAHULUAN	1
2.1 Latar Belakang	1
2.2 Rumusan Masalah	3
2.3 Batasan Masalah	3
2.4 Tujuan	3
2.5 Manfaat	4
2.6 Sitematika Penulisan	4
BAB II TINJAUAN PUSTAKA	5
2.1 Penyakit jantung Koroner	5
2.1.1 Umur	6
2.1.2 Kadar LDL	6
2.1.3 Kadar HDL	7
2.1.4 Kadar Kolesterol Total	7
2.1.5 Tekanan Darah (Tekanan darah sistolik)	8
2.2 Logika Fuzzy	8
2.2.1 Definisi Logika Fuzzy	8
2.2.2 Himpunan Fuzzy	9

2.3	<i>Clustering</i>	10
2.3.1	<i>Hard Clustering dan Fuzzy Clustering</i>	11
2.3.2	<i>Fuzzy C-Means Clustering</i>	11
2.3.3	<i>Fuzzy Subtractive Clustering</i>	14
2.3.4	Indeks Validasi	18

BAB III METODOLOGI DAN PERANCANGAN SISTEM 21

3.1	Studi Kepustakaan	22
3.2	Data Penelitian	22
3.3	Analisis dan Perancangan Sistem	22
3.3.1	Deskripsi Umum Sistem	22
3.3.2	Perancangan Proses	23

A. Proses *Clustering* menggunakan

<i>Fuzzy C-Means Clustering</i> (FCM)	23
1. Proses Pembentukan Matriks Partisi Awal	25
2. Proses Perhitungan Pusat <i>Cluster</i>	27
3. Proses Perhitungan Fungsi Obyektif	28
4. Proses Perubahan Matriks Partisi U	30
5. Proses Pengelompokkan Data	31

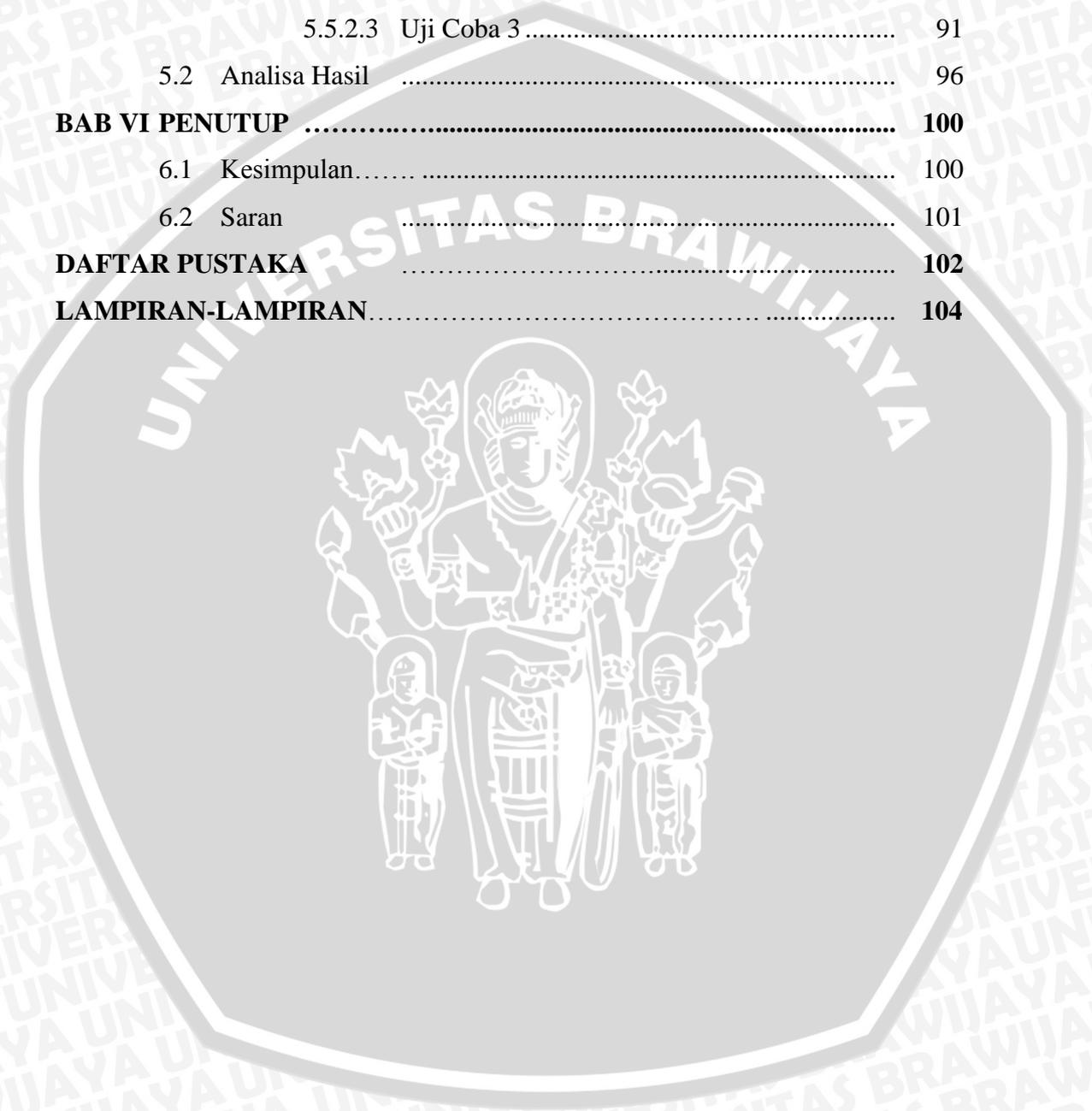
B. Proses *Clustering* menggunakan

<i>Fuzzy Subtractive Clustering</i>	33
1. Proses Normalisasi	34
2. Proses Penentuan Potensi Awal Tiap Data	36
3. Pencarian Titik dengan Potensi Tertinggi	37
4. Penentuan Pusat <i>Cluster</i>	38
5. Perhitungan Nilai Sigma	40
6. Derajat Keanggotaan	41
7. Pengelompokkan Data	42

3.3.3	Perhitungan Manual Proses	43
3.3.3.1	Perhitungan FCM <i>Clustering</i>	43
3.3.3.2	Perhitungan <i>Subtractive Clustering</i>	51
3.3.4	Perancangan Tabel dan Struktur Data	59
3.3.5	Perancangan Antarmuka	60

BAB IV IMPLEMENTASI	63
4.1 Lingkungan Implementasi	63
4.1.1 Lingkungan Implementasi Perangkat Keras	63
4.1.2 Lingkungan Implementasi Perangkat Lunak	63
4.2 Implementasi Program	64
4.2.1 Tahapan <i>Clustering</i> pada Metode	
<i>Fuzzy C-Means Clustering</i>	64
4.2.1.1 Proses Membangkitkan Bilangan Random ...	64
4.2.1.2 Proses Menghitung Pusat <i>Cluster</i>	65
4.2.1.3 Proses Menghitung Fungsi Obyektif	65
4.2.1.4 Proses Perubahan Matriks	66
4.2.1.5 Proses <i>Clustering</i> Data	67
4.2.2 Tahapan <i>Clustering</i> pada Metode	
<i>Fuzzy Subtractive Clustering</i>	68
4.2.2.1 Proses Normalisasi	68
4.2.2.2 Proses Menentukan Potensi Awal	68
4.2.2.3 Proses Hitung Titik Potensi Tertinggi	69
4.2.2.4 Proses Menghitung Pusat <i>Cluster</i>	70
4.2.2.5 Proses Menghitung Sigma	70
4.2.2.6 Proses Menghitung Derajat Keanggotaan	71
4.2.2.7 Proses Penglompokkan Data	72
4.2.3 Tahapan Perbandingan	73
4.2.3.1 Proses Hitung PC, MPC, CE pada	73
4.2.3.1.1 Proses Hitung PC	73
4.2.3.1.2 Proses Hitung MPC	74
4.2.3.1.3 Proses Hitung CE	74
4.3 Implementasi Antarmuka	75
4.3.1 Form Utama	75
4.3.2 Form Data Penyakit Jantung Koroner (PJK)	79
BAB V PENGUJIAN DAN ANALISA	80
5.1 Implementasi Uji Coba	80

5.5.1	Skenario Uji Coba	80
5.5.2	Hasil Uji Coba	80
5.5.2.1	Uji Coba 1	80
5.5.2.2	Uji Coba 2	86
5.5.2.3	Uji Coba 3	91
5.2	Analisa Hasil	96
BAB VI PENUTUP		100
6.1	Kesimpulan	100
6.2	Saran	101
DAFTAR PUSTAKA		102
LAMPIRAN-LAMPIRAN		104



DAFTAR GAMBAR

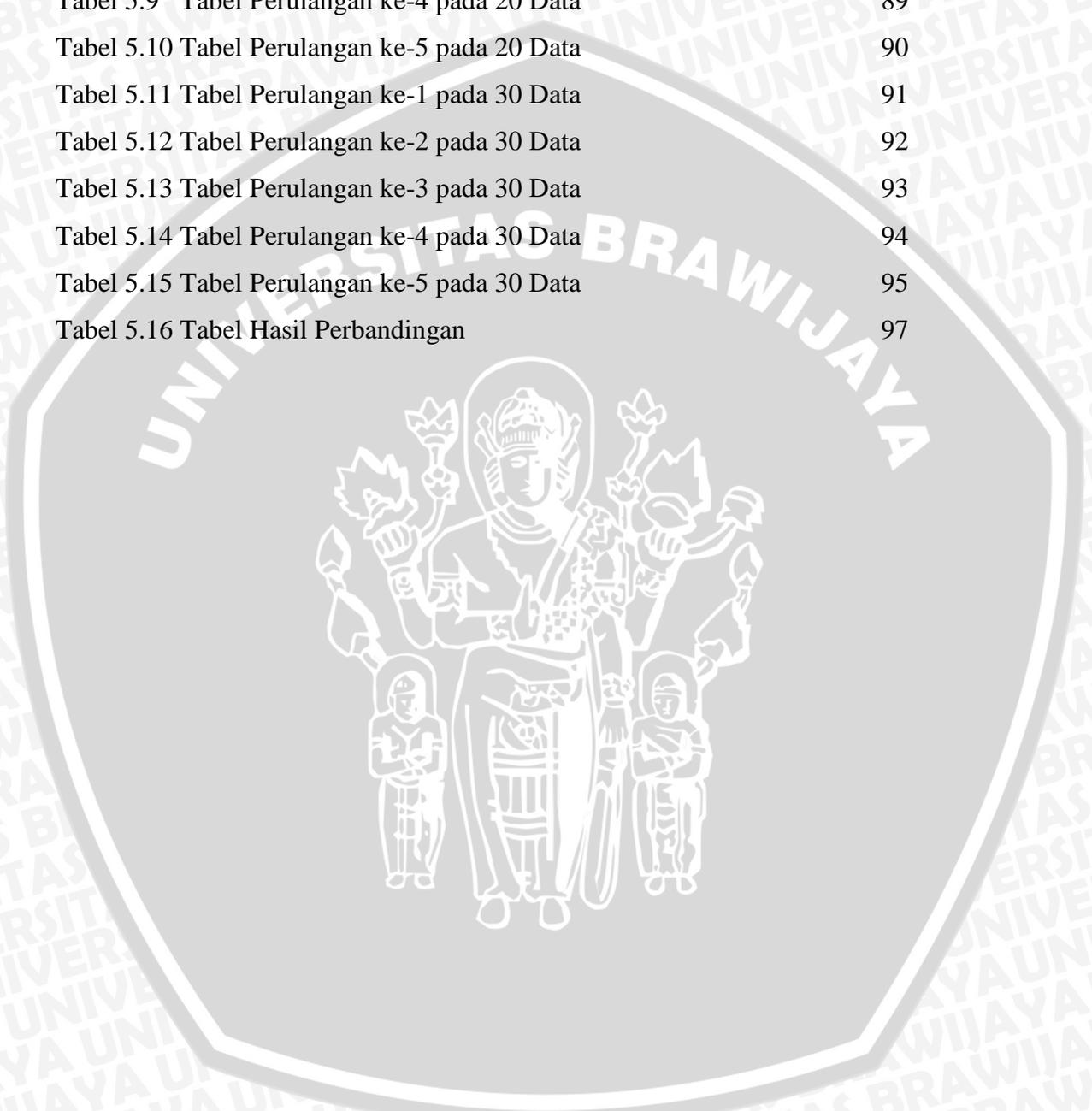
	halaman	
Gambar 2.1	Contoh Proses Clustering	10
Gambar 2.2	<i>Hard Clustering</i> dan <i>Fuzzy Clustering</i>	11
Gambar 2.3	Kurva Gauss	18
Gambar 3.1	Diagram Alur Penelitian	22
Gambar 3.2	Alur proses <i>clustering</i> menggunakan FCM	24
Gambar 3.3	Alur proses pembentukan matriks partisi awal	26
Gambar 3.4	Alur proses perhitungan pusat <i>cluster</i>	28
Gambar 3.5	Alur proses perhitungan fungsi obyektif	29
Gambar 3.6	Alur proses perubahan matriks partisi	31
Gambar 3.7	Alur proses pengelompokkan data FCM	32
Gambar 3.8	Alur proses <i>clustering</i> menggunakan <i>Subtractive Clustering</i>	34
Gambar 3.9	Alur proses normalisasi	35
Gambar 3.10	Alur proses penentuan potensi awal tiap data	36
Gambar 3.11	Alur proses pencarian titik dengan potensi tertinggi	37
Gambar 3.12	Alur proses penentuan pusat <i>cluster</i>	39
Gambar 3.13	Alur proses perhitungan nilai sigma	40
Gambar 3.14	Alur perhitungan derajat keanggotaan	41
Gambar 3.15	Alur proses pengelompokkan data <i>Fuzzy Subtractive</i>	42
Gambar 3.16	Tampilan antarmuka sistem	60
Gambar 3.17	Tampilan proses <i>clustering</i>	60
Gambar 3.18	Tampilan untuk pusat <i>cluster</i> yang telah terbentuk	61
Gambar 3.19	Tampilan untuk data yang sudah ter- <i>cluster</i>	61
Gambar 3.20	Tampilan hasil proses validitas <i>cluster</i>	62
Gambar 3.21	Tampilan hasil perbandingan	62
Gambar 4.1	Form Utama	76
Gambar 4.2	Tampilan Form Utama Setelah Proses Dilakukan	77
Gambar 4.3	Tampilan Hasil Proses Validitas <i>Cluster</i> pada Form Utama	78

Gambar 4.4	Tampilan Hasil Perbandingan pada Form Utama	78
Gambar 4.5	Form Data Penyakit Jantung Koroner (PJK)	79
Gambar 5.1	Grafik Perulangan ke-1 pada 10 Data	81
Gambar 5.2	Grafik Perulangan ke-2 pada 10 Data	82
Gambar 5.3	Grafik Perulangan ke-3 pada 10 Data	83
Gambar 5.4	Grafik Perulangan ke-4 pada 10 Data	84
Gambar 5.5	Grafik Perulangan ke-5 pada 10 Data	85
Gambar 5.6	Grafik Perulangan ke-1 pada 20 Data	86
Gambar 5.7	Grafik Perulangan ke-2 pada 20 Data	87
Gambar 5.8	Grafik Perulangan ke-3 pada 20 Data	88
Gambar 5.9	Grafik Perulangan ke-4 pada 20 Data	89
Gambar 5.10	Grafik Perulangan ke-5 pada 20 Data	90
Gambar 5.11	Grafik Perulangan ke-1 pada 30 Data	92
Gambar 5.12	Grafik Perulangan ke-2 pada 30 Data	93
Gambar 5.13	Grafik Perulangan ke-3 pada 30 Data	94
Gambar 5.14	Grafik Perulangan ke-4 pada 30 Data	95
Gambar 5.15	Grafik Perulangan ke-5 pada 30 Data	96
Gambar 5.6	Grafik Hasil Perbandingan	98

DAFTAR TABEL

	halaman
Tabel 3.1 Contoh data Penyakit Jantung Koroner (PJK)	43
Tabel 3.2 Bilangan random dan nilai Qi	44
Tabel 3.3 Contoh perhitungan pusat <i>cluster</i> 1	45
Tabel 3.4 Kuadrat derajat keanggotaan data ke-I	46
Tabel 3.5 Detail perhitungan fungsi obyektif	46
Tabel 3.6 Detail perhitungan derajat keanggotaan baru	47
Tabel 3.7 Kecenderungan data terhadap <i>cluster</i>	49
Tabel 3.8 Proses perhitungan PC pada FCM	49
Tabel 3.9 Proses perhitungan CE pada FCM	50
Tabel 3.10 Contoh data Penyakit Jantung Koroner (PJK)	51
Tabel 3.11 Tabel data ternormalisasi	52
Tabel 3.12 Contoh perhitungan potensi data pertama	52
Tabel 3.13 Data potensi awal	53
Tabel 3.14 Pusat <i>cluster</i> pada k1	53
Tabel 3.15 Tabel perhitungan potensi baru	54
Tabel 3.16 Potensi baru data ke-3	55
Tabel 3.17 Pusat <i>cluster</i> dari bentuk ternormalisasi yang telah dikembalikan	56
Tabel 3.18 Nilai Sigma	56
Tabel 3.19 Pengelompokkan data	56
Tabel 3.20 Proses perhitungan PC pada FS	57
Tabel 3.21 Proses perhitungan CE pada FS	58
Tabel 3.22 Tabel Dataset PJK	59
Tabel 5.1 Tabel Perulangan ke-1 pada 10 Data	81
Tabel 5.2 Tabel Perulangan ke-2 pada 10 Data	82
Tabel 5.3 Tabel Perulangan ke-3 pada 10 Data	83
Tabel 5.4 Tabel Perulangan ke-4 pada 10 Data	84
Tabel 5.5 Tabel Perulangan ke-5 pada 10 Data	85

Tabel 5.6	Tabel Perulangan ke-1 pada 20 Data	86
Tabel 5.7	Tabel Perulangan ke-2 pada 20 Data	87
Tabel 5.8	Tabel Perulangan ke-3 pada 20 Data	88
Tabel 5.9	Tabel Perulangan ke-4 pada 20 Data	89
Tabel 5.10	Tabel Perulangan ke-5 pada 20 Data	90
Tabel 5.11	Tabel Perulangan ke-1 pada 30 Data	91
Tabel 5.12	Tabel Perulangan ke-2 pada 30 Data	92
Tabel 5.13	Tabel Perulangan ke-3 pada 30 Data	93
Tabel 5.14	Tabel Perulangan ke-4 pada 30 Data	94
Tabel 5.15	Tabel Perulangan ke-5 pada 30 Data	95
Tabel 5.16	Tabel Hasil Perbandingan	97



KODE PROGRAM

		halaman
Kode Program 4.1	Proses Membangkitkan Bilangan Random	64
Kode Program 4.2	Proses Menghitung Pusat <i>Cluster</i>	65
Kode Program 4.3	Proses Perhitungan Fungsi Obyektif	66
Kode Program 4.4	Proses Perubahan Matriks	66
Kode Program 4.5	Proses <i>Clustering</i> Data	67
Kode Program 4.6	Proses Normalisasi	68
Kode Program 4.7	Proses Menentukan Potensi Awal	69
Kode Program 4.8	Proses Hitung Titik Potensi Tertinggi	69
Kode Program 4.9	Proses Menghitung Pusat <i>Cluster</i>	70
Kode Program 4.10	Proses Menghitung Sigma	71
Kode Program 4.11	Proses Menghitung Derajat Keanggotaan	71
Kode Program 4.12	Proses Pengelompokkan Data	72
Kode Program 4.13	Proses Hitung PC	73
Kode Program 4.14	Proses Hitung MPC	74
Kode Program 4.15	Proses Hitung CE	74

BAB I PENDAHULUAN

1.1. Latar Belakang

Dalam beberapa tahun terakhir, teknik pemodelan *fuzzy* telah banyak digunakan dalam menyelesaikan permasalahan ketidakpastian. Hal ini dikarenakan logika *fuzzy* dapat dipakai untuk memodelkan proses berpikir manusia yang menyangkut unsur ketidakpastian, keraguan dan *linguistik* dimana metode klasik yang biasanya digunakan seperti metode matematis sulit untuk diterapkan karena kurang cukupnya pengetahuan [PRA-05]. Karakter utama dari logika *fuzzy* adalah kemampuan untuk mengungkapkan pengetahuan melalui ilmu bahasa, memungkinkan suatu sistem untuk dijabarkan dengan mudah.

Penyakit jantung koroner (PJK) merupakan problema kesehatan utama di negara maju. Di Indonesia telah terjadi pergeseran kejadian Penyakit Jantung dan pembuluh darah dari urutan ke-10 tahun 1980 menjadi urutan ke-8 tahun 1986. Sedangkan penyebab kematian tetap menduduki peringkat ke-3. Banyak faktor yang mempengaruhi terjadinya Penyakit jantung Koroner sehingga usaha pencegahan harus bentuk multifaktorial juga. Pencegahan harus diusahakan sedapat mungkin dengan cara pengendalian faktor-faktor resiko PJK dan merupakan hal yang cukup penting dalam usaha pencegahan PJK, baik primer maupun sekunder. Pencegahan primer lebih ditujukan pada mereka yang sehat tetapi mempunyai risiko tinggi, sedangkan sekunder merupakan upaya memburuknya penyakit yang secara klinis telah diderita. Berbagai penelitian telah dilakukan selama 50 tahun lebih dimana didapatkan variasi insidens PJK yang berbeda pada geografis dan keadaan sosial tertentu yang makin meningkat sejak tahun 1930 dan mulai tahun 1960 merupakan penyebab kematian utama di negara industri. Mengapa didapatkan variasi insidens yang berbeda saat itu belum diketahui dengan pasti, akan tetapi didapatkan jelas terjadi pada keadaan-keadaan tertentu. Penelitian epidemiologis akhirnya mendapatkan hubungan yang jelas antara kematian dengan pengaruh keadaan sosial, kebiasaan merokok, pola diet, exercise, dan sebagainya yang dapat dibuktikan faktor-faktor yang dapat mempengaruhi terjadinya PJK antara lain: umur, kelamin ras, geografis, keadaan

sosial, perubahan masa, kolesterol, hipertensi, merokok, diabetes, obesitas, exercise, diet, perilaku dan kebiasaan lainnya, stress serta keturunan. Penelitian mengenai *clustering* pada data positif Penyakit Jantung Koroner (PJK) pernah dilakukan oleh Effendy, dkk (2008) dan Wahyuni (2011). Pada penelitiannya, Effendy, dkk melakukan penelitian mengenai prediksi PJK berdasarkan factor risiko menggunakan jaringan syaraf tiruan *Backpropagation*, sedangkan penelitian Wahyuni membahas mengenai diagnosis penyakit jantung coroner (PJK) berdasarkan factor resiko menggunakan metode FES, dimana pada keduanya hasil akhirnya sama-sama memiliki proses akurat, cepat, serta dapat meminimalisasi kesalahan. Tetapi adapun kelemahannya yaitu keduanya pada saat pengambilan proses keputusan masih dibutuhkan factor-faktor pendukung atau kebijakan lain. Pada penelitian kali ini, dimana akan membahas mengenai perbandingan dua metode untuk mengetahui serta menguji seberapa efektif proses *clustering* data PJK dengan menggunakan algoritma *Fuzzy C-Means Clustering* dan *Fuzzy Subtractive Clustering* yang mungkin nantinya akan digunakan sebagai acuan untuk para pakar dalam proses mendiagnosa penyakit tersebut. Jadi para pakar tidak ragu dengan adanya nilai perbandingan antara dua metode tersebut. Jadi dari dua metode yang telah diketahui hasil *clustering*-nya nantinya akan dihitung dengan menggunakan metode ukuran validasi *clustering* pada masing-masing metode, sehingga pada hasil akhirnya akan ditemukan hasil perbandingan yang paling efektif yang menunjukkan bahwa salah satu dari metode bahwa metode tersebut efektif digunakan untuk proses *clustering* pada kasus PJK.

Menurut Koczy (2002) algoritma *fuzzy clustering* memiliki kelebihan yaitu efisiensi dalam komputasi. Ada beberapa algoritma *fuzzy clustering* yang digunakan dalam penelitian ini, diantaranya *Fuzzy C-Means Clustering* dan *Fuzzy Subtractive Clustering*. Algoritma *Fuzzy C-Means Clustering* merupakan algoritma *clustering* yang terawasi, sehingga penentuan jumlah *cluster* algoritma tersebut ditentukan dari awal. Sedangkan untuk algoritma *Fuzzy Subtractive Clustering* merupakan algoritma *clustering* tidak terawasi, dimana jumlah *cluster* tidak diketahui sebelumnya. Menurut Sidhu (2010), *Fuzzy Subtractive Clustering* mempunyai kelebihan yaitu dapat memperkirakan pusat *cluster* secara baik

Berdasarkan latar belakang yang telah dikemukakan, maka judul yang diambil dalam skripsi ini adalah **“Perbandingan Fuzzy C-Means Clustering dan Fuzzy Subtractive Clustering pada Kasus Risiko Penyakit Jantung Koroner (PJK)”**.

1.2. Rumusan Masalah

Permasalahan yang ada penelitian ini dapat dirumuskan sebagai berikut:

1. Bagaimana tingkat keefektifan algoritma *Fuzzy C-Means Clustering* pada kasus risiko Penyakit Jantung Koroner (PJK)?
2. Bagaimana tingkat keefektifan algoritma *Fuzzy Subtractive Clustering* pada kasus risiko Penyakit Jantung Koroner (PJK)?
3. Bagaimana keefektifan perbandingan antara *Fuzzy C-Means Clustering* dan *Fuzzy Subtractive Clustering* pada kasus risiko Penyakit Jantung Koroner (PJK)?

1.3. Batasan Masalah

Untuk membatasi masalah yang dikaji, maka penulis membatasi permasalahan dan menitik beratkan penelitian terhadap masalah sebagai berikut:

1. Data yang digunakan adalah data positif Penyakit Jantung Koroner (PJK) yang diambil dari penelitian Effendy (2008) dan Wahyuni (2011).
2. Data didasarkan pada faktor risiko PJK, yaitu umur, kadar LDL, kadar HDL, kolesterol total, dan tekanan darah sistolik.
3. Algoritma perbandingan yang digunakan hanya *Fuzzy C-Means Clustering* dan *Fuzzy Subtractive Clustering*.
4. Tidak mempertimbangkan faktor PJK, jenis kelamin, dan tingkat risiko pada pasien.

1.4. Tujuan

Penelitian ini dibuat dengan tujuan untuk mengetahui nilai perbandingan yang paling efektif untuk proses *clustering* pada kasus risiko penyakit jantung koroner (PJK) antara dua metode *fuzzy clustering* yaitu *Fuzzy C-Means Clustering* dan *Fuzzy Subtractive Clustering*.

1.5. Manfaat

Manfaat yang dapat diambil dari penelitian ini yaitu menghasilkan aplikasi yang dapat digunakan untuk mengetahui metode yang paling efektif untuk digunakan dalam proses *clustering* pada kasus risiko penyakit jantung koroner (PJK) dengan melihat dari dua metode antara *Fuzzy C-Means Clustering* dan *Fuzzy Subtractive Clustering*.

1.6. Sistematika Penulisan

Penelitian ini disusun berdasarkan sistematika penulisan sebagai berikut:

1. BAB I PENDAHULUAN

Bab I ini membahas mengenai latar belakang masalah, rumusan masalah, batasan masalah, tujuan, dan manfaat penelitian, serta sistematika penulisan.

2. BAB II TINJAUAN PUSTAKA

Bab II ini menjelaskan tentang dasar teori yang digunakan dalam menyusun penelitian. Hal tersebut, meliputi penjelasan tentang penyakit jantung koroner, *Clustering*, *Fuzzy C-Means Clustering*, *Fuzzy Subtractive Clustering*, Logika Fuzzy, dan *Fuzzy Clustering Validation*.

3. BAB III METODOLOGI DAN PERANCANGAN SISTEM

Bab III membahas tentang langkah-langkah penyelesaian dalam penelitian seperti perhitungan manual serta membahas spesifikasi perangkat lunak, perancangan sistem, perancangan basis data, dan perancangan antar muka.

4. BAB IV IMPLEMENTASI

Bab IV membahas tentang implementasi dari algoritma metode yang digunakan ke dalam sistem yang telah dirancang dan mengimplementasikan ke dalam bentuk bahasa pemrograman.

5. BAB V PENGUJIAN DAN ANALISA

Bab V membahas tentang analisa dan pembahasan dari pengujian perangkat lunak serta perhitungan pengujian data.

6. BAB VI PENUTUP

Bab VI berisi kesimpulan dari pembahasan pada bab sebelumnya serta saran dari keseluruhan penelitian pada skripsi ini.

BAB II

TINJAUAN PUSTAKA

2.1. Penyakit Jantung Koroner

Penyakit Jantung Koroner (PJK) merupakan penyakit yang masih menjadi masalah baik di Negara maju maupun negara berkembang. PJK merupakan kelainan yang disebabkan oleh penyempitan pembuluh arteri yang mengalirkan darah ke otot jantung [ANW-04]. PJK disebabkan oleh pembentukan plak di dalam arteri pembuluh darah jantung. Plak terdiri atas kolesterol, kalsium, dan bahan lain di dalam pembuluh darah jantung (*arteri koronaria*) serta arteri di tempat lain. Proses ini disebut dengan pengerasan arteri atau *atherosclerosis* atau *atheroma* [DAV-03]. Di USA setiap tahunnya 550.000 orang meninggal karena penyakit ini. Di Eropa diperhitungkan 20 sampai 40.000 orang dari satu juta penduduk menderita PJK. Hasil survey yang dilakukan Departemen Kesehatan RI menyatakan jumlah orang penderita PJK di Indonesia dari tahun ke tahun terus meningkat. Bahkan, sekarang pada tahun 2000-an ini dapat dipastikan, kecenderungan penyebab kematian di Indonesia bergeser dari penyakit infeksi ke penyakit sistem peredaran darah (PJK) dan degenerative [MAJ-07].

Menurut WHO, 60% dari seluruh penyebab kematian jantung adalah PJK. Data WHO tahun 2011 menyatakan bahwa 7,2 juta kematian terjadi akibat PJK [KOM-09]. Penyakit Jantung Koroner (PJK) adalah suatu kelainan disebabkan oleh penyempitan atau penghambatan pembuluh arteri yang mengalirkan darah ke otot jantung. Bilamana penyempitan ini menjadi parah maka dapat terjadi serangan jantung. Adapun penyempitan pembuluh arteri ke otak dapat menimbulkan *stroke*. Kondisi tubuh yang semakin tua dan memburuk oleh bermacam-macam factor risiko, seperti tekanan darah tinggi, merokok, kadar kolesterol darah yang abnormal, dapat menyebabkan terjadinya serangan PJK [SOE-10].

Factor resiko PJK merupakan sejumlah factor yang berhubungan dengan penyakit jantung koroner. Factor risiko PJK dapat dibagi menjadi dua [KRI-02], yaitu:

1. Faktor risiko primer, menyebabkan gangguan arteri tanpa harus dibantu oleh factor lain (*independen*), termasuk risiko primer, yaitu *hiperlidemi*, merokok, dan *hipertensi*.
2. Factor risiko sekunder, menyebabkan gangguan arteri jika ditemukan factor lain secara bersamaan, termasuk *diabetes mellitus* (DM), *obesitas*, stress, kurang olah raga, alcohol, dan riwayat keluarga.

Factor risiko PJK yang menjadi batasan masalah penelitian ini akan dijelaskan sebagai berikut.

2.1.1. Umur

Umur merupakan masa perjalanan hidup manusia. Umur termasuk factor risiko penyakit jantung coroner karena menyebabkan perubahan di dalam jantung dan pembuluh darah. Pada umur lansia, orang menjadi kurang aktif dan berat badan meningkat. Tekanan darah meningkat sesuai umur karena arteri secara perlahan kehilangan elastisitas. Seiring dengan pertambahan umur, kadar kolesterol seseorang akan meningkat. Penderita PJK sering ditemui pada umur 60 tahun ke atas. Namun demikian, penderita PJK ada juga yang berumur di bawah 40 tahun. Secara umum, semakin tua umur, risiko PJK semakin meningkat pula [ANW-04].

2.1.2. Kadar LDL (*Low Density Lipoprotein*)

LDL (*Low Density Lipoprotein*) adalah jenis lipoprotein yang mengangkut kolesterol dan trigliserida dari hati ke jaringan perifer LDL merupakan jenis kolesterol yang bersifat buruk dan merugikan karena dapat menyebabkan penebalan dinding pembuluh darah dan penutupan arteri apabila terdapat pada kadar yang tinggi. Kadar LDL dinyatakan dengan peningkatan risiko PJK. Semakin tinggi kadar LDL, risiko PJK semakin meningkat pula [ANW-04]. Pada kadar LDL terdapat batas-batas yang diinginkan dan bahkan batas yang mendekati berbahaya. Untuk kadar LDL yang optimal bekisar <100, mendekati optimal antara 100-129, untuk yang diinginkan antara 130-159, kadar LDL yang memasuki tingkat tinggi bekisar antara 160-189, dan pada kadar LDL yang membahayakan atau sangat tinggi ≥ 190 [GAN-09].

2.1.3. Kadar HDL (*High Density Lipoprotein*)

HDL (*High Density Lipoprotein*) merupakan kolesterol lipoprotein yang berkepadatan rendah dan merupakan jenis kolesterol yang bersifat baik dan menguntungkan. Peranan kolesterol HDL adalah membawa kembali kolesterol buruk ke organ hati untuk dibuang sehingga mencegah penebalan dinding pembuluh darah atau mencegah terjadi proses aterosklerosis. Seseorang yang mempunyai kadar kolesterol HDL dalam kategori sangat baik masih berisiko terkena penyakit jantung coroner. Kadar HDL dinyatakan dalam satuan mg/dl. Semakin rendah kadar HDL, semakin besar kemungkinan terjadi PJK. Kadar HDL ini dapat dinaikkan dengan mengurangi berat badan dan banyak melakukan olah raga [ANW-04].

Dalam kadar HDL ada juga batas supaya lemak tersebut tidak menjadi kolesterol jahat. Supaya kadar HDL tidak membahayakan bagi seseorang yang memiliki penyakit Jantung coroner setidaknya nilai kadar HDL <40 yaitu rendah. Tetapi jika kadar HDL ≥ 190 yang sudah termasuk tinggi maka seseorang tersebut harus mengurangi dengan cara bila obesitas paling tidak dapat mengurangi berat badan tetapi apabila tubuh tidak obesitas tetapi kadar HDL tinggi, psling tidak dapat dilakukan dengan banyak berolahraga [GAN-09].

2.1.4. Kadar kolesterol total

Kolesterol total merupakan susunan dari banyak zat, termasuk trigliserida, LDL kolesterol, dan HDL kolesterol. Kolesterol yang ada di dalam zat makanan akan meningkatkan kadar kolesterol dalam darah. Sejauh pemasukan itu masih sesuai dengan kebutuhan tubuh, maka akan tetap sehat, tetapi jika lebih maka akan mengendap di dalam pembuluh darah yang menyebabkan penyempitan dan pengerasan yang dikenal dengan aterosklerosis. Peningkatan kadar kolesterol total berbanding lurus dengan peningkatan risiko PJK. Semakin tinggi kolesterol total, risiko PJK semakin meningkat pula [ANW-04].

Untuk kolesterol total, kadar kolesterol optimal sekitar <200 , untuk kadar kolesterol yang diinginkan 200-239, dan untuk kadar kolesterol yang tinggi bisa mencapai ≥ 240 [GAN-09].

2.1.5. Tekanan darah (Tekanan darah sistolik)

Tekanan darah berpengaruh terhadap hampir semua bagian tubuh, terutama jantung, pembuluh darah, otak, ginjal, dan mata. *Hipertensi* atau tekanan darah tinggi merupakan salah satu factor risiko PJK yang dapat menimbulkan komplikasi berbahaya seperti serangan jantung, *stroke* atau kegagalan ginjal. Peningkatan tekanan darah merupakan beban berat untuk jantung yang menimbulkan trauma langsung terhadap dinding pembuluh darah arteri koronari sehingga memudahkan terjadi aterosklerosis coroner (factor coroner), yaitu penebalan dinding pembuluh darah [ANW-04].

Tekanan darah dibagi menjadi dua yaitu tekanan darah sistolik dan tekanan darah diastolik. Tekanan darah sistolik adalah tekanan saat jantung berdenyut atau berdetak (sistol), sering disebut dengan tekanan darah atas, sedangkan tekanan darah diastolik adalah tekanan saat jantung beristirahat di antara saat pemompaan dan sering disebut dengan tekanan darah bawah. Untuk penyakit jantung coroner, tekanan darah yang dijadikan factor risiko penyebab PJK adalah tekanan darah sistolik karena memiliki factor yang lebih besar untuk menjadi PJK. Tekanan darah sistolik dinyatakan dengan satuan mmHg. Peningkatan tekanan darah sistolik berbanding lurus dengan peningkatan risiko PJK. Semakin tinggi tekanan darah sistolik, risiko PJK semakin meningkat pula [ANW-04].

Menurut petunjuk WHO-ISH klasifikasi hipertensi yaitu pada kelas optimal bila tekanan <120 mmHg, normal bila tekanan <130, tinggi bila tekanan antara 130-139 mmHg, ringan bila tekanan antara 140-159 mmHg, sedang bila tekanan sistolik antara 160-179, dan bila tekanan tinggi ≥ 180 mmHg [ANW-04].

2.2. Logika Fuzzy

2.2.1. Definisi Logika Fuzzy

Logika *fuzzy* merupakan logika yang memiliki nilai kekaburan atau kesamaran (*fuzzyness*), yang digunakan untuk membantu pengambilan keputusan dalam suatu permasalahan. Logika *fuzzy* pertama kali dikenalkan oleh Lotfi A. Zadeh. Logika *fuzzy* digunakan untuk menerjemahkan suatu besaran yang diekspresikan menggunakan bahasa/linguistic [KUH-10].

Kata *fuzzy* merupakan kata sifat yang berarti kabur, tidak jelas. *Fuzziness* atau kekaburan atau ketidakjelasan atau ketidakpastian selalu meliputi keseharian

manusia. Orang yang belum pernah mengenal logika *fuzzy* pasti akan mengira bahwa logika *fuzzy* adalah sesuatu yang rumit dan tidak menyenangkan. Namun, sekali seseorang mulai mengenalnya, pasti akan tertarik untuk ikut mempelajari logika *fuzzy*. Logika *fuzzy* dikatakan sebagai logika baru yang lama, sebab ilmu tentang logika *fuzzy* modern dan metodis baru ditemukan beberapa tahun yang lalu, padahal sebenarnya konsep tentang logika *fuzzy* itu sendiri sudah aja sejak lama [KUH-10].

Logika *fuzzy* memiliki beberapa keunggulan sehingga digunakan untuk menyelesaikan berbagai permasalahan, yaitu [KUH-10]:

1. Mudah dimengerti, karena menggunakan dasar teori himpunan sehingga konsep matematis yang mendasari penalaran *fuzzy* mudah untuk dimengerti;
2. Fleksibel, yaitu mampu beradaptasi dengan perubahan-perubahan dan ketidakpastian pada permasalahan;
3. Memiliki toleransi terhadap data yang tidak tepat, apabila diberikan sekelompok data homogen dengan beberapa data pencilan maka logika *fuzzy* memiliki kemampuan untuk menangani data pencilan tersebut;
4. Dapat memodelkan fungsi-fungsi nonlinier yang sangat kompleks;
5. Logika *fuzzy* didasarkan pada bahasa alami sehingga mudah dimengerti.

2.2.2. Himpunan *fuzzy*

Pada himpunan tegas (*crisp*), nilai keanggotaan suatu item x dalam suatu himpunan A (yang sering ditulis dengan $\mu_A(x)$), terdapat dua kemungkinan, yaitu satu (1) dan nol (0). Nilai satu (1) memiliki arti bahwa suatu item menjadi anggota dalam suatu himpunan. Sedangkan nilai nol (0) memiliki arti bahwa suatu item tidak menjadi anggota dalam suatu himpunan [KUH-10].

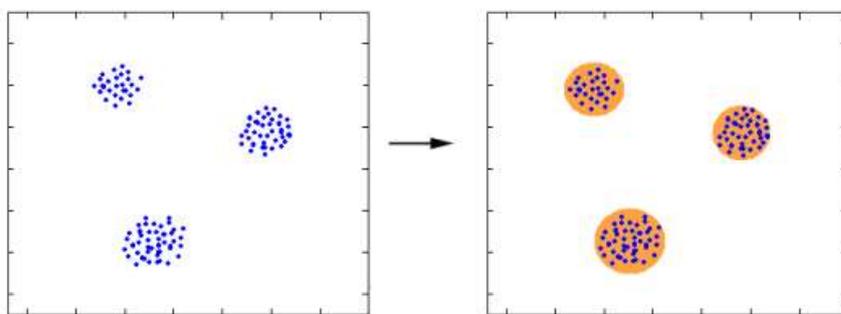
Pemakaian himpunan *crisp* (yang bernilai satu atau nol saja) mengakibatkan perbedaan kategori yang signifikan, walaupun perubahan yang terjadi pada suatu nilai sangat kecil. Himpunan *fuzzy* digunakan untuk mengantisipasi hal tersebut. Suatu item dapat masuk dalam beberapa himpunan yang berbeda. Seberapa besar eksistensi suatu item dalam himpunan dapat dilihat pada nilai keanggotaan. Pada himpunan *fuzzy*, nilai keanggotaan terletak pada rentang 0 sampai dengan 1. Jika x

memiliki nilai keanggotaan *fuzzy* $\mu_A(x) = 0$ berarti x tidak menjadi anggota himpunan A . jika x memiliki nilai keanggotaan *fuzzy* $\mu_A(x) = 1$ berarti x menjadi anggota penuh pada himpunan A [KUH-10].

Himpunan *fuzzy* memiliki 2 atribut yaitu linguistic dan numeris. Linguistic yaitu penamaan suatu grup yang mewakili suatu kondisi tertentu dengan menggunakan bahasa alami, seperti: muda, parobaya, tua. Sedangkan numeris yaitu suatu nilai (angka) yang menunjukkan ukuran dari suatu variable *fuzzy* seperti: 40, 25, 50 [KUH-10].

2.3. Clustering

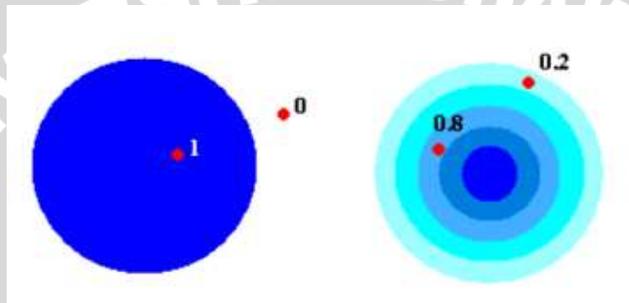
Pengelompokkan (*clustering*) merupakan teknik yang sudah cukup dikenal dan banyak digunakan untuk mengelompokkan data/objek ke dalam kelompok data (*cluster*) sehingga setiap *cluster* memiliki data yang mirip dan berbeda dengan data yang berada dalam *cluster* lain. Jika diberikan himpunan data yang berjumlah terhingga, yaitu x , maka permasalahan *clustering* dalam x adalah mencari beberapa pusat *cluster* yang dapat memberikan ciri kepada masing-masing *cluster* dalam x . Gambar 2.1 menunjukkan contoh sederhana dari proses *clustering*, dimana 3 *cluster* di dalam data dapat teridentifikasi dengan mudah. Kriteria kemiripan yang digunakan dalam kasus ini adalah jarak. Proses ini disebut *distance-based clustering*. Cara lain untuk melakukan *clustering* adalah *conceptual clustering*. Dalam *conceptual clustering*, objek dikelompokkan berdasarkan kecocokannya menurut konsep deskriptif [NUG-11].



Gambar 2.1 Contoh Proses Clustering [NUG-11]

2.3.1. *Hard Clustering* dan *Fuzzy Clustering*

Metode pengelompokan klasik (*hard clustering*) berdasarkan pada teori himpunan klasik, yang menentukan bahwa sebuah objek dapat menjadi anggota atau bukan anggota dari suatu *cluster*. *Fuzzy clustering* memperbolehkan suatu objek untuk menjadi anggota dari beberapa cluster sekaligus dengan derajat keanggotaan yang berbeda-beda. Derajat keanggotaan berada di antara rentang 0 dan 1. Jadi, dataset x dapat dipartisi menjadi c *fuzzy* subset. Di dalam situasi riil, *fuzzy clustering* memiliki hasil yang lebih natural dibandingkan dengan *hard clustering*. Ilustrasi perbedaan antara *hard clustering* dan *fuzzy clustering* ditunjukkan melalui gambar 2.2 [NUG-11].



Gambar 2.2 *Hard Clustering* dan *Fuzzy Clustering* [NUG-11]

2.3.2. *Fuzzy C-Means Clustering*

Fuzzy C-Means (FCM) merupakan salah satu algoritma *fuzzy clustering*. *Fuzzy C-Means* (FCM) adalah suatu teknik peng-*cluster*-an data yang keberadaan setiap titik data dalam suatu *cluster* ditentukan oleh derajat keanggotaan. Teknik ini pertama kali diperkenalkan oleh Jim Bezdek pada tahun 1981 [KUH-10].

Konsep dasar FCM yaitu menentukan pusat cluster, yang akan menandai lokasi rata-rata untuk setiap *cluster*. Dengan cara memperbaiki pusat *cluster* dan derajat keanggotaan setiap titik data secara berulang, maka akan dapat dilihat bahwa pusat *cluster* akan bergerak menuju lokasi yang tepat. Perulangan ini didasarkan pada minimasi fungsi obyektif yang menggambarkan jarak dari titik data yang diberikan ke pusat *cluster* yang terbobot oleh derajat keanggotaan titik data tersebut [KUH-10].

Output dari FCM bukan merupakan *fuzzy inference system*, namun merupakan deretan pusat *cluster* dan beberapa derajat keanggotaan untuk tiap-tiap titik data.

Informasi ini dapat digunakan untuk membangun suatu *fuzzy inference system* [KUH-10].

Algoritma *Fuzzy C-Means clustering* adalah sebagai berikut [KUH-10]:

1. *Input* data yang akan di-*cluster* X, berupa matriks berukuran n x m (n = jumlah sampel data, m = atribut setiap data). X_{ij} = data sampel ke-i (i = 1, 2, 3, ..., n), atribut ke-j (j = 1, 2, 3, ..., m),

2. Menentukan:

Jumlah *cluster* = c;

Pangkat = w;

Maksimum iterasi = MaxIter;

Error terkecil yang diharapkan = ξ ;

Fungsi obyektif awal = $P_0 = 0$;

Iterasi awal = $t = 1$;

3. Membangkitkan bilangan random μ_{ik} , i = 1, 2, ..., n; k = 1, 2, ..., c; sebagai elemen-elemen matriks partisi awal U. kemudian menghitung jumlah setiap kolom dengan persamaan (2.1),

$$Q_i = \sum_{k=1}^c \mu_{ik} \quad (2.1)$$

Dengan j = 1, 2, ..., n. Setelah itu, menghitung derajat keanggotaan awal dengan persamaan (2.2),

$$\mu_{ik} = \frac{\mu_{ik}}{Q_i} \quad (2.2)$$

4. Menghitung pusat *cluster* ke-k (V_{kj}) berdasarkan persamaan (2.3), dengan k = 1, 2, ..., c; dan j = 1, 2, ..., m,

$$V_{kj} = \frac{\sum_{i=1}^n ((\mu_{ik})^w * X_{ij})}{\sum_{ij} (\mu_{ik})^w} \quad (2.3)$$

dengan:

V_{kj} = pusat *cluster* pada *cluster* ke-k dan atribut ke-j

μ_{ik} = derajat keanggotaan untuk data sampel ke-i pada *cluster* ke-k

X_{ij} = data ke-i, atribut ke-j

w = pembobot

5. Menghitung fungsi obyektif pada iterasi ke t (P_t) sesuai persamaan (2.4),

$$P_t = \sum_{i=1}^n \sum_{k=1}^c \left(\sum_{j=1}^m (X_{ij} - V_{kj})^2 \right) (\mu_{ik})^w \quad (2.4)$$

dengan:

V_{kj} = pusat *cluster* pada *cluster* ke-k dan atribut ke-j

μ_{ik} = derajat keanggotaan untuk data sampel ke-i pada *cluster* ke-k

X_{ij} = data ke-i, atribut ke-j

P_t = fungsi obyektif pada iterasi ke-t

6. Menghitung perubahan matriks partisi berdasarkan persamaan (2.5),

$$\mu_{ik} = \frac{\left[\sum_{j=1}^m (X_{ik} - V_{kj})^2 \right]^{-1}}{\sum_{k=1}^c \left[\sum_{j=1}^m (X_{ik} - V_{kj})^2 \right]^{-1}} \quad (2.5)$$

dengan:

V_{kj} = pusat *cluster* pada *cluster* ke-k dan atribut ke-j

μ_{ik} = derajat keanggotaan untuk data sampel ke-i pada *cluster* ke-k

X_{ij} = data ke-i, atribut ke-j

7. Memeriksa kondisi berhenti:

Jika: $(|P_t - P_{t-1}| < \xi)$ atau $(t > \text{MaxIter})$ maka berhenti;

Jika tidak: $t = t + 1$, ulangi langkah ke-4.

8. Jika iterasi berhenti, ditentukan cluster dari tiap-tiap data. *Cluster* dipilih berdasarkan nilai matriks partisi terbesar.

2.3.3. Fuzzy Subtractive Clustering

Dasar dari metode *Fuzzy Subtractive Clustering* adalah ukuran densitas (potensi) titik-titik data dalam suatu ruang (variable). Konsep dasar dari metode *Fuzzy Subtractive Clustering* adalah menentukan daerah-daerah dalam suatu variable yang memiliki densitas tinggi terhadap titik-titik di sekitarnya. Titik dengan jumlah tetangga terbanyak akan dipilih untuk menjadi pusat kelompok. Titik yang sudah dipilih menjadi pusat kelompok ini kemudian akan dikurangi densitasnya. Selanjutnya akan dipilih titik lain yang menjadi tetangga terbanyak untuk dijadikan pusat kelompok yang lain. Hal ini akan dilakukan berulang-ulang sampai semua titik teruji. Metode *Fuzzy Subtractive Clustering* tergolong metode *unsupervised clustering* dimana jumlah pusat *cluster* tidak diketahui. Metode ini menggunakan data sebagai kandidat dari pusat *cluster*, sehingga beban komputasi tergantung dari jumlah data dan tidak bergantung dari dimensi data. Jumlah pusat *cluster* yang dicari ditentukan melalui proses iterasi untuk mencari titik-titik dengan jumlah tetangga terbanyak [NUG-11].

Algoritma *Fuzzy Subtractive Clustering* adalah sebagai berikut [KUH-10]:

1. Menentukan:

Matriks X yang merupakan data yang akan di-*cluster*, berukuran $i \times j$, dengan i = jumlah data yang akan di-*cluster*; dan j = jumlah variable/atribut (kriteria),

$$\begin{bmatrix} X_{11} & X_{12} & \dots & X_{1j} \\ X_{21} & X_{22} & & X_{2j} \\ \vdots & \vdots & \ddots & \vdots \\ X_{i1} & X_{i2} & \dots & X_{ij} \end{bmatrix}$$

2. Menentukan:

- a. r_j (jari-jari setiap atribut data); $j = 1, 2, \dots, m$
- b. Q (squash factor)
- c. Accept ratio
- d. Reject ratio
- e. $X_{\min j}$ (min data yang diperbolehkan dalam setiap atribut data); $j = 1, 2, \dots, m$
- f. $X_{\max j}$ (max data yang diperbolehkan dalam setiap atribut data); $j = 1, 2, \dots, m$

3. Normalisasi dapat dilakukan menggunakan persamaan (2.6),

$$X_{ij} = \frac{X_{ij} - X_{\min j}}{X_{\max j} - X_{\min j}}, \quad i=1,2,\dots,n; j=1,2,\dots,m \quad (2.6)$$

dimana:

X_{ij} = data ke-I atribut ke-j

$X_{\min j}$ = minimum data pada atribut ke-j

$X_{\max j}$ = maximum data pada atribut ke-j

4. Tentukan potensi awal tiap-tiap titik data

a. $i=1$; i adalah data ke

b. kerjakan hingga $i=n$

i. $T_j = X_{ij}$ (T_j merupakan data ke- i atribut ke- j); $j=1,2,\dots,m$

ii. Hitung menggunakan persamaan (2.7),

$$\text{Dist}_{kj} = \left(\frac{T_j - X_{kj}}{r_j} \right) \quad (2.7)$$

iii. Hitung potensi awal

Jika $m = 1$, maka hitung potensi awal menggunakan persamaan (2.8),

$$D_i = \sum_{k=1}^n e^{-4(\text{Dist}_{ki}^2)} \quad (2.8)$$

Jika $m > 1$, maka hitung potensi awal menggunakan persamaan (2.9),

$$D_i = \sum_{k=1}^n e^{-4 \left(\sum_{j=1}^m \text{Dist}_{kj}^2 \right)} \quad (2.9)$$

iv. $i = i + 1$

5. cari titik dengan potensi tertinggi

a. $M = \max[D_i | i=1,2,\dots,n]$;

b. $h = i$, sedemikian hingga $D_i = M$;

6. tentukan pusat *cluster* dan kurangi potensinya terhadap titik-titik di sekitarnya.

a. Center = []

b. $V_j = X_{ij}; j=1,2,\dots,m;$

c. C = 0 (jumlah *cluster*);

d. Kondisi = 1;

e. Z = M;

f. Kerjakan jika (Kondisi $\neq 0$) & (Z $\neq 0$):

i. Kondisi = 0 (sudah tidak ada calon pusat baru lagi);

ii. Rasio = Z/M.

iii. Jika rasio > *accept_rasio*, maka Kondisi = 1; (ada calon pusat baru)

iv. Jika tidak,

- Jika Rasio > *reject_ratio*, (calon baru akan diterima sebagai pusat jika keberadaannya akan memberikan keseimbangan terhadap data-data yang letaknya cukup jauh dengan pusat *cluster* yang telah ada), maka kerjakan

- Md = -1;

- Kerjakan untuk i = 1 sampai i = C, menggunakan persamaan (2.10) dan persamaan (2.11):

$$i. G_{ij} \frac{V_j - \text{Center}_{ij}}{r}, j=1,2,\dots,m \quad (2.10)$$

$$ii. Sd_i = \sum_{j=1}^m (G_{ij})^2 \quad (2.11)$$

iii. Jika (Md < 0) atau (Sd < Md), maka Md + Sd;

- $Smd = \sqrt{Md}; \quad (2.12)$

- Jika (Rasio + Smd) ≥ 1 , maka Kondisi = 1; (Data diterima sebagai pusat *cluster*)

- Jika (Rasio + Smd) < 1, maka Kondisi = 2; (Data tidak akan dipertimbangkan kembali sebagai pusat *cluster*).

v. Jika Kondisi = 1 (calon pusat baru diterima sebagai pusat baru), kerjakan menggunakan persamaan (2.13),

- $C = C + 1$; (2.13)

- $Center_c = V$;

- Kurangi potensi dari titik-titik di dekat pusat *cluster*:

- $S_{ij} \frac{V_j - X_{ij}}{r_j * q}$; $j=1,2,\dots,m$; $i=1,2,\dots,n$ (2.14)

- $De_i = M * e^{-4 \left(\sum_{j=1}^m (s_{ij})^2 \right)}$ (2.15)

- $D = D - Dc$; (2.16)

- iv. Jika $D_i \leq 0$, maka $D_i = 0$; $i=1,2,\dots,n$

- v. $Z = \max [D_i | i=1,2,\dots,n]$;

- vi. Pilih $h = i$, sedemikian hingga $D_i = Z$;

- Jika Kondisi = 2 (calon pusat baru tidak diterima sebagai pusat baru), maka:

- i. $D_h = 0$;

- ii. $Z = \max [D_i | i=1,2,\dots,n]$;

- iii. Pilih $h = i$, sedemikian hingga $D_i = Z$;

7. Kembalikan pusat *cluster* dari bentuk ternormalisasi ke bentuk semula, menggunakan persamaan (2.17),

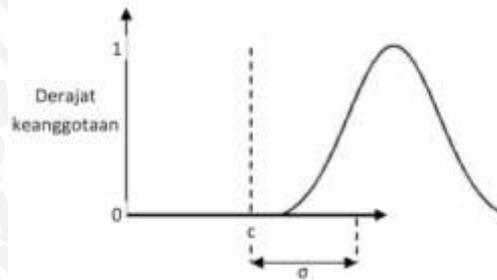
$$Center_{ij} = Center_{ij} * (X \max_j - X \min_j) + X \min_j ; \quad (2.17)$$

8. Hitung nilai sigma *cluster*, menggunakan persamaan (2.18),

$$\sigma_j = r_j * (X \max_j - X \min_j) / \sqrt{8} \quad (2.18)$$

Untuk mempersingkat notasi, selanjutnya Center akan disebut sebagai C.

hasil dari algoritma *Subtractive clustering* ini berupa matriks pusat *cluster* (C) dan sigma (σ) akan digunakan untuk menentukan nilai parameter fungsi keanggotaan *Gauss* seperti terlihat pada Gambar 2.3.



Gambar 2.3 Kurva Gauss

Dengan kurva Gauss tersebut, maka derajat keanggotaan suatu titik data X_i pada *cluster* ke- k , adalah:

$$\mu_{ki} = e^{-\sum_{j=1}^m \frac{(X_{ij} - C_{kj})^2}{2\sigma_j^2}} \quad (2.19)$$

2.3.4. Indeks Validasi

Sejak konsep himpunan *fuzzy* diperkenalkan oleh Zadeh pada tahun 1965, konsep *fuzzy clustering* banyak dikembangkan dan diterapkan. Dalam konsep *fuzzy clustering*, suatu data dapat menjadi anggota dari beberapa *cluster* sekaligus menurut derajat keanggotaannya [WUY-05]. Proses *clustering* dalam algoritma *fuzzy clustering* selalu mencari solusi terbaik untuk parameter yang telah didefinisikan. Namun solusi terbaik ini belum tentu dapat menentukan deskripsi terbaik dari struktur data. Dalam hal ini jumlah *cluster* yang ditentukan atau bentuk *cluster* mungkin tidak sesuai dengan data. Untuk menentukan jumlah *cluster* yang paling optimal dan dapat memvalidasi apakah partisi *fuzzy* yang diterapkan dalam proses *clustering* sesuai dengan data, digunakan indeks pengukuran validitas *cluster*. Berikut ini adalah beberapa metode pengukuran validitas *cluster* yang umumnya digunakan untuk algoritma *fuzzy clustering* [WUY-05].

1. Partition Coefficient

Partition Coefficient (PC) merupakan metode yang mengukur jumlah *cluster* yang mengalami *overlap*. Indeks PC mengukur validitas *cluster* dapat dihitung menggunakan persamaan (2.19).

$$PC(c) = \frac{1}{n} \sum_{i=1}^c \sum_{j=1}^n (\mu_{ij}^2) \quad (2.20)$$

dimana:

c = jumlah *cluster*

n = jumlah data

μ_{ij} = derajat keanggotaan data ke- j pada *cluster* ke- i

$PC(c)$ = nilai indeks PC pada *cluster* ke- c

Nilai PC berada dalam batas $\frac{1}{c} \leq PC(c) \leq 1$. Pada umumnya jumlah *cluster* yang paling optimal ditentukan dari nilai PC yang paling besar

$$(\max_{2 \leq c \leq n-1} PC(c)).$$

2. Modified Partition Coefficient

Partition Coefficient cenderung mengalami perubahan yang monoton terhadap beragam nilai c (jumlah *cluster*). Modifikasi dari indeks PC (*Modified Partition Coefficient*/MPC) dapat mengurangi perubahan yang monoton tersebut dan didefinisikan dalam persamaan (2.20).

$$MPC(c) = 1 - \frac{c}{c-1} (1 - PC(c)) \quad (2.21)$$

dimana:

c = jumlah *cluster*

$MPC(c)$ = nilai indeks MPC pada *cluster* ke- c

Nilai MPC berada dalam batas $0 \leq MPC(c) \leq 1$. Pada umumnya jumlah *cluster* yang optimal ditentukan dari nilai MPC yang paling besar

$$(\max_{2 \leq c \leq n-1} MPC(c)).$$

3. Clasification Entropy

Clasification Entropy (CE) merupakan metode yang mengukur tingkat keaburan (*fuzziness*) dari partisi *cluster*. Indeks CE mengukur validitas *cluster* menggunakan persamaan (2.21).

$$CE(c) = -\frac{1}{n} \sum_{i=1}^c \sum_{j=1}^n \mu_{ij} \log(\mu_{ij}) \quad (2.22)$$

dimana:

c = jumlah *cluster*

n = jumlah data

μ_{ij} = derajat keanggotaan data ke- j pada *cluster* ke- i

$CE(c)$ = nilai indeks CE pada *cluster* ke- c

Nilai CE berada dalam batas $0 \leq CE(c) \leq \log_2 c$. Pada umumnya jumlah *cluster* yang optimal ditentukan dari nilai PE yang paling kecil

$$\left(\min_{2 \leq c \leq n-1} CE(c) \right).$$

Semua indeks yang dihitung dapat memberikan alternative untuk memilih metode optimum dengan mengkombinasikan indeks validitas masing-masing metode yang kemudian dapat dipilih metode optimumnya pada saat indeks tersebut berkombinasi paling banyak. Langkahnya adalah menghitung ketiga indeks validitas, lalu memberi ranking tiap metode yang mungkin pada masing-masing indeks. Metode optimum diperoleh pada rata-rata ranking yang paling tinggi.



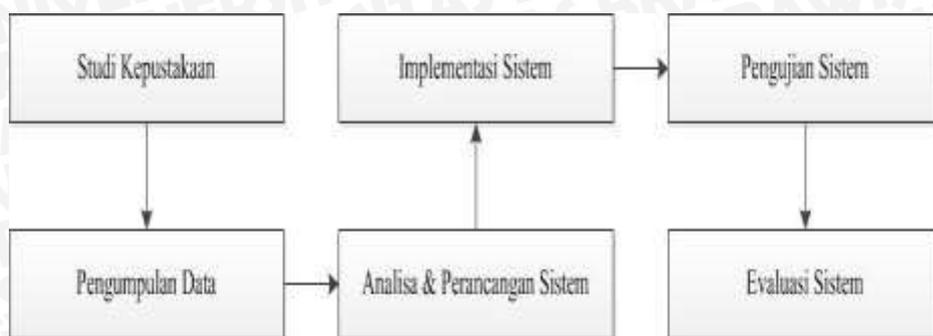
BAB III

METODOLOGI DAN PERANCANGAN SISTEM

Alur penelitian pada penelitian ini melalui beberapa tahap yaitu:

1. Studi kepustakaan
Studi kepustakaan atau literature dilakukan untuk mendukung penelitian serta meningkatkan pemahaman terhadap permasalahan yang diangkat serta penyelesaian dari permasalahan tersebut.
2. Pengumpulan data
Data yang diambil untuk bahan dalam penelitian ini adalah data rekam medic pasien PJK.
3. Analisis dan perancangan sistem
Analisis dan perancangan sistem ini meliputi perancangan sistem, tabel data, antarmuka, dan *clustering*.
4. Implementasi sistem
Pada tahap ini, dibangun sebuah aplikasi untuk membandingkan dua metode dalam masalah risiko PJK sebagai wujud implementasi untuk menyelesaikan masalah sesuai dengan perancangan sistem.
5. Pengujian sistem
Pada tahap ini, dilakukan *clustering* pada data latih dengan menggunakan dua metode serta metode validitas untuk menguji tingkat akurasi dalam permasalahan tersebut.
6. Evaluasi sistem
Hasil dari *clustering* tersebut dianalisis dan dievaluasi untuk mengetahui akurasi sistem dalam membandingkan dua metode untuk PJK

Bagan alur penelitian pada perbandingan *Fuzzy C-Means Clustering* dan *Fuzzy Subtractive Clustering* pada kasus risiko PJK dapat dilihat pada Gambar (3.1)



Gambar 3.1 Diagram Alur Penelitian

3.1. Studi Kepustakaan

Pada tahap ini, dilakukan studi kepustakaan atau literature yang dapat mendukung penelitian serta meningkatkan pemahaman terhadap permasalahan yang diangkat. Pengetahuan diambil dari berbagai sumber, seperti buku, jurnal, serta sumber lain yang dinilai dapat memberi tambahan wawasan untuk penelitian ini.

3.2. Data Penelitian

Dataset PJK yang digunakan merupakan data dari penelitian tentang PJK yang dilakukan oleh Wahyuni (2011), yang diperoleh dari data rekam medic pasien RSU Saiful Anwar, RSI, Unisma Malang dan Laboratorium cek fisik kesehatan. Selain itu, digunakan juga data dari penelitian Effendi (2008) yang merupakan data catatan rekam medis penderita PJK yang dirawat di Instalasi Rawat Inap Unit Penyakit Dalam RSUP Dr. Sardjito Yogyakarta.

Dataset PJK merupakan dataset pasien positif PJK yang terdiri dari lima factor risiko PJK, meliputi umur, LDL, kolesterol total, HDL, dan tekanan darah sistolik.

3.3. Analisis dan Perancangan Sistem

3.3.1. Deskripsi umum sistem

Sistem memiliki 2 proses utama, yaitu proses *clustering* dan proses perbandingan dengan sistem *fuzzy*. Pada proses *clustering*, dibutuhkan *input* dataset rekam medic pasien PJK untuk dilakukan proses *clustering* menggunakan dua metode. Sedangkan untuk proses perbandingan dibutuhkan hasil dari proses *clustering*.

1. Proses *clustering* dan perbandingan

Tujuan proses ini adalah mengelompokkan data pasien PJK melalui perhitungan menurut algoritma pada masing-masing kedua metode. Selain itu juga membandingkan kedua data tersebut dari derajat keanggotaan yang telah diperoleh menggunakan ukuran validitas *cluster*.

Proses *clustering* dan perbandingan yang dilakukan pengguna yaitu:

- a. Pengguna menentukan parameter pada kedua metode untuk menentukan hasil *clustering*
- b. Pengguna menyimpan hasil *clustering* berupa data yang telah di *cluster*
- c. Pengguna menampilkan hasil *clustering* untuk dilakukan proses perhitungan validitas *cluster*
- d. Pengguna mengetahui validitas *cluster* dari masing-masing metode, serta hasil perbandingan dari kedua metode.

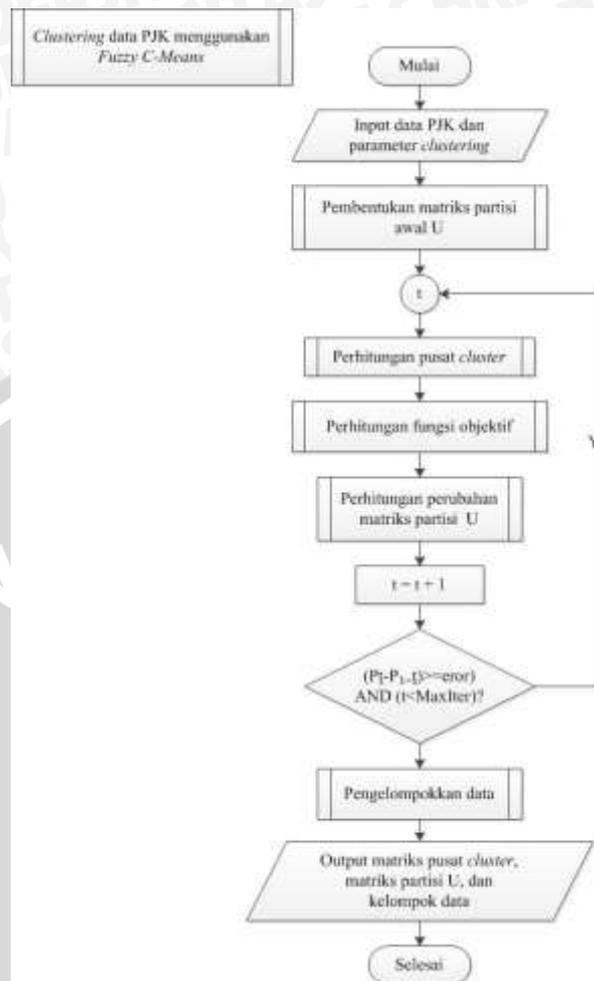
3.3.2. Perancangan Proses

Proses *clustering* dataset PJK ini dilakukan menggunakan dua metode, yaitu *Fuzzy C-Means Clustering* dan *Fuzzy Subtractive Clustering*.

A. Proses *Clustering* menggunakan *Fuzzy C-Means Clustering* (FCM)

Alur proses *clustering* data PJK menggunakan *Fuzzy C-Means Clustering* terdiri atas 5 subproses, yaitu pembentukan matriks awal U , perhitungan pusat *cluster*, perhitungan fungsi obyektif, perhitungan perubahan matriks partisi awal U , dan pengelompokan data. *Input* proses *clustering* berupa data PJK 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 latih pada semua *cluster* (μ_{ik}), dan kelompok data berisi hasil dari pengelompokan data latih berdasarkan derajat keanggotaan data terhadap suatu *cluster*.

Proses *Clustering* menggunakan *Fuzzy C-Means Clustering* (FCM) ditunjukkan pada Gambar (3.2).



Gambar 3.2 Alur proses *clustering* menggunakan FCM

Rincian alur proses *clustering* data dengan algoritma *Fuzzy C-Means Clustering* (FCM) adalah sebagai berikut:

1. Pembentuk matriks partisi awal U

Hal ini dilakukan dengan cara membangkitkan bilangan secara random kemudian melakukan perhitungan dengan persamaan (2.1 dan 2.2)

2. Selama memenuhi kondisi dimana nilai fungsi obyektif lebih besar sama dengan kesalahan minimum yang diharapkan ($P_t \geq \xi$) atau iterasi lebih kecil sama dengan iterasi maksimum ($t \leq \text{maxIter}$), maka dilakukan proses berikut secara berulang.
 - a. Menghitung pusat *cluster* semua atribut pada setiap *cluster* dengan menggunakan persamaan (2.3).

- b. Menghitung nilai fungsi obyektif (P_t) dengan menggunakan persamaan (2.4).
- c. Menghitung perubahan matriks partisi U dengan menggunakan persamaan (2.10).

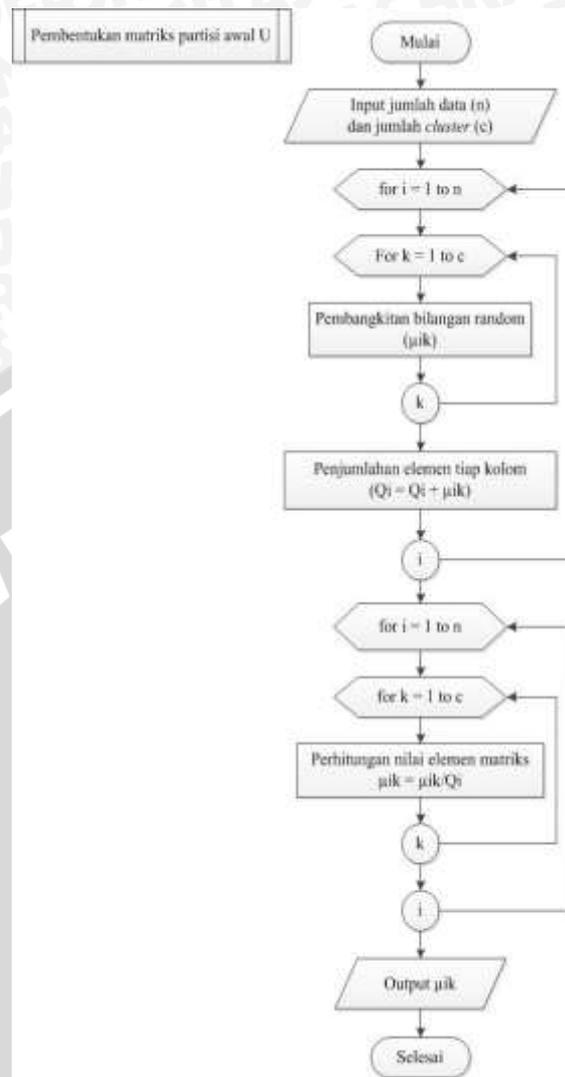
Nilai kesalahan (*error*) dan \maxIter berupa variable yang ditentukan oleh pengguna. Apabila terdapat suatu kondisi, dimana nilai selisih fungsi obyektif iterasi ke- t dengan $t-1$ lebih kecil dari kesalahan minimum (ξ) atau iterasi (t) sudah mencapai iterasi maksimum ($(P_t - P_{t-1} < \xi)$ atau $t \geq \maxIter$), maka proses dilanjutkan ke langkah-3.

3. Pengelompokkan data ke *cluster* yang sesuai. Pengelompokkan ini didasarkan pada kecenderungan derajat keanggotaan data (μ_{ik}). Data akan dimasukkan ke *cluster* dengan nilai derajat keanggotaan yang paling tinggi.

1. Proses pembentukan matriks partisi awal

Proses pembentukan matriks partisi awal ditunjukkan pada Gambar (3.2).

Proses pembentukan matriks partisi awal U tidak melibatkan data PJK. *Input* proses ini hanya memerlukan parameter jumlah data PJK (n) dan jumlah *cluster* yang akan dibentuk (c). *Output* proses ini adalah matriks derajat keanggotaan data terhadap *cluster* (μ_{ik}), dengan dimensi jumlah data x jumlah *cluster* ($n \times c$).



Gambar 3.3 Alur proses pembentukan matriks partisi awal

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

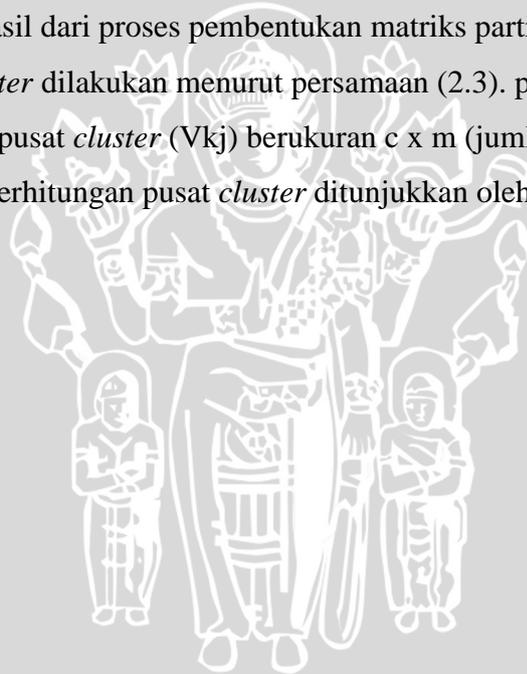
1. Pembangkitan bilangan random
 Bilangan random yang dibangkitkan berguna untuk merepresentasikan derajat keanggotaan suatu data ke- i ke dalam *cluster* ke- k (μ_{ik}). Rentang bilangan random yang ditentukan yaitu bilangan antara 0 sampai 1.
2. Penjumlahan elemen setiap kolom
 Proses ini menjumlah bilangan random μ_{ik} setiap iterasi ke- i ($Q_i = \mu_{i1} + \mu_{i2} + \mu_{i3} + \dots + \mu_{ik}$) sehingga didapatkan Q_i satu dimensi berukuran n .

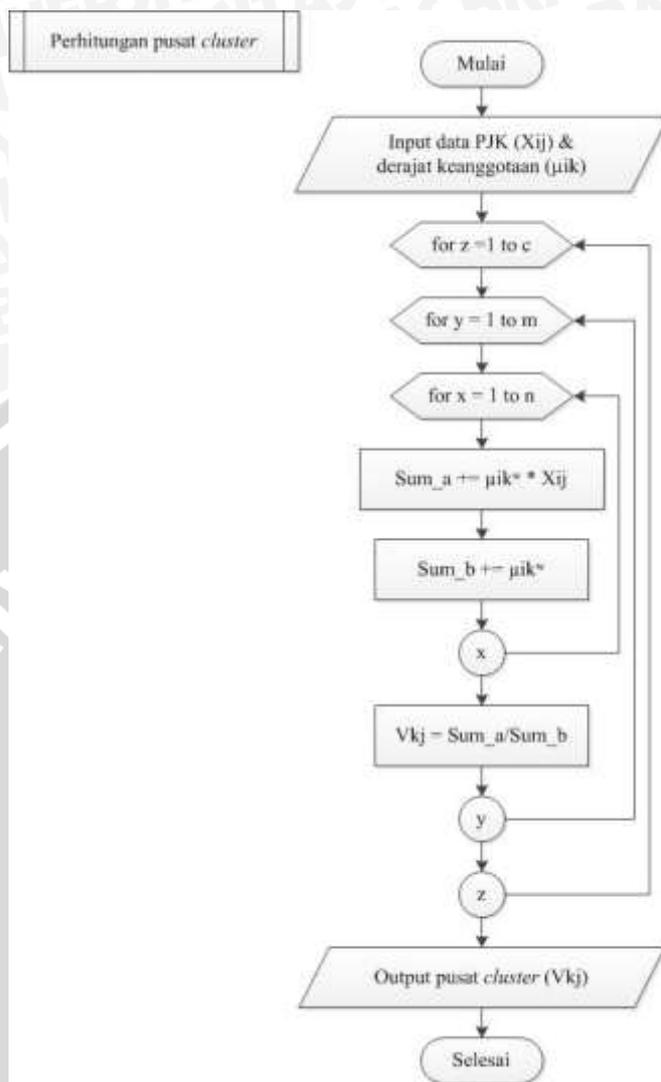
3. Perhitungan nilai elemen matriks

Setelah perhitungan nilai Q_i , dilakukan perhitungan dengan persamaan (2.1) untuk memperbaiki nilai elemen matriks (μ_{ik}) sehingga dihasilkan menghasilkan derajat keanggotaan (μ_{ik}). Jika perhitungan benar, maka 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 perhitungan derajat keanggotaan μ_{ik} .

2. Proses perhitungan pusat *cluster*

Proses perhitungan pusat *cluster* memerlukan *input* berupa dataset PJK (X_{ij}) dan derajat keanggotaan hasil dari proses pembentukan matriks partisi awal $U (\mu_{ik})$. 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 perhitungan pusat *cluster* ditunjukkan oleh Gambar (3.3).

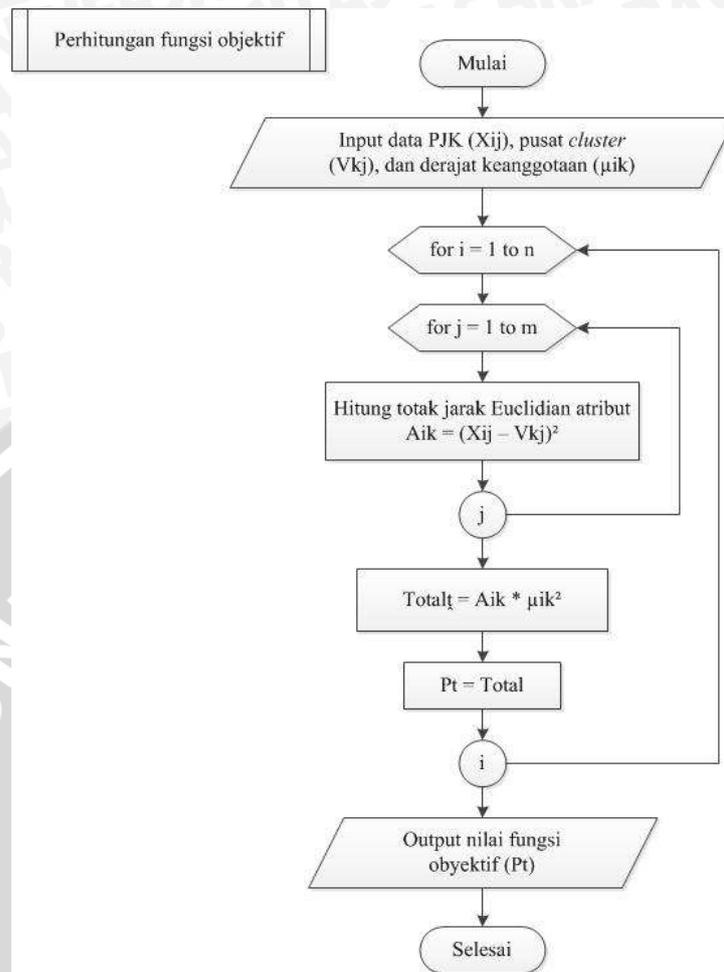




Gambar 3.4 Alur proses perhitungan pusat cluster

3. Proses perhitungan fungsi obyektif

Proses perhitungan fungsi obyektif ditunjukkan oleh Gambar (3.4).



Gambar 3.5 Alur proses perhitungan fungsi obyektif

Proses perhitungan fungsi obyektif melibatkan *input* data PJK (X_{ij}), pusat *cluster* (V_{kj}), dan derajat keanggotaan (μ_{ik}). Proses ini akan menghasilkan *output* berupa nilai fungsi obyektif iterasi ke- t (P_t) sesuai dengan alur diagram *clustering* data pada Gambar (3.2), perhitungan fungsi obyektif dilakukan dengan menggunakan persamaan (2.4).

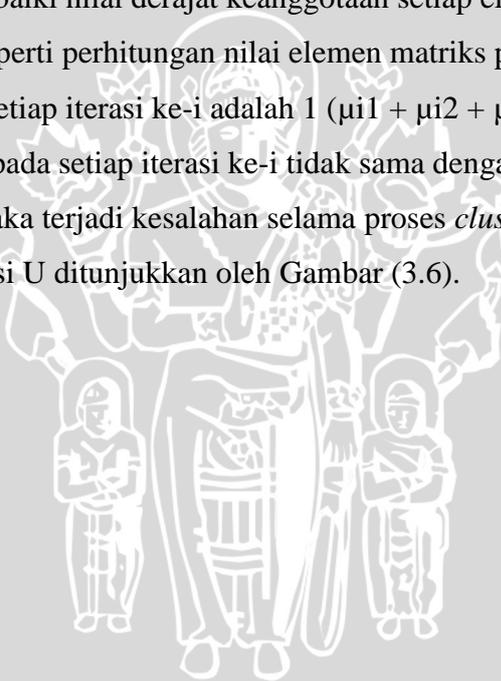
Rincian alur proses perhitungan fungsi obyektif adalah sebagai berikut. Pada setiap iterasi ke- i dilakukan k sebanyak c (jumlah *cluster* yang akan dibentuk) untuk melakukan perhitungan total jarak atribut ($A_{ij} \times (X_{i1} - V_{k1})^2 + (X_{i2} - V_{k2})^2 + (X_{i3} - V_{k3})^2 + \dots + (X_{ij} - V_{kj})^2$). Setelah itu, nilai A_{ij} dikalikan dengan derajat keanggotaan μ_{ik} kemudian ditotal untuk tiap iterasi i . Pada akhirnya, semua total perhitungan akan dijumlah sehingga menghasilkan satu nilai fungsi obyektif (P_t). Selisih nilai P_t dengan nilai P_{t-1} dihitung kemudian

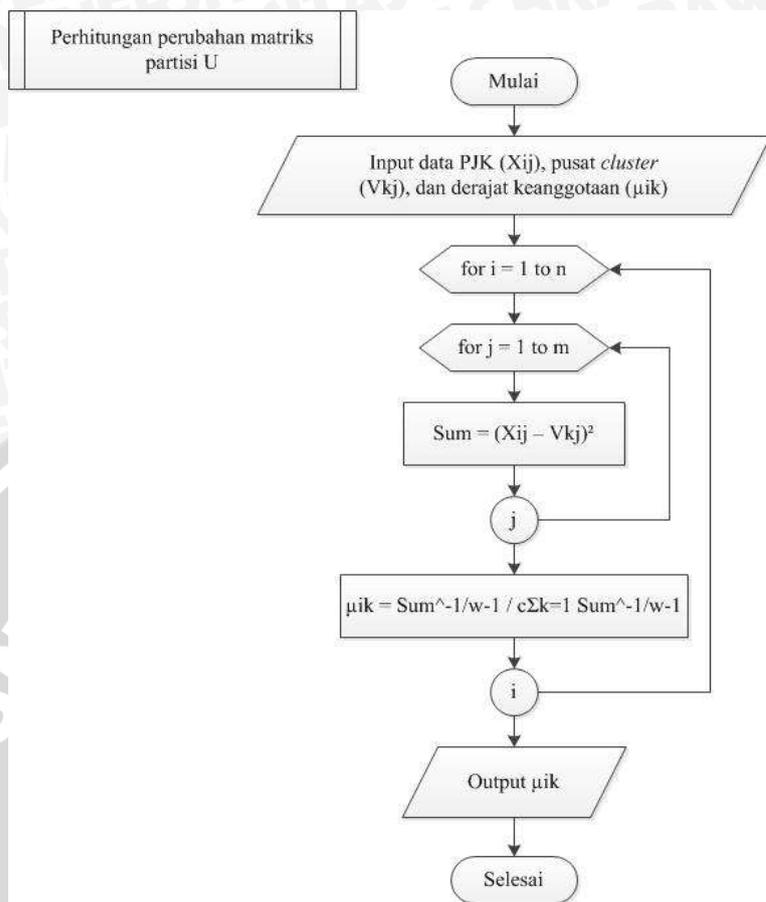
dibandingkan dengan nilai kesalahan minimum (ξ) yang telah ditentukan untuk pemeriksaan kondisi berhenti pada proses *clustering* data. Jika $|P_t - P_{t-1}| < \xi$ maka perulangan pada proses *clustering* dihentikan.

4. Proses perubahan matriks partisi U

Seperti pada perhitungan fungsi obyektif, proses perubahan matriks partisi U memiliki *input* berupa data PJK (X_{ij}), pusat *cluster* (V_{kj}), dan derajat keanggotaan (μ_{ik}). *Output* proses ini adalah perbaikan derajat keanggotaan μ_{ik} yang merupakan hasil perubahan matriks U.

Perubahan matriks partisi U dapat dihitung dengan menggunakan persamaan (2.5) sehingga dapat memperbaiki nilai derajat keanggotaan setiap elemen matriks partisi U (μ_{ik}). 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. Proses perubahan matriks partisi U ditunjukkan oleh Gambar (3.6).

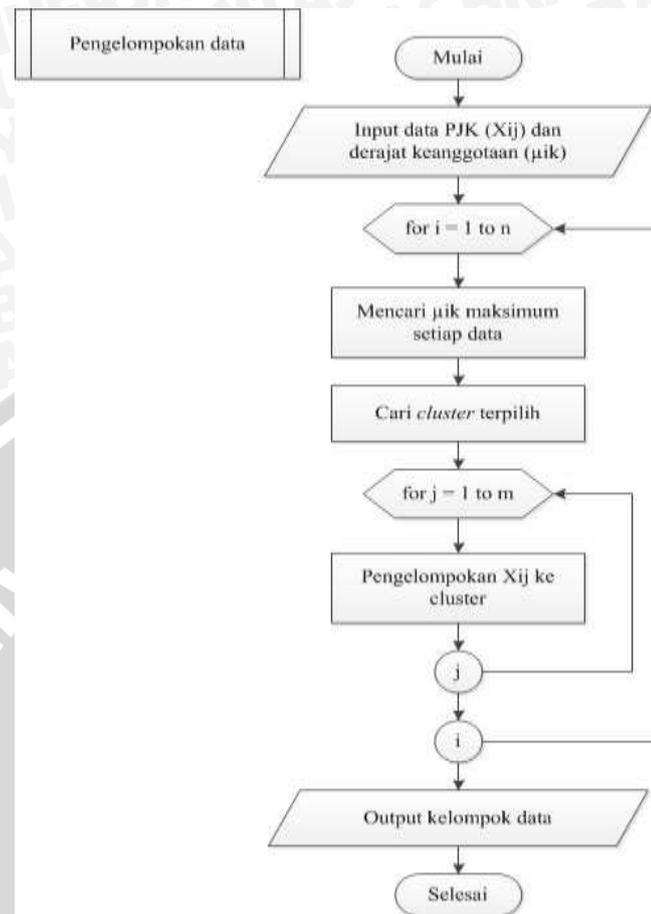




Gambar 3.6 Alur proses perubahan matriks partisi

5. Proses pengelompokkan data

Proses pengelompokkan data ditunjukkan pada Gambar (3.7).



Gambar 3.7 Alur proses pengelompokan data FCM

Proses pengelompokan data melibatkan *input* berupa data PJK (X_{ij}) dan derajat keanggotaan (μ_{ik}). Proses ini menghasilkan *output* berupa kelompok data B_{ij} yang menyatakan *cluster* dari setiap data X_{ij} . Rincian alur proses pengelompokan data adalah sebagai berikut:

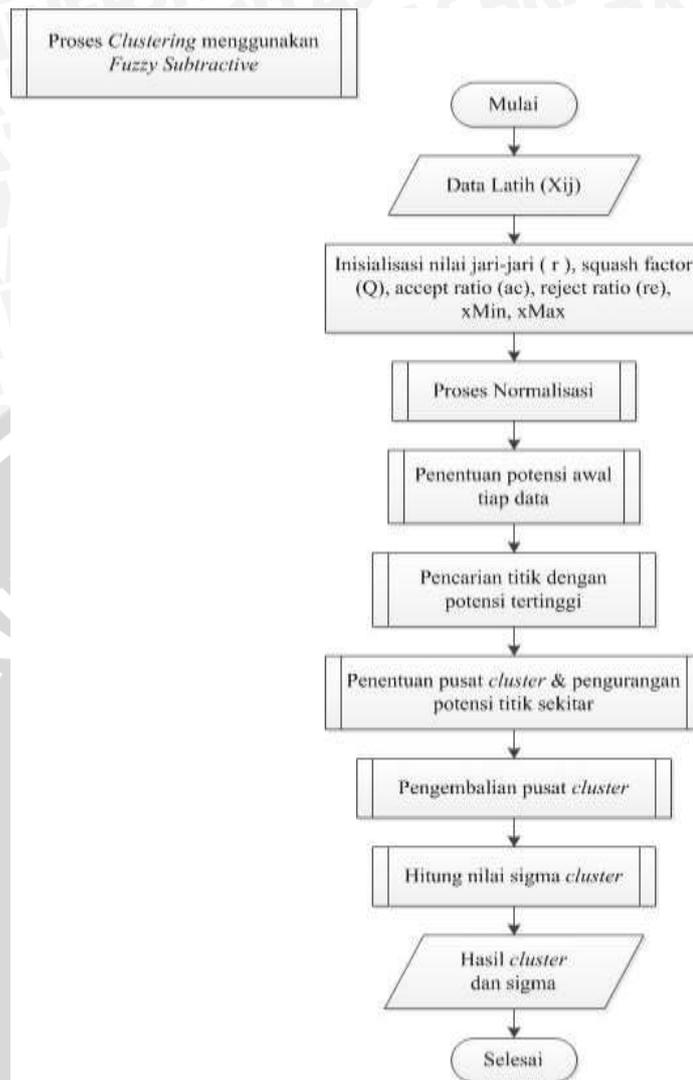
1. Mencari μ_{ik} maksimum setiap data, yaitu dengan cara membandingkan derajat keanggotaan μ_{ik} yang dipilih yaitu nilai yang paling mendekati 1.
2. Mencari *cluster* terpilih (k_p), yaitu dengan memilih *cluster* k yang memiliki nilai μ_{ik} maksimum.
3. Pengelompokan data ke *cluster*, yaitu mengelompokkan data ke *cluster* berdasarkan nilai k_p .

B. Proses *Clustering* menggunakan *Fuzzy Subtractive Clustering*

Pada proses *clustering* ini, data latih yang jumlahnya berbeda-beda yaitu 30 data, 40 data, dan 50 data diproses untuk didapatkan *cluster-cluster* data. Data latih yang merupakan dataset PJK, terdiri dari factor risiko PJK. Rangkaian proses *clustering* ini ditunjukkan pada Gambar (3.8).

Penjabaran detail langkah-langkah proses *clustering* adalah sebagai berikut:

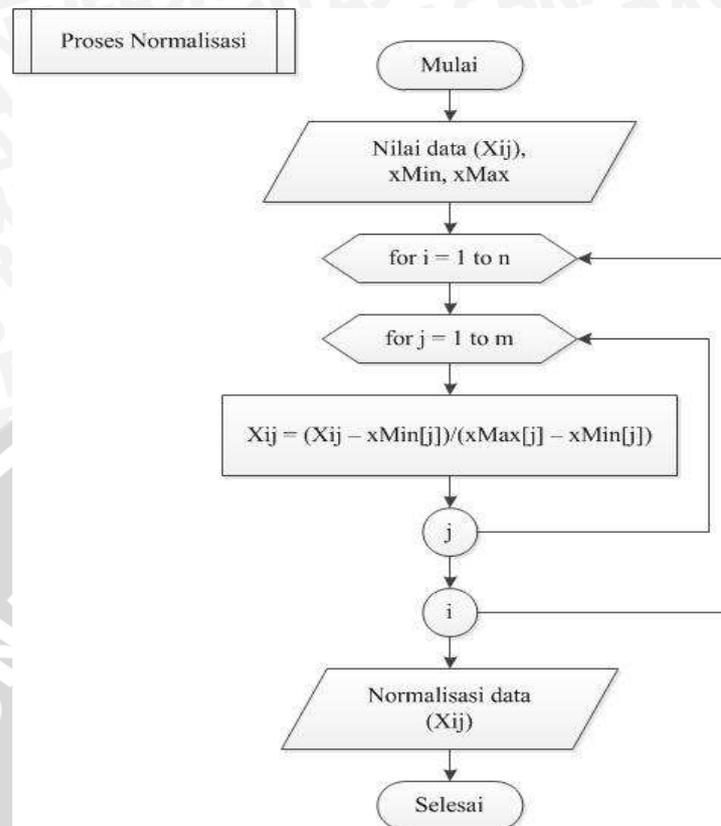
1. Memasukkan dataset PJK yang merupakan data latih yang akan dilakukan proses *clustering*.
2. Menentukan nilai dari variable jari-jari setiap atribut data (r), nilai *squash factor* (q), nilai *accept_ratio*, nilai *reject_ratio*, nilai minimum data diperbolehkan (X_{Min}) dan nilai maksimum data diperbolehkan (x_{Max}).
3. Melakukan normalisasi terhadap data masukan.
4. Menentukan potensi awal tiap-tiap data.
5. Mencari titik dengan potensi tertinggi.
6. Menentukan pusat *cluster* dan mengurangi potensinya terhadap titik-titik di sekitarnya.
7. Mengembalikan pusat *cluster* dari bentuk ternormalisasi ke bentuk semula.
8. Menghitung nilai sigma *cluster* yang nantinya digunakan untuk mengetahui derajat keanggotaan suatu titik data.



Gambar 3.8 Alur proses *clustering* menggunakan *Subtractive Clustering*

1. Proses Normalisasi

Proses normalisasi merupakan langkah awal dari proses *clustering*. Alur proses dari proses ini ditunjukkan pada Gambar (3.9).



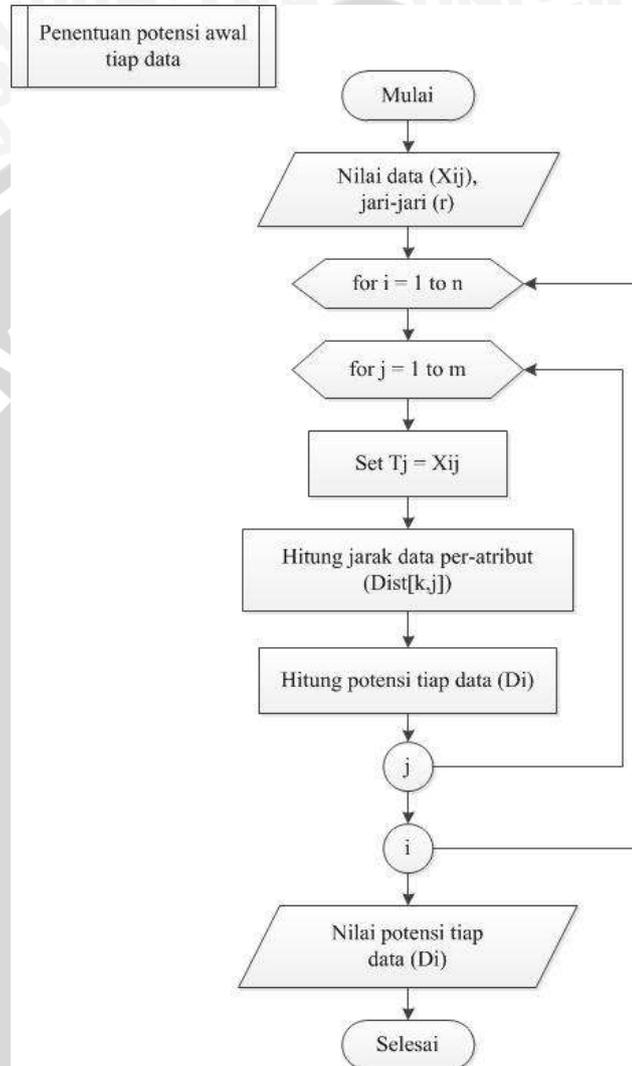
Gambar 3.9 Alur proses normalisasi

Dari Gambar (3.9), penjabaran dari langkah-langkah proses normalisasi adalah sebagai berikut:

1. Persiapan data input (x), nilai $xMin$, nilai $xMax$.
2. Inisialisasi variable m untuk jumlah atribut dataset, untuk jumlah data input n dan nilai $i=1$.
3. Untuk $i=1$ sampai n , dilakukan:
 - Untuk $j=1$ sampai m , dilakukan proses perhitungan nilai x dengan persamaan (2.6).
4. Jika $i > n$, maka proses normalisasi berhenti dan dihasilkan data input yang telah dinormalisasi (x).

2. Proses Penentuan Potensi Awal Tiap Data

Proses penentuan potensi awal merupakan proses untuk mencari nilai potensi masing-masing data yang akan dijadikan kandidat pusat *cluster*. Alur proses ini ditunjukkan pada Gambar (3.10).



Gambar 3.10 Alur proses penentuan potensi awal tiap data

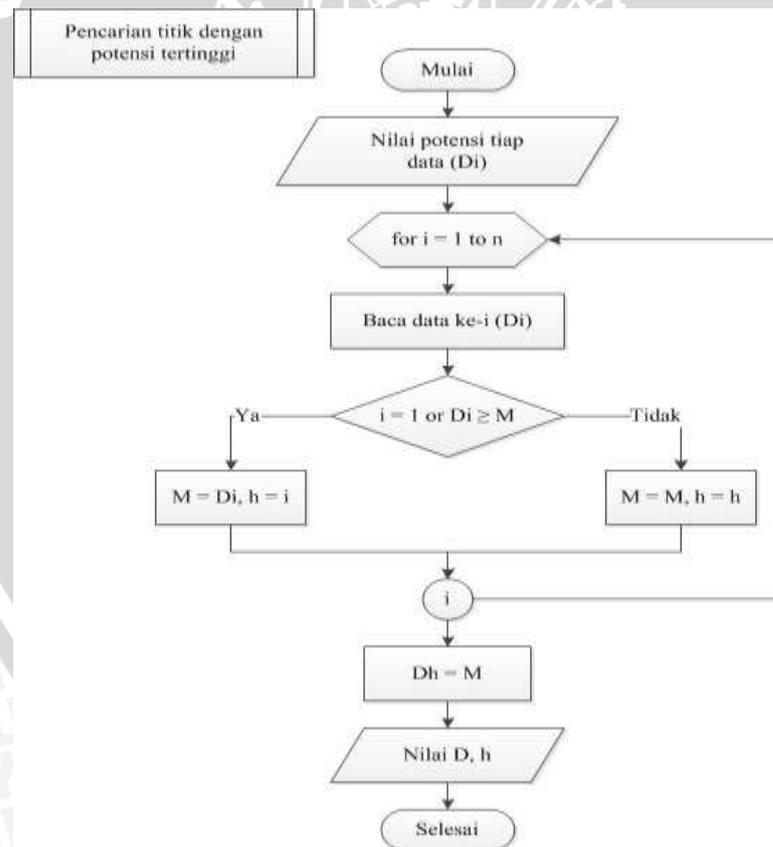
Dari Gambar (3.10), penjabaran langkah-langkah proses penentuan potensi awal tiap titik adalah sebagai berikut:

1. Persiapan data ternormalisasi (x), nilai jari-jari (r).
2. Inisialisasi variable m untuk jumlah atribut dataset yaitu 5, t untuk jumlah data input dan nilai i=1.
3. Untuk i=1 sampai i=n, dilakukan:

- Untuk $j=1$ sampai $j=m$, $T_j = X_{ij}$.
 - Untuk $k=1$ sampai $k=n$, dilakukan perhitungan potensi jarak setiap data per atribut ($Dist [k,j]$) dengan menggunakan persamaan (2.7).
 - Setelah tiap-tiap data diketahui nilai $Dist [k,j]$, dilakukan perhitungan potensi awal setiap titik data (D_i) menggunakan persamaan (2.8) atau (2.9).
4. Hasil dari proses ini berupa nilai D yang merupakan potensi tiap-tiap titik data.

3. Pencarian Titik dengan Potensi Tertinggi

Proses pencarian titik dengan potensi tertinggi merupakan proses untuk mencari nilai yang paling tinggi dari semua data yang digunakan. Alur proses ini ditunjukkan pada Gambar (3.11).



Gambar 3.11 Alur proses pencarian titik dengan potensi tertinggi

Dari Gambar (3.11), penjabaran dari langkah-langkah proses pencarian titik dengan potensi tertinggi adalah sebagai berikut:

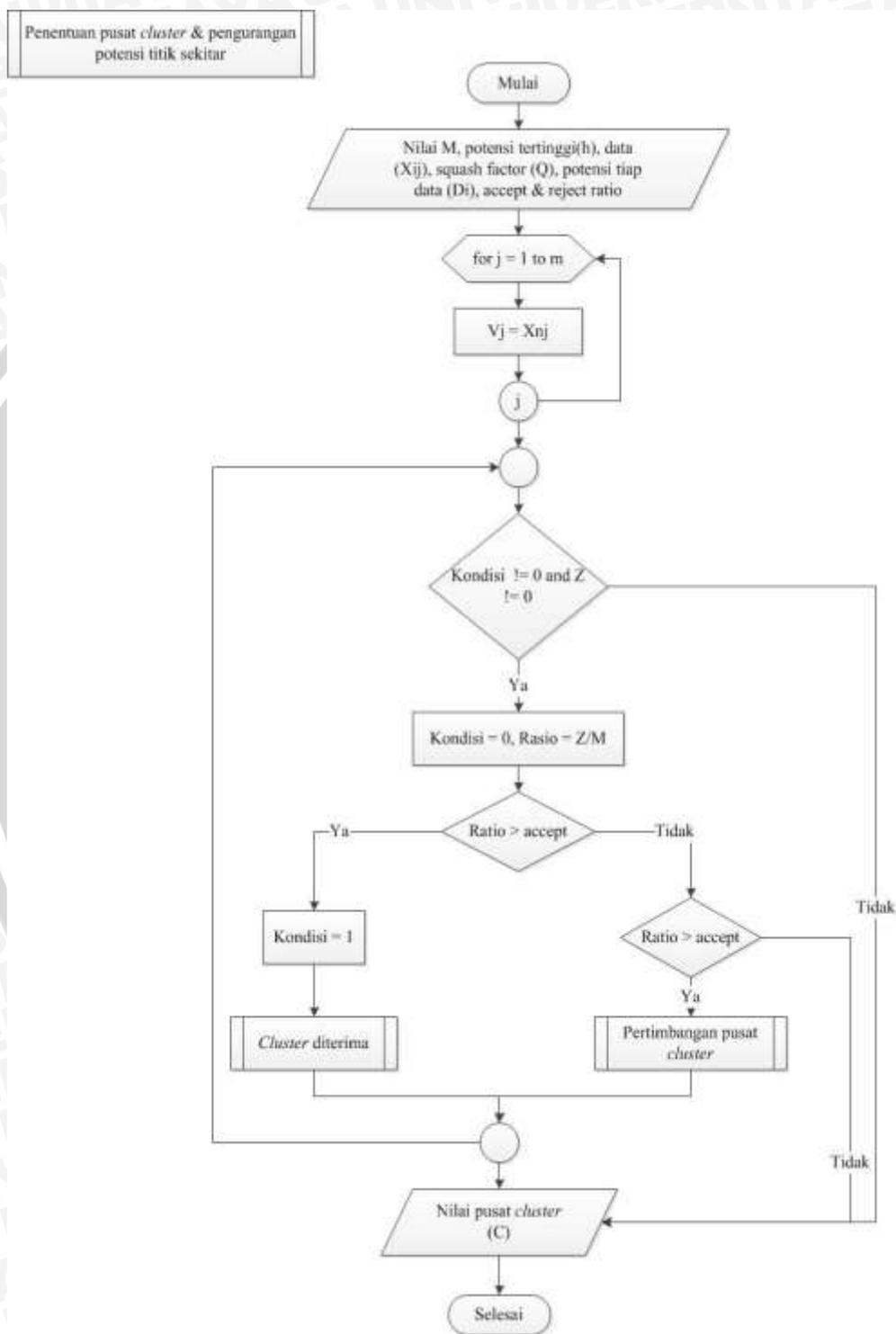
1. Persiapan nilai potensi awal setiap titik data (D).
2. Inisialisasi variable n untuk jumlah data latih, h untuk letak posisi data tertinggi dan M untuk nilai data tertinggi.
3. Untuk $i=1$ sampai $i < n$, maka dilakukan proses:
 - Membaca data ke- i (D_i).
 - Jika $i=1$ atau $D[i] \geq M$, maka $M=D_i$, $h=i$.
 - Jika tidak, maka $M=M$, $h=h$.
4. Jika $i > n$, maka proses berhenti dan didapatkan nilai M dan nilai h .

4. Penentuan Pusat Cluster

Langkah-langkah proses penentuan pusat *cluster* adalah sebagai berikut:

1. Persiapan nilai M yang merupakan titik dengan potensi awal tertinggi, h yang merupakan posisi titik tertinggi, data latih ternormalisasi (x), jari-jari (r), *squash factor* (q), *accept ratio*, *reject ratio*, dan potensi awal setiap titik data (D).
2. Menginisialisasi center untuk pusat *cluster* yang terbentuk, $V_j=X_{hj}$, C untuk jumlah *cluster*, $kondisi=1$, $Z=M$.
3. Jika $kondisi \neq 0$ dan $Z \neq 0$, maka dilakukan proses:
 - $Kondisi = 0$ (sudah tidak ada calon pusat *cluster* baru lagi).
 - $Rasio = Z/M$
 - Jika $rasio > accept\ ratio$, maka $kondisi = 1$ yang artinya ada calon pusat baru.
 - Jika tidak, dilakukan proses berikut, jika $rasio > reject\ ratio$, maka calon pusat *cluster* akan diterima sebagai pusat jika keberadaannya akan memberikan keseimbangan terhadap data-data yang letaknya cukup jauh dengan pusat *cluster*.
4. Hasil dari proses penentuan pusat *cluster* adalah pusat *cluster* yang terbentuk (center).

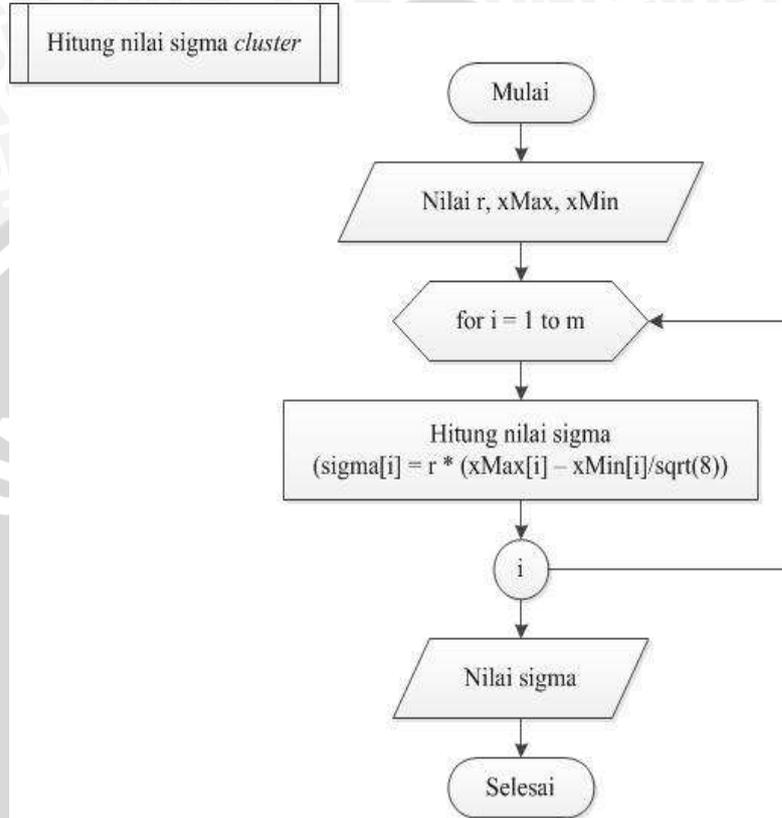
Proses penentuan pusat *cluster* merupakan proses untuk menentukan potensi data manakah yang akan dipilih sebagai pusat *cluster*. Alur proses ini ditunjukkan pada Gambar (3.12).



Gambar 3.12 Alur proses penentuan pusat *cluster*

5. Perhitungan Nilai Sigma

Proses perhitungan nilai sigma merupakan proses untuk mendapatkan nilai sigma yang digunakan untuk mencari nilai derajat keanggotaan suatu data. Alur proses ini ditunjukkan pada Gambar (3.13).

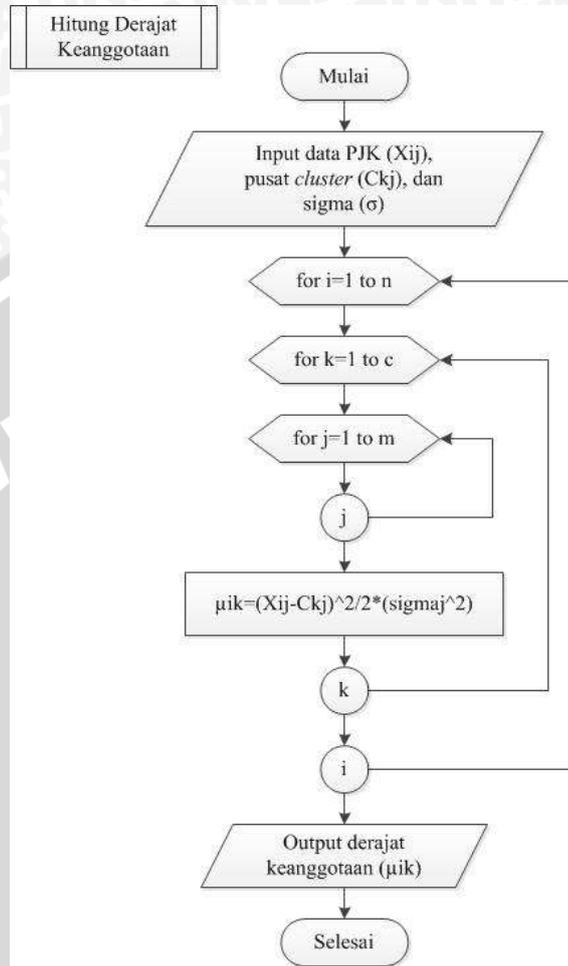


Gambar 3.13 Alur proses perhitungan nilai sigma

Pada proses perhitungan nilai sigma, masukan dari proses ini adalah nilai xMax, xMin, dan jari-jari yang telah ditentukan sejak awal dan hasil dari proses ini adalah nilai sigma yang digunakan pada saat perhitungan derajat keanggotaan. Perhitungan nilai sigma dihitung dengan menggunakan persamaan (2.18).

6. Derajat Keanggotaan

Dalam proses menghitung derajat keanggotaan ditunjukkan pada Gambar 3.14.

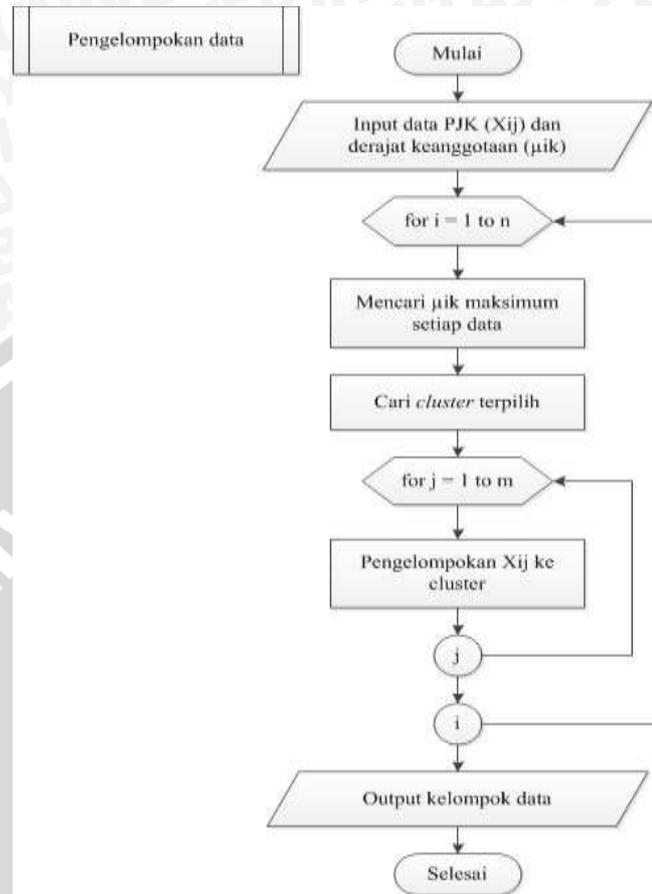


Gambar 3.14 Alur perhitungan derajat keanggotaan

Proses untuk menghitung derajat keanggotaan melibatkan data PJK (X_{ij}), pusat cluster (C_{kj}), dan sigma (σ). Proses ini menghasilkan output berupa derajat keanggotaan (μ_{ik}). Perhitungan derajat keanggotaan dihitung dengan menggunakan persamaan (2.19).

7. Pengelompokkan Data

Proses pengelompokan data ditunjukkan pada Gambar 3.15.



Gambar 3.15 Alur proses pengelompokan data *Fuzzy Subtractive*

Proses pengelompokan data melibatkan *input* berupa data PJK (X_{ij}) dan derajat keanggotaan (μ_{ik}). Proses ini menghasilkan *output* berupa kelompok data B_{ij} yang menyatakan *cluster* dari setiap data X_{ij} . Rincian alur proses pengelompokan data adalah sebagai berikut:

1. Mencari μ_{ik} maksimum setiap data, yaitu dengan cara membandingkan derajat keanggotaan μ_{ik} yang dipilih yaitu nilai yang paling mendekati 1.
2. Mencari *cluster* terpilih (k_p), yaitu dengan memilih *cluster* k yang memiliki nilai μ_{ik} maksimum.
3. Pengelompokan data ke *cluster*, yaitu mengelompokkan data ke *cluster* berdasarkan nilai k_p .

3.3.3. Perhitungan Manual Proses

3.3.3.1. Perhitungan FCM Clustering

Data rekam medic kesehatan pasien PJK terdiri atas factor risiko PJK.

Factor risiko PJK meliputi umur, LDL, kolesterol total, HDL, dan tekanan darah sistolik. Contoh data rekam medik PJK tersebut disajikan pada Tabel (3.1).

Tabel 3.1 Contoh data Penyakit Jantung Koroner (PJK)

Data ke-i (n)	Atribut (m)				
	Umur	LDL	Kol. Total	HDL	Sistolik
1	58	134	223	25	150
2	70	264	226	44	140
3	66	148	130	48	140
4	51	79	189	34	140
5	61	90	155	90	130
6	58	79	143	50	100
7	63	136	196	42	130
8	63	113	213	74	170
9	76	156	244	52	120
10	53	135	185	38	170

Berdasarkan tabel di atas, terdapat data sejumlah 10 ($n=10$) dengan atribut sejumlah 5 ($m=5$). Misalkan saja data akan dikelompokkan ke dalam 3 kelompok dengan menggunakan *clustering* FCM. Maka, nilai awal variable pada algoritma FCM dapat diinisialisasi sebagai berikut:

- Jumlah *cluster* = c = 3;
- Pangkat = w = 2;
- Maksimum Iterasi = maxIter = 50;
- Error terkecil yang diharapkan = ξ = 0.001;
- Fungsi obyektif awal = P_0 = 0;
- Iterasi awal = t = 1;

Setelah inisialisasi variable, langkah selanjutnya yaitu membangkitkan bilangan random μ_{ik} dalam matrik berukuran ($n \times m$). Dengan menjumlah kolom setiap

baris, maka didapat nilai Q_i untuk masing-masing elemen ke- i seperti ditunjukkan Tabel (3.2).

Tabel 3.2 Bilangan random dan nilai Q_i

Matriks Bilangan Random (μ_{ik})			Q_i
0.253547	0.794945	0.805154	1.853646
0.158547	0.233163	0.18771	0.57942
0.420449	0.489407	0.191354	1.10121
0.369414	0.471519	0.206901	1.047834
0.940368	0.350617	0.44564	1.736625
0.919663	0.707951	0.36234	1.989954
0.568638	0.789622	0.00809	1.36635
0.931781	0.125649	0.494481	1.551911
0.206026	0.175944	0.109059	0.491029
0.686318	0.962733	0.552904	2.201955

Setiap bilangan random tersebut kemudian dibagi dengan Q_i , sehingga terbentuk matriks partisi awal U sebagai berikut:

0.136783	0.428855	0.434362
0.273631	0.402408	0.323962
0.381806	0.444427	0.173767
0.35255	0.449994	0.197456
0.541492	0.201896	0.256613
0.462153	0.355762	0.182085
0.416173	0.577906	0.005921
0.600409	0.080964	0.318627
0.41958	0.358317	0.222103
0.311686	0.437217	0.251097

Pada iterasi pertama, dengan menggunakan persamaan (2.3) dapat dihitung 3 pusat *cluster* V_{kj} dengan $k=1,2,3$; dan $j=1,2,\dots,5$ sebagai berikut:

62.42717558	120.3093	185.5241	58.93311	138.3234
61.35990057	138.7484	190.3827	42.29274	137.7071
61.70860342	143.2009	203.2808	47.60511	145.7007

Informasi yang terkandung dalam matriks di atas adalah baris menunjukkan jumlah *cluster* ($c=3$) sedangkan kolom menunjukkan *cluster* pertama, baris kedua

menunjukkan *cluster* kedua, dan baris ketiga menunjukkan *cluster* ketiga. Pada *cluster* ke-1 terdapat 5 pusat *cluster*. Secara berturut-turut dari kiri ke kanan adalah pusat *cluster* umur, LDL, kolesterol total, HDL, dan tekanan darah sistolik. Pusat *cluster* umur adalah 62.4271; pusat *cluster* LDL adalah 120.3093; pusat *cluster* kolesterol total adalah 185.5241; pusat *cluster* HDL adalah 58.9331; pusat *cluster* tekanan darah sistolik adalah 138.3234.

Berikut proses perhitungan yang menunjukkan salah satu contoh perhitungan pusat *cluster*, yaitu perhitungan *cluster* pertama ditunjukkan pada Tabel (3.3).

Tabel 3.3 Contoh perhitungan pusat *cluster* 1

μ_{ik}	Data yang dicluster					μ_{ik}^2	$((\mu_{ik})^w * X_{ij})$				
	i=1	i=2	i=3	i=4	i=5		j=1	j=2	j=3	j=4	j=5
0.2535	58	134	223	25	150	0.01871	1.0852	2.5071	4.1722	0.4677	2.8064
0.1585	70	264	226	44	140	0.074874	5.2412	19.767	16.921	3.2944	10.482
0.4204	66	148	130	48	140	0.145776	9.6212	21.575	18.951	6.9973	20.409
0.3694	51	79	189	34	140	0.124292	6.3389	9.819	23.491	4.2259	17.401
0.9404	61	90	155	90	130	0.293213	17.886	26.389	45.448	26.389	38.118
0.9197	58	79	143	50	100	0.213585	12.388	16.873	30.543	10.679	21.359
0.5686	63	136	196	42	130	0.1732	10.912	23.555	33.947	7.2744	22.516
0.9318	63	113	213	74	170	0.360491	22.711	40.735	76.785	26.676	61.283
0.206	76	156	244	52	120	0.176047	13.38	27.463	42.956	9.1545	21.126
0.6863	53	135	185	38	170	0.097148	5.1488	13.115	17.972	3.6916	16.515
Σ						1.677336	104.71	201.8	311.19	98.851	232.01
$\Sigma((\mu_{ik})^w * X_{ij}) / \Sigma \mu_{ik}^2$							62.427	120.31	185.52	58.933	138.32

Fungsi obyektif pada iterasi pertama P_1 dapat dihitung dengan menggunakan persamaan (2.4) sebagai:

$$P_1 = \sum_{i=1}^{10} \sum_{k=1}^3 \left(\sum_{j=1}^5 (X_{ij} - V_{kj})^2 \right) (\mu_{ik})^w = 16382.7353$$

Kuadrat derajat keanggotaan data ke-i dapat dilihat pada Tabel (3.4), detail perhitungan fungsi obyektif dapat dilihat pada Tabel (3.5), sedangkan detail perhitungan derajat keanggotaan baru dapat dilihat pada Tabel (3.6).

Tabel 3.4 Kuadrat derajat keanggotaan data ke-i

Derajat Keanggotaan derajat ke-i		
μ_{i1}^2	μ_{i2}^2	μ_{i3}^2

0.018709	0.1839164	0.1886706
0.0748737	0.1619319	0.1049513
0.1457761	0.197515	0.030195
0.1242916	0.2024946	0.0389888
0.2932132	0.0407618	0.0658501
0.2135853	0.126567	0.0331548
0.1732	0.3339755	3.506E-05
0.3604907	0.0065552	0.1015233
0.1760475	0.128391	0.0493297
0.097148	0.191159	0.0630496

Tabel 3.5 Detail perhitungan fungsi obyektif

$\Sigma(X_{ij}-V_{kj})^2 \times (\mu_{ik})^w$			L1 + L2 +L3
L1	L2	L3	
54.244232	284.68067	191.82758	530.75248
1689.7843	2759.2188	1597.6561	6046.6592
580.8893	748.78293	164.38676	1494.059
307.44676	759.98995	181.60864	1249.0453
846.45963	243.09757	474.49644	1564.0536
1085.6193	924.88886	327.02479	2337.533
123.36694	33.826319	0.013478	157.20674
735.07665	21.143023	233.03338	989.25305
926.23804	487.18526	133.48421	1546.9075
169.67535	224.45279	73.137209	467.26536

Tabel 3.6 Detail perhitungan derajat keanggotaan baru



$[m\sum_{j=1}^3(X_{ik}-V_{kj})^2]-1/(w-1)$			LT=L1+L2+L3	μ_{i1}	μ_{i2}	μ_{i3}
L1	L2	L3		L1/LT	L2/LT	L3/LT
0.000345	0.000646	0.000984	0.001974501	0.174684	0.327194	0.498122
4.43E-05	5.87E-05	6.57E-05	0.000168688	0.262672	0.347906	0.389422
0.000251	0.000264	0.000184	0.000698417	0.359317	0.377684	0.262998
0.000404	0.000266	0.000215	0.0008854	0.456596	0.30093	0.242473
0.000346	0.000168	0.000139	0.000652855	0.530592	0.256836	0.212572
0.000197	0.000137	0.000101	0.000434969	0.452309	0.31461	0.233081
0.001404	0.009873	0.002601	0.013878233	0.101161	0.711419	0.187419
0.00049	0.00031	0.000436	0.001236112	0.396738	0.250819	0.352444
0.00019	0.000264	0.00037	0.000823158	0.2309	0.320153	0.448947
0.000573	0.000852	0.000862	0.002286293	0.250428	0.37251	0.377062

Matriks U diperbaiki berdasarkan persamaan (2.15). Berikut merupakan hasil dari matriks U yang sudah diperbaiki.

0.08126	0.05033	0.16516
0.01044	0.00457	0.01103
0.05913	0.02055	0.03084
0.09527	0.02076	0.03604
0.08163	0.01306	0.02330
0.04636	0.01066	0.01702
0.33068	0.76901	0.43668
0.11556	0.02415	0.07315
0.04478	0.02053	0.06206
0.13489	0.06636	0.14473

Kemudian, dilakukan pemeriksaan kondisi berhenti. Karena

$$|P_1 - P_n| = |16382,7353 - 0| = 16382.7353 \gg \xi = 10^{-3} \text{ dan iterasi} = 1 < \maxIter (\neq 2), \text{ maka proses dilanjutkan ke iterasi } 2 (t=2).$$

Pada iterasi ke-2, dapat dihitung kembali 3 pusat *cluster* sebagai berikut:

60.53068285	116.294	176.8809	56.06969	136.1484
62.5633506	139.5628	193.3501	43.9282	135.3388
63.19564707	148.9093	207.525	44.15589	142.8888

Fungsi obyektif pada iterasi kedua P_2 didapatkan sebagai berikut:

$$P_2 = \sum_{i=1}^{10} \sum_{k=1}^3 \left(\left[\sum_{j=1}^5 (X_{ij} - V_{ij})^2 \right] (\mu_{i1})^2 \right) = 17401.263193$$

Kemudian nilai matriks partisi U diperbaiki dan hasilnya adalah sebagai berikut:

0.07303	0.03021	0.20814
0.01075	0.00273	0.01383
0.07946	0.01099	0.03121
0.12357	0.01153	0.03435
0.11159	0.00747	0.02218
0.06770	0.00605	0.01722
0.26430	0.87106	0.40094
0.09438	0.01421	0.06404
0.03989	0.01370	0.08856
0.13533	0.03204	0.11953

Kemudian, dilakukan pemeriksaan kondisi berhenti. Karena syarat berhenti belum terpenuhi, dimana:

$$[P_2 - P_1] = [17401.263193 - 16382.7353] = 1018.5953884 \gg \xi = 0.001 \text{ dan iterasi } = 2 = \text{maxIter, maka iterasi diteruskan.}$$

Setelah iterasi ke-10, salah satu syarat kondisi berhenti terpenuhi, yaitu $[P_{10} - P_9] = [14364.316455 - 14230.948091] = 133.368363$. Sehingga, diperoleh matriks pusat *cluster* (matriks V) akhir, adalah:

$$V = \begin{vmatrix} 69.00 & 180.97 & 222.21 & 46.41 & 135.36 \\ 58.91 & 92.56 & 157.32 & 60.99 & 123.71 \\ 57.07 & 131.05 & 193.69 & 42.12 & 158.53 \end{vmatrix}$$

Nilai matriks U akhir berisi informasi mengenai kecenderungan suatu data untuk masuk ke dalam kelompok (*cluster*) risiko Penyakit Jantung Koroner (PJK) tertentu. Data memiliki derajat keanggotaan tertentu untuk menjadi anggota suatu kelompok risiko PJK. Derajat keanggotaan terbesar menunjukkan kecenderungan tertinggi untuk menjadi anggota kelompok.

Tabel 3.7 Kecenderungan data terhadap *cluster*

Data ke-	Derajat Keanggotaan (μ) data pada cluster ke-			Data Cenderung masuk cluster ke-		
	1	2	3	1	2	3
1	0.12104	0.41927	0.45969			√
2	0.12479	0.11944	0.75577			√
3	0.50007	0.22358	0.27635	√		
4	0.63017	0.21040	0.15943	√		
5	0.84909	0.07046	0.08045	√		
6	0.84154	0.06992	0.08855	√		
7	0.17122	0.52759	0.30119		√	
8	0.28226	0.38512	0.33262		√	
9	0.05994	0.07971	0.86036			√
10	0.12588	0.71110	0.16302		√	

Setelah dilakukan proses pengelompokkan, selanjutnya dihitung validitas *cluster* dari metode *Fuzzy Subtractive Clustering*:

1. *Partition Coefficient* (PC)

Untuk menghitung PC dapat dihitung menggunakan persamaan 2.19, ditunjukkan pada Tabel 3.8.

Tabel 3.8 Proses Perhitungan PC pada FCM

Data ke-j	μ_{1j}	μ_{1j}^2	μ_{2j}	μ_{2j}^2	μ_{3j}	μ_{3j}^2
1	0.12104	0.01465	0.41927	0.17579	0.45969	0.21132
2	0.12479	0.01557	0.11944	0.01427	0.75577	0.57119
3	0.50007	0.25007	0.22358	0.04999	0.27635	0.07637
4	0.63017	0.39712	0.21040	0.04427	0.15943	0.02542
5	0.84909	0.72095	0.07046	0.00496	0.08045	0.00647
6	0.84154	0.70818	0.06992	0.00489	0.08855	0.00784
7	0.17122	0.02932	0.52759	0.27835	0.30119	0.09072
8	0.28226	0.07967	0.38512	0.14832	0.33262	0.11063
9	0.05994	0.00359	0.07971	0.00635	0.86036	0.74021
10	0.12588	0.01585	0.71110	0.50566	0.16302	0.02658

Pada setiap μ_{ik}^2 dijumlahkan, untuk $\mu_{1j}^2 = 2.23497$; $\mu_{2j}^2 = 1.232845$;

$\mu_{3j}^2 = 1.866744$ kemudian dijumlahkan seluruhnya

$\mu_{1j} + \mu_{2j} + \mu_{3j} = 5.334559$. Sehingga nilai PC(c) dapat diketahui yaitu PC/10 = $5.334559/10 = 0.5334559$.

2. *Modified Partition Coefficient (MPC)*

Dalam perhitungan MPC dapat menggunakan persamaan 2.20, dimana nilai MPC memiliki nilai 1.0.

3. *Clasification Entropy (CE)*

Untuk proses perhitungan diatas menggunakan persamaan 2.21, ditunjukkan pada Tabel 3.9.

Tabel 3.9 Proses Perhitungan CE pada FCM

Data ke / j	μ_{1j}	$\mu_{1j} \log_2 \mu_{1j}$	μ_{2j}	$\mu_{2j} \log_2 \mu_{2j}$	μ_{3j}	$\mu_{3j} \log_2 \mu_{3j}$
1	0.12104	-0.36874	0.41927	-0.52578	0.45969	-0.51544
2	0.12479	-0.37467	0.11944	-0.36616	0.75577	-0.30532
3	0.50007	-0.49997	0.22358	-0.48319	0.27635	-0.51275
4	0.63017	-0.41981	0.21040	-0.47314	0.15943	-0.42233
5	0.84909	-0.20039	0.07046	-0.26965	0.08045	-0.29250
6	0.84154	-0.20946	0.06992	-0.26835	0.08855	-0.30969
7	0.17122	-0.43594	0.52759	-0.48671	0.30119	-0.52144
8	0.28226	-0.51510	0.38512	-0.53016	0.33262	-0.52822
9	0.05994	-0.24336	0.07971	-0.29087	0.86036	-0.18669
10	0.12588	-0.37637	0.71110	-0.34978	0.16302	-0.42661

Kemudian data yang sudah di-log dijumlahkan dan didapatkan hasil CE sebesar 11,708577 sedangkan untuk nilai CE(c) dimana nilai CE dibagi dengan 10 sehingga nilai CE(c) = 1.1708577.

3.3.3.2. Perhitungan Subtractive Clustering



Contoh perhitungan manual dari skripsi ini adalah dengan menggunakan data yang terdapat pada Tabel 3.10.

Tabel 3.10 Contoh Data Penyakit Jantung Koroner (PJK)

Data ke-i (n)	Atribut (m)				
	Umur	LDL	Kol. Total	HDL	Sistolik
1	58	134	223	25	150
2	70	264	226	44	140
3	66	148	130	48	140
4	51	79	189	34	140
5	61	90	155	90	130
6	58	79	143	50	100
7	63	136	196	42	130
8	63	113	213	74	170
9	76	156	244	52	120
10	53	135	185	38	170

Dari data tersebut dilakukan proses *clustering* menggunakan metode *Subtractive Clustering*.

1. Nilai jari-jari yang diberikan = 0.65, nilai *squash factor* (q) = 1.25, nilai *acceptrato* = 0.3 dan nilai *reject rasio* = 0.1. Untuk nilai xMin = [32; 79; 113; 13; 120] dan xMax = [82; 380; 390; 90; 200].
2. Data pada tabel 3.10 dinormalisasikan dengan menggunakan persamaan 2.6. Data yang telah dinormalisasi yaitu X ditunjukkan pada Tabel 3.11.

Tabel 3.11 Tabel data ternormalisasi

Data ke-i	j1	j2	j3	j4	j5
1	0.52	0.182724252	0.397112	0.155844	0.375
2	0.76	0.61461794	0.407942	0.402597	0.25
3	0.68	0.22923588	0.061372	0.454545	0.25

4	0.38	0	0.274368	0.272727	0.25
5	0.58	0.03654485	0.151625	1	0.125
6	0.52	0	0.108303	0.480519	-0.25
7	0.62	0.189368771	0.299639	0.376623	0.125
8	0.62	0.112956811	0.361011	0.792208	0.625
9	0.88	0.255813953	0.472924	0.506494	0
10	0.42	0.186046512	0.259928	0.324675	0.625

3. Dari data tabel 3.11 dilakukan proses penentuan potensi awal setiap titik. Untuk contoh perhitungan Potensi awal data pertama (D1) ditunjukkan pada Tabel 3.12.

Tabel 3.12 Contoh perhitungan potensi data pertama

k	j = 1	j=2	j=3	j=4	j=5	DSk	exp(-4 * DSk)
	Distkj^2	Distkj^2	Distkj^2	Distkj^2	Distkj^2		
1	0	0	0	0	0	0	1
2	1.44	4.66330	0.00293	1.52218	0.39063	8.01904	0
3	0.64	0.05408	2.81803	2.23056	0.39063	6.13330	0
4	0.49	0.83470	0.37665	0.34154	0.39063	2.43352	0.00006
5	0.09	0.53421	1.50660	17.81498	1.56250	21.50829	0
6	0	0.83470	2.08526	2.63535	9.76563	15.32094	0
7	0.25	0.00110	0.23752	1.21859	1.56250	3.26971	0
8	0.25	0.12169	0.03258	10.12397	1.56250	12.09074	0
9	3.24	0.13355	0.14369	3.07387	3.51563	10.10674	0
10	0.25	0.00028	0.47049	0.71260	1.56250	2.99586	0.00001
						D1	1.000067565

Demikian seterusnya, potensi untuk 10 data berikutnya dapat dicari dengan cara yang sama. Hasil akhir potensi awal untuk 10 data ditunjukkan pada Tabel 3.13.

Tabel 3.13 Data potensi awal

I	Di
1	1.000067565
2	1.000000000381

3	1.000232703
4	1.000056079
5	1.0000000000000001
6	1.000000000069559
7	1.0002420623
8	1.000000000001243
9	1.00000143662
10	1.00000626238

4. Dari data 10 tersebut, ternyata potensi tertinggi terletak pada data ke-7 yaitu sebesar 1.0002420623. Hal ini berarti bahwa data tersebut menjadi calon pusat *cluster* (V). Vektor $V = X_7$ yaitu $V_1 = 0.62$; $V_2 = 0.18937$; $V_3 = 0.29964$; $V_4 = 0.37662$; $V_5 = 0.125$; serta nilai $M = 2.66087255$, nilai $Z = 2.66087255$ juga. Sehingga untuk iterasi ke-1:

- Rasio $= Z/M = 1$, dan rasio $> \textit{accept ratio}$ sehingga kondisi $= 1$, maka calon pusat *cluster* akan diterima sebagai pusat *cluster*, sehingga hasil calon pusat *cluster* ditunjukkan pada Tabel 3.14.

Tabel 3.14 Pusat *cluster* pada k1

J	V_j	Center $_j$
1	0.62	0.62
2	0.189369	0.189368771
3	0.299639	0.299638989
4	0.376623	0.376623377
5	0.125	0.125

5. Karena telah ditentukan pusat *cluster*, maka potensi setiap titik data dikurangi. Hasil dari perhitungan potensi baru ditunjukkan pada Tabel 3.15.

Tabel 3.15 Tabel Perhitungan Potensi baru

i	j = 1	j = 2	j = 3	j = 4	j = 5	STi	Dci	Di
	S_{ij}^2	S_{ij}^2	S_{ij}^2	S_{ij}^2	S_{ij}^2			
1	0.16	0.00071	0.15202	0.77990	1	2.092617	2.316628674	0.999835

2	0.31360	2,89339	0.18767	0.01079	0.25	3.655457	4.466063137	0.999999
3	0.05760	0.02543	0.90834	0.09715	0.25	1.338519	0.004729971	0.995502
4	0.92160	0.57377	0.01022	0.17271	0.25	1.928296	4.470032393	0.999618
5	0.02560	0.37368	0.35053	6.21757	0	6.967389	7.879705244	0.999999
6	0.16	0.57377	0.58575	0.17271	2.25	3.742228	3.156370100	0.999999
7	0	0	0	0	0	0	1.000242062	0
8	0	0.09342	0.06026	2.76337	4	6.9170511	9.637325506	1.000000
9	1.08160	0.07064	0.48044	0.26986	0.25	2.152543	1.822862621	0.999819
10	0.64	0.00018	0.02523	0.04318	4	4.708585	6.613840056	1.000006

6. Dari nilai potensi baru tiap titik tersebut dicari nilai tertinggi, sehingga potensi tertinggi terletak pada data ke-10, yaitu sebesar 1.00000625576. Hal ini menunjukkan bahwa data ke-10 akan terpilih menjadi calon pusat *cluster*. Vektor $V = X_{10}$ yaitu $V_1 = 0.42$; $V_2 = 0.18604$; $V_3 = 0.25992$; $V_4 = 0.32467$; $V_5 = 0.625$ serta nilai $Z = 1.28764$. Sehingga untuk iterasi ke-2:

- Rasio = $Z/M = 0.99976425$, dan rasio $> \text{accept ratio}$, maka pada data ke-10 merupakan calon pusat baru yang diterima sebagai pusat *cluster*.

7. Sampai ditemukan di titik iterasi akhir, yaitu pada iterasi ke-10, dengan potensi tertinggi terletak pada data ke-4, yaitu sebesar 0.995501. Hal ini menunjukkan bahwa data ke-3 akan terpilih menjadi calon pusat *cluster*. Vektor $V = X_4$ yaitu $V_1 = 0.68$; $V_2 = 0.2292$; $V_3 = 0.0.61371$; $V_4 = 0.45454$; $V_5 = 0.25$ serta nilai $Z = 0.0$. Sehingga untuk iterasi ke-10:

- Rasio = $Z/M = 0.995501$, dan rasio $< \text{accept ratio}$ tetapi rasio $> \text{reject ratio}$, maka calon pusat *cluster* (data ke-3) baru akan diterima sebagai pusat *cluster* baru jika jaraknya cukup jauh dengan pusat *cluster* yang sudah ada. Untuk itu dikerjakan langkah-langkah sebagai berikut:
 - $Md = -1$
 - Kerjakan $i=1$ sampai $i=C$, karena *cluster* yang sudah diketahui sejumlah 3 *cluster*.
 - Pada $Sd1$ diketahui dengan hasil $Sd1 = 0.285250989$, maka $Md=Sd1$. Setelah dilakukan perhitungan untuk $i=2$ didapatkan hasil

$Sd_2 = 0.42543$, dimana $Sd_2 > Md$ sehingga untuk nilai Md tetap pada Sd_1 , selanjutnya dilakukan perhitungan pada $i=3$, didapatkan hasil $Sd_3 = 1.42237$, maka $Sd_3 < Md$, sehingga nilai Md berubah menjadi $Md = Sd_3$ yaitu 1.42237 . Jarak terdekat data ke-3 dengan pusat *cluster* ($Smd = 1.19263$), sehingga $rasio+Smd \geq 1$, maka data ke-3 diterima sebagai pusat *cluster*.

8. Selanjutnya dilakukan pencarian potensi baru (D). potensi baru untuk data ke-4 ditunjukkan pada Tabel 3.16.

Tabel 3.16 Potensi baru data ke-3

Di	Perbaikan Di
0	0
0	0
0.995501	0
0	0
0	0
0	0
0	0
0	0
0	0
0	0

Dari potensi baru diperoleh data tertinggi data ke-3. Sehingga $rasio = Z/M = 0.0$. Karena $rasio < reject\ ratio$ iterasi dihentikan karena tidak ada lagi data yang akan dipertimbangkan menjadi pusat *cluster*.

9. Karena iterasi pencarian pusat *cluster* telah dihentikan, langkah selanjutnya adalah mengembalikan nilai pusat *cluster* dan menghitung nilai sigma yang ditunjukkan pada Tabel 3.17 dan Tabel 3.18.

Tabel 3.17 Pusat *cluster* dari bentuk ternormalisasi yang telah dikembalikan

i	J				
	1	2	3	4	5
	Centerij	Centerij	Centerij	Centerij	Centerij
1	63	136	115	42	130

2	53	135	104	38	170
3	61	90	74	90	130

Tabel 3.18 Nilai Sigma

j	Sigmaj
1	11.49049
2	69.17272
3	63.65729
4	17.69535
5	18.38478

- Selanjutnya untuk hasil kecenderungan tiap data masuk pada suatu *cluster* ditunjukkan pada Tabel 3.19.

Tabel 3.19 Pengelompokkan data

Data ke- i	μ_{1i}	μ_{2i}	μ_{3i}	Cluster 1	Cluster 2	Cluster 3
1	0.2898954	0.32157	0.0003		√	
2	0.1149805	0.01192	0.0005	√		
3	0.4529274	0.0803	0.0306	√		
4	0.3194981	0.18237	0.0034	√		
5	0.0162005	0.00071	1			√
6	0.109226	0.0003	0.0192	√		
7	1	0.06166	0.0162	√		
8	0.0166903	0.07461	0.0383		√	
9	0.279796	0.00152	0.0088	√		
10	0.0616584	1	0.0007		√	

Setelah dilakukan proses pengelompokkan, selanjutnya dihitung validitas *cluster* dari metode *Fuzzy Subtractive Clustering*:

- Partition Coefficient* (PC)

Untuk menghitung PC dapat dihitung menggunakan persamaan 2.19, ditunjukkan pada Tabel 3.20.

Tabel 3.20 Proses Perhitungan PC pada FSC

Data ke / j	μ_{1j}	μ_{1j}^2	μ_{2j}	μ_{2j}^2	μ_{3j}	μ_{3j}^2
1	0.2898954	0.0840393	0.32157	0.1034	0.0003	8E-08
2	0.1149805	0.0132205	0.01192	0.0001	0.0005	2E-07
3	0.4529274	0.2051432	0.0803	0.0064	0.0306	0.0009
4	0.3194981	0.102079	0.18237	0.0333	0.0034	1E-05
5	0.0162005	0.0002625	0.00071	5E-07	1	1
6	0.109226	0.0119303	0.0003	8.9E-08	0.0192	0.0004
7	1	1	0.06166	0.0038	0.0162	0.0003
8	0.0166903	0.0002786	0.07461	0.00557	0.0383	0.0015
9	0.279796	0.0782858	0.00152	2.3E-06	0.0088	8E-05
10	0.0616584	0.0038018	1	1	0.0007	5E-07

Pada setiap μ_{ik}^2 dijumlahkan, untuk $\mu_{1j}^2 = 1.499041$; $\mu_{2j}^2 = 1.15263$;

$\mu_{3j}^2 = 1.0031$ kemudian dijumlahkan seluruhnya $\mu_{1j} + \mu_{2j} + \mu_{3j} = 3.6548$.

Sehingga nilai PC(c) dapat diketahui yaitu $PC/10 = 3.6548/10 = 0.3654$.

2. *Modified Partition Coefficient (MPC)*

Dalam perhitungan MPC dapat menggunakan persamaan 2.20, dimana nilai MPC memiliki nilai 0.90356.

3. *Clasifcation Entropy (CE)*

Untuk proses perhitungan diatas menggunakan persamaan 2.21, ditunjukkan pada Tabel 3.21.

Tabel 3.21 Proses Perhitungan CE pada FSC

Data ke / j	μ_{1j}	$\mu_{1j} \log_2 \mu_{1j}$	μ_{2j}	$\mu_{2j} \log_2 \mu_{2j}$	μ_{3j}	$\mu_{3j} \log_2 \mu_{3j}$
-------------	------------	----------------------------	------------	----------------------------	------------	----------------------------

1	0.2898954	-0.517868	0.32157	-0.5263	0.0003	-0.003
2	0.1149805	-0.358801	0.01192	-0.0762	0.0005	-0.005
3	0.4529274	-0.517537	0.0803	-0.2922	0.0306	-0.154
4	0.3194981	-0.525932	0.18237	-0.4477	0.0034	-0.028
5	0.0162005	-0.096357	0.00071	-0.0074	1	0
6	0.109226	-0.348935	0.0003	-0.0035	0.0192	-0.11
7	1	0	0.06166	-0.2478	0.0162	-0.096
8	0.0166903	-0.098553	0.07461	-0.2794	0.0383	-0.18
9	0.279796	-0.51414	0.00152	-0.0142	0.0088	-0.06
10	0.0616584	-0.24784	1	0	0.0007	-0.007

Kemudian data yang sudah di-log dijumlahkan dan didapatkan hasil CE sebesar 5.765, sedangkan untuk nilai CE(c) dimana nilai CE dibagi dengan 10 sehingga nilai CE(c) = 0.576478.

Jadi dari hasil perhitungan di atas, dimana hasil dari FCM dengan parameter jumlah *cluster* (c) = 3; maksimum iterasi (maxIter) = 50; error terkecil yang diharapkan (ξ) = 0.001; fungsi obyektif awal (P_0) = 0; iterasi awal (t) = 1 dengan melalui proses perhitungan yang sesuai dengan algoritma *Fuzzy C-Means Clustering* didapatkan hasil indeks validitas *cluster* antara lain diukur dengan parameter *Partition Coefficient* (PC) = 0.5334559; *Modified Partition Coefficient* (MPC) = 1.0; dan *Clasification Entropy* (CE) = 1.1708577. Sedangkan untuk hasil dari *Fuzzy Subtractive* dengan parameter jari-jari (r) = 0.65; *squash factor* (q) = 1.25; *accept ratio* = 0.3; *reject ratio* = 0.1 dengan melalui proses perhitungan yang sesuai dengan algoritma *Fuzzy Subtractive Clustering* didapatkan hasil indeks validitas *cluster* antara lain yang diukur dengan parameter *Partition Coefficient* (PC) = 0.3654; *Modified Partition Coefficient* (MPC) = 0.90356; dan *Clasification Entropy* (CE) = 0.576478. Jadi dari kedua hasil *clustering* dan hasil indeks validitas *cluster* di atas didapatkan bahwa kemungkinan yang terbaik dan efektif terdapat pada metode *Fuzzy C-Means Clustering* karena perbandingan dari kedua metode yaitu *Fuzzy C-Means Clustering* dan *Fuzzy Subtractive Clustering*, dimana FCM lebih banyak memiliki kombinasi indeks validitas terbanyak karena proses perbandingan indeks validitas pada FCM lebih unggul pada saat perankingan. Dalam proses ini hanya pada satu percobaan dengan 10 data dan satu kelompok parameter dari kedua metode, jadi kemungkinan bisa terdapat perubahan yang nantinya dapat dilihat pada hasil akhir pada penelitian ini, dimana

akan dilakukan beberapa percobaan dengan mengubah nilai parameter dan menggunakan seluruh data PJK yaitu 100 data PJK.

3.3.4 Perancangan Tabel dan Struktur Data

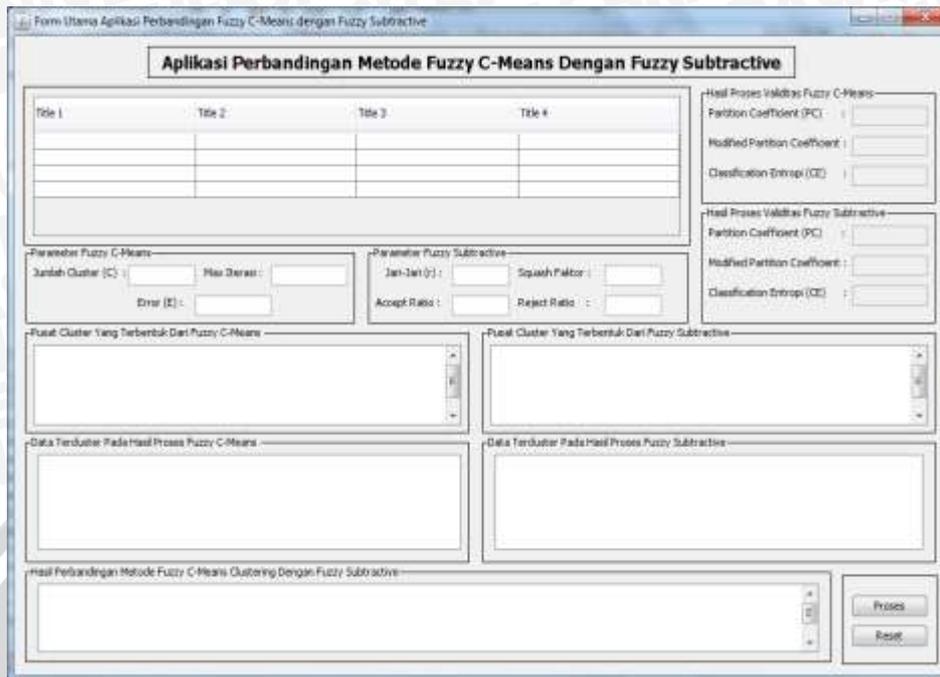
Dalam penelitian ini, digunakan satu tabel untuk data PJK. Atribut dari tabel tersebut adalah id_data, umur, kadar LDL, kolesterol total, kadar HDL, tekanan darah sistolik. Struktur dari tabel dataset PJK ditunjukkan pada Tabel 3.22.

Tabel 3.22 Tabel Dataset PJK

Nama Kolom	Tipe Data	Keterangan
Id_data	Auto Number	Id dari tiap data pada tabel dataset PJK
Umur	Number	Umur pasien
Kadar LDL	Number	Kadar LDL pasien
Kolesterol Total	Number	Kadar kolesterol pasien
Kadar HDL	Number	Kadar HDL pasien
Tekanan Darah Sistolik	Number	Tekanan darah sistolik pasien

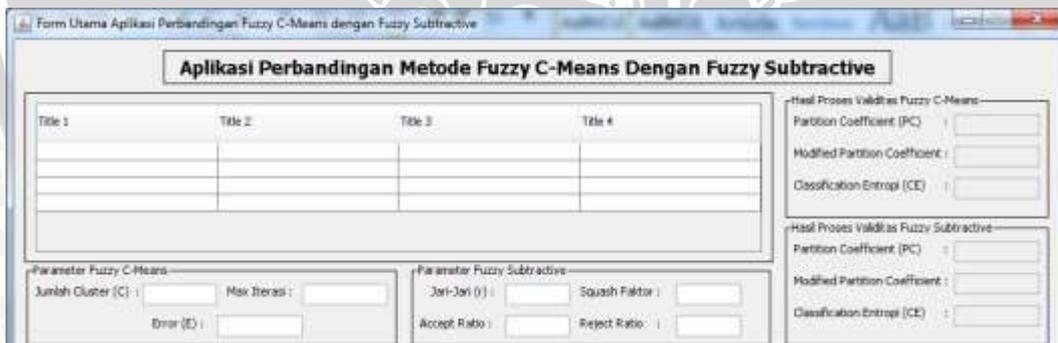
3.3.5 Perancangan Antarmuka

Secara umum tampilan antarmuka sistem hanya terdapat satu bagian, yang seluruhnya terdapat didalamnya. Untuk tampilan antarmuka ditunjukkan pada Gambar 3.16.



Gambar 3.16 Tampilan antarmuka sistem

Pada tampilan antarmuka sistem tersebut terdapat bagian-bagian yang nantinya akan digunakan untuk melakukan proses *clustering* serta mengetahui hasil perbandingan kedua metode yang ditunjukkan pada Gambar 3.17.



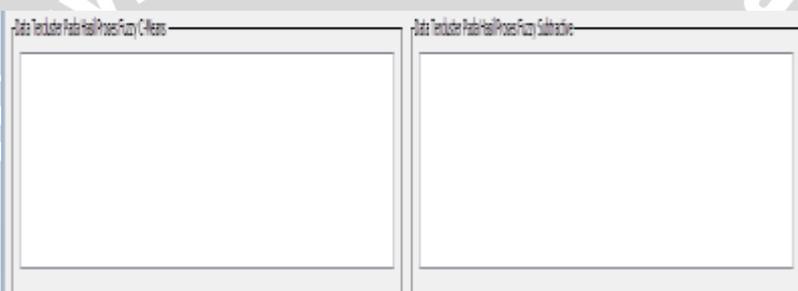
Gambar 3.17 Tampilan Proses *Clustering*

Pada Gambar 3.17 di atas, menampilkan tabel dataset PJK serta user yang nantinya menginputkan parameter pada metode *Fuzzy C-Means Clustering* dan *Fuzzy Subtractive Clustering*. Selain itu ada juga untuk menampilkan pusat *cluster* yang telah terbentuk dari kedua metode tersebut yang akan ditunjukkan pada Gambar 3.18.



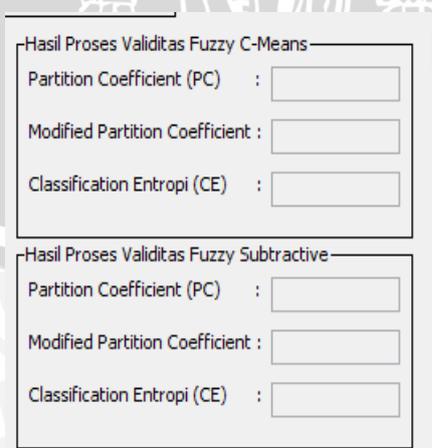
Gambar 3.18 Tampilan untuk pusat *cluster* yang telah terbentuk

Setelah user sudah melalui tahap proses perhitungan kedua metode tersebut maka untuk hasil pengelompokkan data dapat ditampilkan pada sistem yang ditunjukkan pada Gambar 3.19.



Gambar 3.19 Tampilan untuk data yang sudah ter-*cluster*

Untuk hasil akhir pengelompokkan ini, lalu dilakukan proses perhitungan indeks validitas yang fungsinya untuk mengetahui hasil perbandingan antara kedua metode pada kasus PJK tersebut, yang tampilan pada sistem ditunjukkan pada Gambar 3.20.



Gambar 3.20 Tampilan hasil proses validitas *cluster*

Setelah mengetahui hasil perbandingan tersebut maka sistem akan memproses hasil akhir dimana hasil perbandingan dari proses indeks validitas dibandingkan metode mana yang paling efektif dan ini dapat dilihat pada tampilan yang akan ditunjukkan pada Gambar 3.21.



Gambar 3.21 Tampilan hasil perbandingan



BAB IV IMPLEMENTASI

4.1. Lingkungan Implementasi

Dalam menerapkan metode penelitian ke dalam sistem serta dalam pengembangan sistem, perlu adanya beberapa faktor yang perlu diperhatikan dalam menerapkan metode dan pengembangan sistem. Tujuannya yaitu untuk memenuhi kebutuhan dari sistem yang nantinya akan dikembangkan dan metode yang diimplementasikan. Beberapa aspek yang perlu diperhatikan yaitu dari segi perangkat keras (*hardware*) maupun perangkat lunak (*software*).

4.1.1. Lingkungan Implementasi Perangkat Keras

Dalam mengembangkan sistem dan penerapan metode dalam penelitian mengenai perbandingan metode *Fuzzy C-Means Clustering* dan *Fuzzy Subtractive Clustering* pada kasus risiko penyakit jantung koroner (PJK) digunakan beberapa komponen perangkat keras sebagai berikut:

1. Processor Intel Core2Duo CPU T8100 @ 2.10 Ghz (2CPUs)
2. Memory 2048MB RAM
3. Hard disk 250GB
4. Keyboard
5. Mouse

4.1.2. Lingkungan Implementasi Perangkat Lunak

Pengembangan sistem dan penelitian mengenai perbandingan metode *Fuzzy C-Means Clustering* dan *Fuzzy Subtractive Clustering* pada kasus risiko penyakit jantung koroner (PJK) juga membutuhkan beberapa perangkat lunak. Beberapa perangkat lunak yang digunakan sebagai berikut:

1. Sistem operasi yang digunakan Windows 7 Ultimate 32-bit
2. Aplikasi pembangun GUI dan code menggunakan NetBeans IDE 6.7.1
3. Bahasa pemrograman yang dipakai yaitu bahasa pemrograman java
4. Komponen java yang digunakan yaitu JDK 1.6.0

5. Basis data yang digunakan yaitu MySQL Server Versi 5.1.33 dengan menggunakan tool XAMPP Versi 1.7.1
6. Microsoft Word 2010, Microsoft Excel 2010, Microsoft Visio 2010

4.2. Implementasi Program

Pada implementasi program, program dibuat dengan menggunakan method-method. Dimana method-method tersebut merupakan tahapan proses yang dilakukan dalam program. Method-method ini mewakili proses dari perhitungan perbandingan kedua metode. Dimana proses perbandingan program ini proses *clustering* pada metode *Fuzzy C-Means Clustering* dan pada metode *Fuzzy Subtractive Clustering*, proses validitas dan proses perbandingan kedua metode.

4.2.1. Tahapan *Clustering* pada Metode *Fuzzy C-Means Clustering*

Pada tahapan ini dilakukan beberapa proses dengan menggunakan data PJK. Proses-proses yang dilakukan dalam tahapan ini antara lain proses membangkitkan bilangan random, hitung pusat *cluster*, hitung fungsi obyektif, hitung perubahan matriks, proses *clustering* data.

4.2.1.1. Proses Membangkitkan Bilangan Random

Proses membangkitkan bilangan random tersebut digunakan sebagai elemen-elemen matriks partisi awal U. Kode program proses membangkitkan bilangan random digambarkan pada Kode Program 4.1.

```
public static void randomMatrixU() {
    /*for (int i=0; i<jumlahData; i++)
    {
        for (int j=0; j<jumlahClusterFcm; j++)
        {
            bilRandomMatrixU[i][j]=Math.random();
        }*/
    }
```

Kode Program 4.1 Proses Membangkitkan Bilangan Random

4.2.1.2. Proses Menghitung Pusat Cluster

Dalam proses menghitung pusat *cluster* diperlukan *input* data berupa data PJK dan serajat keanggotaan hasil dari proses pembentukan matriks partisi awal U. Kode program proses menghitung pusat *cluster* digambarkan pada Kode Program 4.2.

```
public static void hitungPusatCluster()
{
    for (int x=0; x<jumlahClusterFcm; x++)
    {
        for (int y=0; y<jumlahAtribut; y++)
        {
            double temp=0;
            for(int z=0; z<jumlahData; z++)
            {
                temp=temp+prosesPusatCluster[x][z][y];
            }
            if(temp==0){
                pusatClusterFCM[x][y]=0;
            }else{
                pusatClusterFCM[x][y]=temp/tempMiuKuadrat[x];
            }
        }
        System.out.println("====Perhitungan Pusat
        Cluster====");
    }
}
```

Kode Program 4.2 Proses Menghitung Pusat Cluster

4.2.1.3. Proses Menghitung Fungsi Obyektif

Pada proses menghitung fungsi obyektif tersebut diperlukan *input* data antara lain jumlah *cluster*, jumlah atribut, dan jumlah data. Untuk data-data tersebut diproses menggunakan Persamaan 2.4. Kode program proses menghitung fungsi obyektif dapat digambarkan pada Kode Program 4.3.

```

public static void hitungFungsiObjektif()
{
    double temp=0;
    for(int i=0; i<jumlahData; i++)
    {
        temp=0;
        for(int j=0; j<jumlahClusterFcm; j++)
        {
            temp=temp+fO[i][j];
        }
        finalFO[i]=temp;
    }
    System.out.println("====Perhitungan Nilai Akhir
    Fungsi Objektif====");
    for(int i=0; i<jumlahData; i++)
    {
        System.out.println(finalFO[i]);
    }
    Pt=0;
    for(int i=0; i<jumlahData; i++)
    {
        Pt=Pt+finalFO[i];
    }
    System.out.println("====Nilai Pt====");
    System.out.println(Pt);
}

```

Kode Program 4.3 Proses Perhitungan Fungsi Obyektif

4.2.1.4. Proses Perubahan Matriks

Proses perubahan matriks dilakukan untuk memperbaharui nilai matriks awal berdasarkan pusat *cluster* yang terbentuk sesuai dengan Persamaan 2.5.

Kode Program pada proses perubahan matriks dapat digambarkan pada Kode Program 4.4.

```

public static void hitungPerubahanMatriks(){
    for(int i=0; i<jumlahClusterFcm; i++){
        for (int j=0; j<jumlahData; j++){
            if(sumFO[i]==0) {
            }else{
                System.out.println("====Perubahan Matriks
                U====");
            }
        }
    }
}

```

Kode Program 4.4 Proses Perubahan Matriks

4.2.1.5. Proses *Clustering* Data

Proses *clustering* dimulai dengan membentuk matriks awal derajat keanggotaan. Pembentukan matriks awal ini dilakukan sekali pada setiap proses *clustering*. Setelah itu, secara berturut-turut dilakukan pencarian pusat *cluster*, perhitungan fungsi obyektif, perubahan nilai matriks derajat keanggotaan. Rangkaian proses tersebut dilakukan secara iteratif hingga kondisi berhenti dengan melihat kondisi nilai *error* atau nilai maksimum iterasi. Kode program proses *clustering* data dapat digambarkan pada Kode Program 4.5.

```

public static void prosesClusteringData ()
{
    int tempCluster=0;
    double tempMiu =0.0;
    for(int i=0; i<jumlahData; i++)
    {
        tempCluster=0;
        tempMiu=0.0;
        for(int j=0; j<jumlahClusterFcm; j++)
        {
            if(tempMiu<miu[i][j])
            {
                tempMiu=miu[i][j];
                tempCluster=j;
            }
        }
        if(dataTercluster[tempCluster]==null)
        {
            dataTercluster[tempCluster]="          Data          Ke-
            "+(i+1)+", ";
        }else{
        }
    }
    for(int i=0; i<jumlahClusterFcm; i++)
    {
        System.out.println("=====Data          Miu
        Akhir=====");
    }
}

```

Kode Program 4.5 Proses *Clustering* Data

4.2.2. Tahapan *Clustering* pada Metode *Fuzzy Subtractive Clustering*

Pada tahapan ini dilakukan beberapa proses dengan menggunakan data PJK. Proses-proses yang dilakukan dalam tahap ini antara lain proses normalisasi, hitung potensi awal, hitung titik potensi tertinggi, hitung pusat *cluster*, hitung sigma, hitung derajat keanggotaan, hasil *cluster*.

4.2.2.1. Proses Normalisasi

Proses normalisasi merupakan langkah selanjutnya setelah proses inisialisasi awal yang telah diketahui. Proses ini merupakan untuk mendapatkan nilai data ternormalisasi. Kode program proses normalisasi dapat digambarkan Kode Program 4.6.

```
public static void prosesNormlisasi() {  
    for(int i=0; i<jumlahAtribut; i++)  
    {  
        for(int j=0; j<jumlahData; j++)  
        {  
            normalisasiAtribut[j][i]=((dataSet[j][i]-  
            xMin[i])/(xMax[i]-xMin[i]));  
        }  
    }  
    System.out.println("=====Normalisasi=====  
    ===  
    =====");  
}
```

Kode Program 4.6 Proses Normalisasi

4.2.2.2. Proses Menentukan Potensi Awal

Setelah masing-masing data telah ternormalisasi, langkah selanjutnya diimplementasikan pada proses menentukan potensi awal. Proses ini merupakan proses untuk menentukan potensi titik awal tiap data. Kode program proses menentukan potensi awal dapat digambarkan pada Kode Program 4.7.

```

public static void hitungPotensiAwal() {
    double tempD=0.0;
    x=0; y=0;
    for(int i=0; i<jumlahData; i++){
        for(int j=0; j<jumlahData; j++){
            tempD=tempD+expDsk[j][i];
            x++;
            if(x==10) {
                nilaiD[i]=tempD;
                x=0;tempD=0.0;
            }
        }
        System.out.println(nilaiD[i]); }
}

```

Kode Program 4.7 Proses Menentukan Potensi Awal

4.2.2.3. Proses Hitung Titik Potensi Tertinggi

Langkah selanjutnya setelah potensi awal tiap data ditemukan adalah mencari nilai tertinggi dari semua data tersebut yang diimplementasikan pada metode cari titik potensi tertinggi. Kode program proses hitung titik potensi tertinggi dapat digambarkan pada Kode Program 4.8.

```

public static void hitungTitikPotensiTertinggi() {
    M=0.0;
    maxH=0;
    for(int i=0; i<jumlahData; i++)
    {
        if(M<nilaiD[i])
        {
            M=nilaiD[i];
            maxH=i;
        }
    }
    mPar=M;
    Z=M;
    System.out.println("=====Potensi Nilai
        Tertinggi=====");
    int i=0;
    System.out.println("=====Vj=====");
    for(int j=0; j<jumlahAtribut; j++)
    {
        tempV[j]=normalisasiAtribut[maxH][j];
    }
}

```

```

        System.out.print(tempV[j]+" ");
    }
    System.out.println("");
}

```

Kode Program 4.8 Proses Hitung Titik Potensi Tertinggi

4.2.2.4. Proses Menghitung Pusat Cluster

Langkah selanjutnya setelah nilai potensi titik tertinggi diketahui adalah menentukan pusat *cluster* data yang diimplementasikan pada proses menghitung pusat *cluster*. Kode program proses menghitung pusat *cluster* dapat digambarkan pada Kode Program 4.9.

```

public static void hasilPusatCluster(){

    for(int i=0;i<C; i++)
    {
        for(int j=0; j<jumlahAtribut; j++)
        {
            tempCenter[j][i]=center[j][i];
        }
    }
    center=new double[jumlahAtribut][C];
    for(int i=0;i<C; i++)
    {
        for(int j=0; j<jumlahAtribut; j++)
        {
            center[j][i]=tempCenter[j][i];
        }
    }
    System.out.println("=====Hasil          Akhir
Pusat
Cluster=====");
}

```

```

for(int i=0; i<C;i++)
{
    for(int j=0; j<jumlahAtribut;j++)
    {
        pusatCluster[i][j]=(center[j][i]*(xMax[j]-
        xMin[j]))+xMin[j];
    }
}
System.out.println("");
System.out.println("=====Hasil Perubahan
    Nilai Akhir Pusat Cluster=====");
}

```

Kode Program 4.9 Proses Menghitung Pusat *Cluster*

4.2.2.5. Proses Menghitung Sigma

Setelah pusat *cluster* diketahui adalah menghitung nilai sigma. Nilai sigma ini nantinya akan digunakan untuk menentukan nilai parameter fungsi keanggotaan. Kode program nilai sigma dapat digambarkan pada Kode Program 4.10.

```

public static void hitungSigma() {
    System.out.println("=====Hasil Perhitungan
    Sigma=====");
    for(int i=0; i<jumlahAtribut; i++)
    {
        SigmaIJ[i]=(jariJari*(xMax[i]-xMin[i]))/Math.pow(8,
        0.5);
        System.out.println(SigmaIJ[i]);
    }
}

```

Kode Program 4.10 Proses Menghitung Sigma

4.2.2.6. Proses Menghitung Derajat Keanggotaan

Setelah sigma diketahui maka selanjutnya proses menghitung derajat keanggotaan. Kode program proses menghitung derajat keanggotaan dapat digambarkan pada Kode Program 4.11.

```

public static void hitungDerajatKeanggotaan() {
    for(int i=0; i<C; i++) {
        for(int j=0; j<jumlahAtribut; j++) {
            for(int k=0; k<jumlahData; k++)
            {
                }}}
    int a=0,b=0;
}

```

```

for(int i=0;i<jumlahData;i++){
    for(int k=0;k<C;k++){
        for(int j=0;j<jumlahAtribut;j++){
            tempDerajatKeanggotaan[a][b]=derajatKeanggotaan[k][i][j];
            if(b==tempDerajatKeanggotaan[a].length-1){
                b=0;
                a++;
            }
            else b++;
        }
    }
}
System.out.println("=====Hasil
Perhitungan
Derajat Keanggotaan=====");
double tempCKij=0.0; int par=0;
for(int i=0; i<C; i++){
    for(int j=0; j<jumlahData; j++){
        for(int k=0; k<jumlahAtribut; k++){

tempCKij=tempCKij+derajatKeanggotaan[i][j][k];
            par++;
            if(par==5)
            {
                CKij[j][i]=tempCKij;
                tempCKij=0.0;
                par=0;
            }
        }
    }
}
System.out.println("=====Hasil
Perhitungan
CKij=====");
tempCKij=0.0; par=0;
for(int i=0; i<C; i++)
{
    for(int j=0; j<jumlahData; j++)
    {
        MiuKI[j][i]=Math.exp(-1*CKij[j][i]);
    }
}
System.out.println("=====Hasil
Perhitungan
Miu KI=====");
}

```

Kode Program 4.11 Proses Menghitung Derajat Keanggotaan

4.2.2.7. Proses Pengelompokkan Data

Proses pengelompokkan data merupakan proses untuk mengelompokkan data pada *cluster-cluster* berdasarkan nilai derajat keanggotaan. Kode program proses pengelompokkan data dapat digambarkan pada Kode Program 4.12.

```

public static void hasilCluster(){
    System.out.println("=====Hasil Pengclusteran
Data=====");
    double tempClusterMiu=0.0;
    for(int i=0; i<jumlahData; i++){
        tempClusterMiu=0.0;
        for(int j=0; j<C;j++){
            if(MiuKI[i][j]>tempClusterMiu) {
                tempClusterMiu=MiuKI[i][j];
                hasilCluster[i]=(j+1);
            }
        }
        System.out.println("Data Ke-"+i+" termasuk cluster
"+hasilCluster[i]);
    }
    for(int i=0; i<C; i++)
    {
        for(int j=0; j<jumlahData; j++)
        {
            if(tempHas[i]==null)
            {
                if(hasilCluster[j]==i+1)
                {
                    tempHas[i]="Data ke-"+(j+1)+",";
                }
            }else{
                if(hasilCluster[j]==i+1)
                {
                    tempHas[i]=tempHas[i]+"Data ke-
"+(j+1)+",";
                }
            }
        }
    }
    for(int i=0; i<C; i++)
    {
        textFS=textFS+"Data Anggota Cluster Ke-"+(i+1)+" :
";
        textFS=textFS+tempHas[i)+"\n";
    }
}

```

Kode Program 4.12 Proses Pengelompokkan Data

4.2.3. Tahapan Perbandingan

Pada tahapan ini dilakukan beberapa proses untuk membandingkan antara kedua metode dengan parameter *Partition Coefficient* (PC), *Modified Partition Coefficient* (MPC), dan *Clasification Entropy* (CE). Ketiga parameter tersebut diperoleh dengan menggunakan data derajat keanggotaan dari masing-masing metode. Proses yang dilakukan antara lain hitung PC pada FCM, hitung MPC

pada FCM, hitung CE pada FCM, hitung PC pada FSC, hitung MPC pada FSC, hitung CE pada FSC, dan proses hasil perbandingan dari FCM dan FSC.

4.2.3.1. Proses Hitung PC, MPC, CE pada FCM dan FSC

4.2.3.1.1. Proses Hitung PC

Pada proses ini bertujuan untuk menghitung salah satu parameter yang nantinya digunakan untuk membandingkan kedua metode. Pada proses ini dibutuhkan derajat keanggotaan. Proses ini berfungsi untuk mengukur jumlah cluster yang mengalami overlap. Kode program pada proses hitung PC pada FCM dan pada FSC dapat digambarkan pada Kode Program 4.13.

```

public static void hitungPC(){
    for(int i=0; i<jumlahData; i++){
        for(int j=0; j<jumlahClusterFcm; j++){
            miuPC[i][j]=Math.pow(miu[i][j],2);
        }
        System.out.println("====Data          Miu^2
PC=====");
        System.out.println("====Sigma Miu^2 PC====");
        for(int i=0; i<jumlahClusterFcm; i++)
        {
            for(int j=0; j<jumlahData; j++)
            {
                tempPC[i]=tempPC[i]+miuPC[j][i];
            }
            System.out.print(tempPC[i)+"\t");
        }
        for(int i=0; i<jumlahClusterFcm; i++)
        {
            PC=PC+tempPC[i];
        }
        PC=PC/jumlahData;
        System.out.println("====PC=====");
    }
}

```

Kode Program 4.13 Proses Hitung PC

4.2.3.1.2. Proses Hitung MPC

Setelah menghitung proses PC maka selanjutnya menghitung parameter yang digunakan yaitu MPC, dimana fungsi dari MPC tersebut adalah mengurangi perubahan yang monoton terhadap jumlah *cluster*. Kode program untuk proses

hitung MPC pada FCM dan pada FSC dapat digambarkan pada Kode Program 4.14.

```
public static void hitungMPC()
{
    MPC=1-((jumlahClusterFcm/jumlahClusterFcm-1)*(1-PC));
    System.out.println("");
    System.out.println("=====MPC=====");
    System.out.println(MPC);
}
}
```

Kode Program 4.14 Proses Hitung MPC

4.2.3.1.3. Proses Hitung CE

Pada proses selanjutnya adalah menghitung proses CE yang merupakan mengukur tingkat kekaburan (*fuzziness*) dari partisi *cluster*. Kode program untuk proses hitung CE pada FCM dan pada FSC dapat digambarkan pada Kode Program 4.15.

```
public static void hitungCE(){
    for (int i=0; i<jumlahData; i++)
    {
        for (int j=0; j<jumlahClusterFcm; j++){
            miuLogCE[i][j]=(miu[i][j])*(Math.log(miu[i][j])/Math.log(2));
        }
    }
    System.out.println("=====Data Miu * Log Miu Basis
2=====");
    System.out.println("=====Sigma Miu * Log Miu Basis
2=====");
    for (int i =0; i<jumlahClusterFcm; i++)
    {
        for(int j =0; j<jumlahData; j++)
        {
            tempMiuCE[i]=tempMiuCE[i]+miuLogCE[j][i];
        }
        System.out.print(tempMiuCE[i)+"\t");
    }
    double temCE=0.0;
```

```
for(int i=0; i<jumlahClusterFcm; i++)
{
    temCE=temCE+tempMiuCE[i];
}
CE=(1*(temCE/jumlahData));
System.out.println("=====CE=====");
}
```

Kode Program 4.15 Proses Hitung CE

4.3. Implementasi Antarmuka

Seperti yang sudah dijelaskan pada sub bab 3.3.3, bahwa sistem secara umum ada satu bagian utama. Sistem yang dibuat ini terdiri dari beberapa form tampilan untuk masing-masing metode, yaitu *input* parameter, pusat *cluster*, hasil nilai validitas *cluster*, data yang sudah ter-*cluster*, dan hasil perbandingan dari kedua metode.

4.3.1. Form Utama

Form utama yang dibuat merupakan bagian proses utama. Dimana untuk parameter dapat dilakukan inputan secara manual pada field text yang telah disediakan, dan terdapat *button* proses untuk dapat melihat hasil dari proses perbandingan kedua metode. Form utama serta bagian-bagiannya dari form dapat ditunjukkan pada Gambar 4.1.

The screenshot shows a software interface with the following sections:

- Title Fields:** Four input boxes labeled 'Title 1', 'Title 2', 'Title 3', and 'Title 4'.
- Parameter Fuzzy C-Means:** Includes 'Jumlah Cluster (C)', 'Max Iterasi', and 'Error (E)'. 'Max Iterasi' and 'Error (E)' are accompanied by spinners.
- Parameter Fuzzy Subtractive:** Includes 'Jari-Jari (r)', 'Squash Faktor', 'Accept Ratio', and 'Reject Ratio', each with a spinner.
- Validation Results:** Two sections on the right, one for 'Fuzzy C-Means' and one for 'Fuzzy Subtractive', each containing 'Partition Coefficient (PC)', 'Modified Partition Coefficient', and 'Classification Entropy (CE)' with spinners.
- Output Areas:** Two large text boxes for 'Pusat Cluster Yang Terbentuk Dari Fuzzy C-Means' and 'Pusat Cluster Yang Terbentuk Dari Fuzzy Subtractive'. Below them are two smaller text boxes for 'Data Tercluster Pada Hasil Proses Fuzzy C-Means' and 'Data Tercluster Pada Hasil Proses Fuzzy Subtractive'.
- Comparison Section:** A large text box at the bottom for 'Hasil Perbandingan Metode Fuzzy C-Means Clustering Dengan Fuzzy Subtractive'.
- Buttons:** 'Proses' and 'Reset' buttons are located at the bottom right.

Gambar 4.1 Form Utama

Proses untuk menjalankan dan melakukan proses *clustering* dan perbandingan. Proses bisa berjalan apabila semua parameter yang diinginkan sudah terisi semua. Setelah itu pengguna dapat menekan tombol proses. Setelah pengguna menekan tombol proses, maka pengguna dapat melihat hasil dari proses *clustering* dan perbandingan pada field text yang telah disediakan. Apabila proses berjalan dan kita menekan tombol proses maka akan muncul tampilan seperti pada Gambar 4.2.

Form Utama Aplikasi Perbandingan Fuzzy C-Means dengan Fuzzy Subtractive

Aplikasi Perbandingan Metode Fuzzy C-Means Dengan Fuzzy Subtractive

ID DATA	ID KATEGORI	UMUR PASIEN	LDL	KOLESTEROL TOTAL	HDL	TD SISTOLIK
92	K1	58	134	223	25	150
37	K1	70	264	226	44	140
76	K2	66	148	130	48	140
93	K2	51	79	189	34	140
78	K2	61	90	155	90	130
35	K2	58	79	143	50	100

Parameter Fuzzy C-Means

Jumlah Cluster (C) : 3 Max Iterasi : 50

Error (E) : 0.001

Parameter Fuzzy Subtractive

Jari-Jari (r) : 0.65 Squash Faktor : 1.25

Accept Ratio : 0.3 Reject Ratio : 0.1

Hasil Proses Validasi Fuzzy C-Means

Partition Coefficient (PC) : 0.548267

Modified Partition Coefficient : 1.000000

Classification Entropi (CE) : -1.136300

Hasil Proses Validasi Fuzzy Subtractive

Partition Coefficient (PC) : 0.365480

Modified Partition Coefficient : -0.903561

Classification Entropi (CE) : -0.576478

Pusat Cluster Yang Terbentuk Dari Fuzzy C-Means

69.00	180.97	222.21	46.41	135.36
58.81	92.56	157.32	60.99	123.71

Pusat Cluster Yang Terbentuk Dari Fuzzy Subtractive

53.00	135.00	185.00	38.00	170.00
61.00	90.00	155.00	90.00	130.00

Data Terkluster Pada Hasil Proses Fuzzy C-Means

Anggota Cluster Ke-1 : Data Ke-7, Data Ke-8, Data Ke-9

Anggota Cluster Ke-2 : Data Ke-1, Data Ke-2, Data Ke-3

Anggota Cluster Ke-3 : Data Ke-3, Data Ke-4, Data Ke-5

Data Terkluster Pada Hasil Proses Fuzzy Subtractive

Data Anggota Cluster Ke-1 : Data ke-2, Data ke-3, Data ke-4

Data Anggota Cluster Ke-2 : Data ke-1, Data ke-6, Data ke-7

Data Anggota Cluster Ke-3 : Data ke-5,

Hasil Perbandingan Metode Fuzzy C-Means Clustering Dengan Fuzzy Subtractive

Max Iterasi = 50

Error = 0.001

Lebih Baik dari FS.

Proses

Reset

Gambar 4.2 Tampilan Form Utama Setelah Proses Dilakukan

Setelah proses dilakukan selesai, maka terlihat seluruh hasil yang terbentuk. Untuk proses kali ini dapat kita lihat dari hasil proses validitas yang fungsinya untuk dilakukan perbandingan dengan melihat hasil tersebut. Hasil proses validitas dapat ditunjukkan pada Gambar 4.3.

Hasil Proses Validitas Fuzzy C-Means	
Partition Coefficient (PC) :	0.548367
Modified Partition Coefficient :	1.000000
Classification Entropi (CE) :	-1.136300

Hasil Proses Validitas Fuzzy Subtractive	
Partition Coefficient (PC) :	0.365480
Modified Partition Coefficient :	-0.903561
Classification Entropi (CE) :	-0.576478

Gambar 4.3 Tampilan Hasil Proses Validitas *Cluster* pada Form Utama

Proses validitas *cluster* telah diketahui maka bisa diketahui hasil perbandingan pada kasus PJK tersebut. Dengan membandingkan parameter dari proses validitas dari tiap metode, hasil perbandingan dapat ditunjukkan pada Gambar 4.4.

Hasil Perbandingan Metode Fuzzy C-Means Clustering Dengan Fuzzy Subtractive	
Max Iterasi = 50	
Error = 0.001	
Lebih Baik dari FS.	

Gambar 4.4 Tampilan Hasil Perbandingan pada Form Utama

4.3.2. Form Data Penyakit Jantung Koroner (PJK)

Form ini digunakan untuk melihat seluruh data Penyakit Jantung Koroner (PJK).

Form ini bisa juga sebagai media input data training dalam proses perhitungan.

Tampilan dari Form Data Penyakit Jantung Koroner (PJK) ditunjukkan pada

Gambar 4.5.

The screenshot shows a window titled "Form Data PJK" with a central table and input fields below it. The table is titled "DATA PENYAKIT JANTUNG KORONER" and contains 13 rows of patient data. Below the table are input fields for "Id Data Training", "Id Data", "Id Kategori", "Umur Pasien", "Ldl", "Kolesterol Total", "Hdl", and "Td Sistolik", along with a "Tambah" button.

ID DATA	ID KATEGORI	UMUR PASIEN	LDL	KOLESTEROL TOTAL	HDL	TD SI
1	K2	45	123	266	54	20
2	K2	67	167	255	39	14
3	K1	60	150	218	32	14
4	K2	55	119	167	35	14
5	K3	60	136	193	37	18
6	K3	61	98	256	33	15
7	K2	67	143	221	25	10
8	K2	57	115	218	66	15
9	K2	77	148	208	33	18
10	K2	67	94	182	75	16
11	K3	54	284	113	21	10
12	K1	53	160	200	39	15
13	K1	58	105	170	34	14

Id Data Training : Id Data : Id Kategori : Umur Pasien :

Ldl : Kolesterol Total : Hdl : Td Sistolik :

Gambar 4.5 Form Data Penyakit Jantung Koroner (PJK)

BAB V

ANALISA DAN PEMBAHASAN

5.1. Implementasi Uji Coba

5.5.1. Skenario Uji Coba

Dalam proses pengujian, nantinya scenario tersebut terdiri dari 3 kali uji coba dengan jumlah data training yang berbeda. Uji coba dilakukan terhadap beberapa data yang berbeda serta parameter yang berbeda, yang nantinya akan dihitung akurasi untuk hasil perbandingan kedua metode.

Pada proses pengujian, yang membedakan pada saat proses uji coba yaitu untuk uji coba pertama akan dilakukan proses uji coba untuk 10 data yang akan dilakukan 5 kali proses perulangan. Pada uji coba kedua, akan dilakukan uji coba data sebanyak 20 data, sama seperti pada uji coba pertama yang akan dilakukan proses perulangan sebanyak 5 kali. Begitupun juga pada uji coba ketiga, akan dilakukan uji coba sebanyak 30 data dan ini hanya dilakukan proses perulangan sebanyak 5 kali. Uji coba tersebut dilakukan pada kedua metode yaitu *Fuzzy C-Means Clustering* dan *Fuzzy Subtractive Clustering*, yang pada dasarnya pada proses perulangan nantinya akan dilakukan perbandingan antara kedua metode tersebut.

5.5.2. Hasil Uji Coba

5.5.2.1. Uji Coba 1

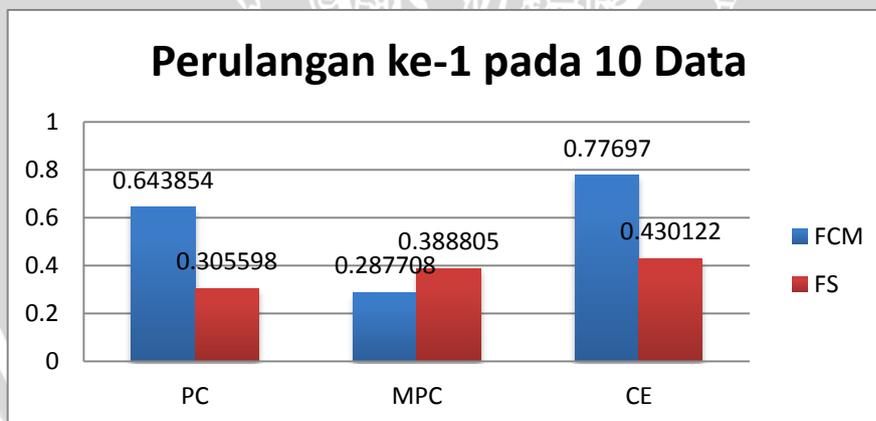
Pada uji coba 1, akan dilakukan proses uji coba sebanyak 10 data dan pada proses uji coba ini akan dilakukan juga proses perulangan 5 kali dengan 10 data yang berbeda yang fungsinya untuk mengetahui hasil perbandingan dari pada masing-masing metode.

Tabel 5.1 Tabel Perulangan ke-1 pada 10 Data

Perulangan Ke-1 pada 10 Data

Indeks	FCM	FS	Perbandingan
PC	0.643854	0.305598	FCM
MPC	0.287708	0.388805	FS
CE	0.77697	0.430122	FS

Dari Tabel 5.1 bisa kita lihat bahwa pada uji coba 10 data untuk perulangan ke-1 didapatkan nilai PC, MPC, dan CE, dimana pada perbandingan nilai PC dan MPC antara *Fuzzy C-Means Clustering* dengan *Fuzzy Subtractive Clustering* diambil nilai max (*maximum*) sedangkan pada nilai CE diambil nilai min (*minimum*) dan didapatkan kesimpulan 1 artinya bahwa *Fuzzy Subtractive Clustering* lebih optimal pada 10 data tersebut. Dengan *inputan* parameter pada masing-masing metode yaitu FCM ($c=3$; MaxIter=100; $E=0.001$) dan FS ($r=0.67$; *accept ratio*=0.6; *reject ratio*=0.1; terbentuk $c=4$). Untuk lebih jelasnya perbandingan ketiga indeks validitas dapat dilihat pada Gambar 5.1.



Gambar 5.1 Grafik Perulangan ke-1 pada 10 Data

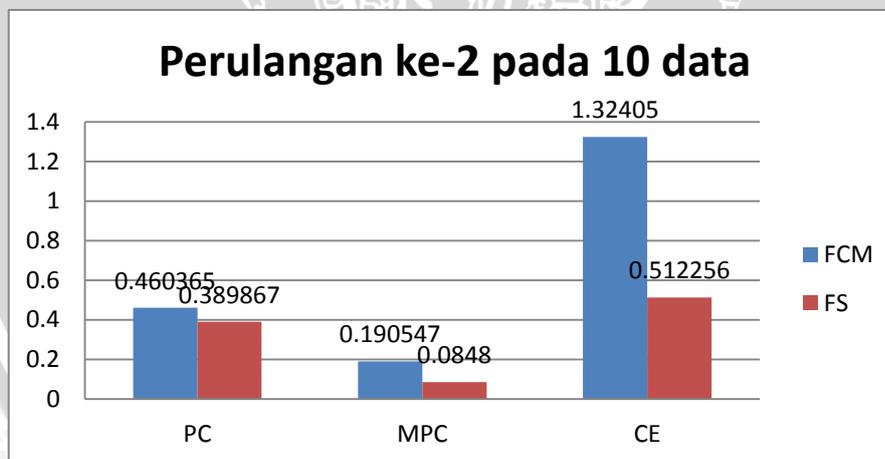
Dilihat dari grafik di atas, bisa dilihat bahwa nilai PC FCM dibandingkan dengan nilai PC FS dengan syarat diambil nilai yang paling besar adalah nilai PC pada FCM, untuk nilai MPC pada MPC FCM dan MPC FS nilai yang paling besar adalah nilai MPC pada FS, sedangkan untuk nilai CE dari CE FCM dan CE FS yang paling kecil adalah nilai CE pada FS.

Tabel 5.2 Tabel Perulangan ke-2 pada 10 Data

Perulangan Ke-2 pada 10 Data

Indeks	FCM	FS	Perbandingan
PC	0.460365	0.389867	FCM
MPC	0.190547	0.0848	FCM
CE	1.32405	0.512256	FS

Dari Tabel 5.2 bisa kita lihat bahwa pada uji coba 10 data untuk perulangan ke-2 didapatkan nilai PC, MPC, dan CE, dimana pada perbandingan nilai PC dan MPC antara *Fuzzy C-Means Clustering* dengan *Fuzzy Subtractive Clustering* diambil nilai max (*maximum*) sedangkan pada nilai CE diambil nilai min (*minimum*) dan didapatkan kesimpulan 0 artinya bahwa *Fuzzy C-Means Clustering* lebih optimal pada 10 data tersebut. Dengan *inputan* parameter pada masing-masing metode yaitu FCM ($c=3$; $\text{MaxIter}=50$; $E=0.000001$) dan FS ($r=0.6$; $\text{accept ratio}=0.4$; $\text{reject ratio}=0.1$; terbentuk $c=3$). Untuk lebih jelasnya perbandingan ketiga indeks validitas dapat dilihat pada Gambar 5.2.



Gambar 5.2 Grafik Perulangan ke-2 pada 10 Data

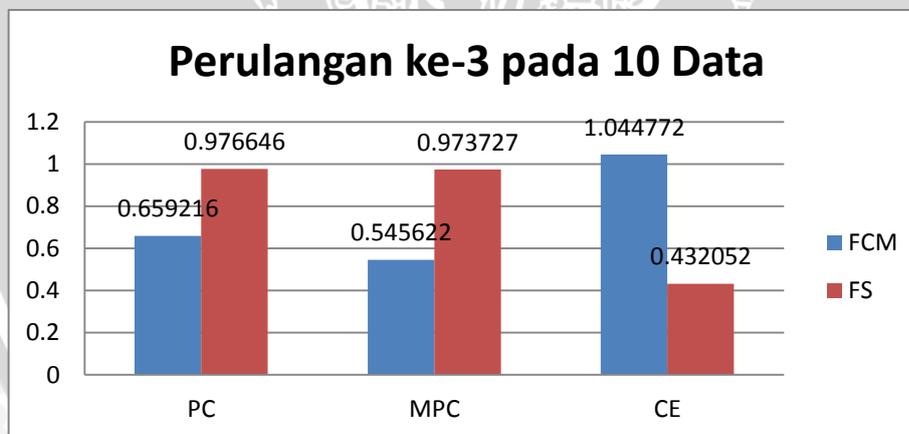
Dilihat dari grafik di atas, bisa dilihat bahwa nilai PC FCM dibandingkan dengan nilai PC FS dengan syarat diambil nilai yang paling besar adalah nilai PC pada FCM, untuk nilai MPC pada MPC FCM dan MPC FS nilai yang paling besar adalah nilai MPC pada FCM, sedangkan untuk nilai CE dari CE FCM dan CE FS yang paling kecil adalah nilai CE pada FS.

Tabel 5.3 Tabel Perulangan ke-3 pada 10 Data

Perulangan Ke-3 pada 10 Data

Indeks	FCM	FS	Perbandingan
PC	0.659216	0.976646	FS
MPC	0.545622	0.973727	FS
CE	1.044772	0.432052	FS

Dari Tabel 5.3 bisa kita lihat bahwa pada uji coba 10 data untuk perulangan ke-3 didapatkan nilai PC, MPC, dan CE, dimana pada perbandingan nilai PC dan MPC antara *Fuzzy C-Means Clustering* dengan *Fuzzy Subtractive Clustering* diambil nilai max (*maximum*) sedangkan pada nilai CE diambil nilai min (*minimum*) dan didapatkan kesimpulan 1 artinya bahwa *Fuzzy Subtractive Clustering* lebih optimal pada 10 data tersebut. Dengan *inputan* parameter pada masing-masing metode yaitu FCM ($c=4$; $\text{MaxIter}=50$; $E=0.001$) dan FS ($r=0.55$; $\text{accept ratio}=0.3$; $\text{reject ratio}=0.1$; terbentuk $c=4$). Untuk lebih jelasnya perbandingan ketiga indeks validitas dapat dilihat pada Gambar 5.3.



Gambar 5.3 Grafik Perulangan ke-3 pada 10 Data

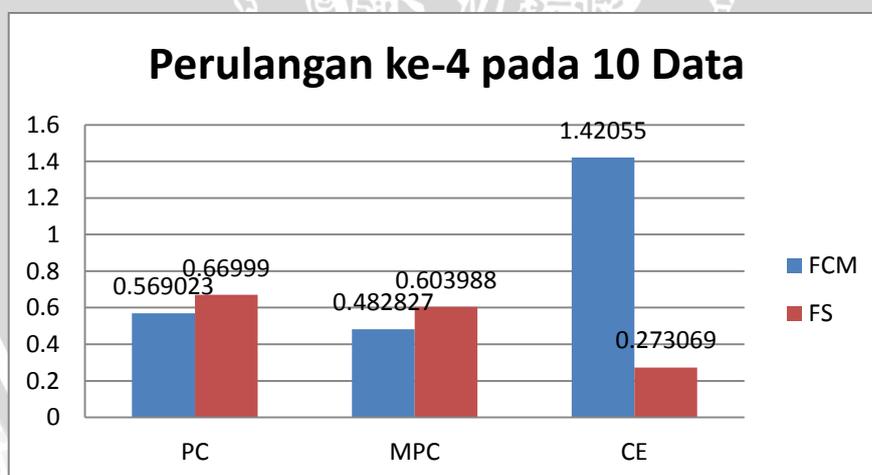
Dilihat dari grafik di atas, bisa dilihat bahwa nilai PC FCM dibandingkan dengan nilai PC FS dengan syarat diambil nilai yang paling besar adalah nilai PC pada FS, untuk nilai MPC pada MPC FCM dan MPC FS nilai yang paling besar adalah nilai MPC pada FS, sedangkan untuk nilai CE dari CE FCM dan CE FS yang paling kecil adalah nilai CE pada FS.

Tabel 5.4 Tabel Perulangan ke-4 pada 10 Data

Perulangan Ke-4 pada 10 Data

Indeks	FCM	FS	Perbandingan
PC	0.569023	0.66999	FS
MPC	0.482827	0.603988	FS
CE	1.42055	0.273069	FS

Dari Tabel 5.4 bisa kita lihat bahwa pada uji coba 10 data untuk perulangan ke-4 didapatkan nilai PC, MPC, dan CE, dimana pada perbandingan nilai PC dan MPC antara *Fuzzy C-Means Clustering* dengan *Fuzzy Subtractive Clustering* diambil nilai max (*maximum*) sedangkan pada nilai CE diambil nilai min (*minimum*) dan didapatkan kesimpulan 1 artinya bahwa *Fuzzy Subtractive Clustering* lebih optimal pada 10 data tersebut. Dengan *inputan* parameter pada masing-masing metode yaitu FCM ($c=6$; $\text{MaxIter}=100$; $E=0.00001$) dan FS ($r=0.37$; $\text{accept ratio}=0.7$; $\text{reject ratio}=0.5$; terbentuk $c=6$). Untuk lebih jelasnya perbandingan ketiga indeks validitas dapat dilihat pada Gambar 5.4.



Gambar 5.4 Grafik Perulangan ke-4 pada 10 Data

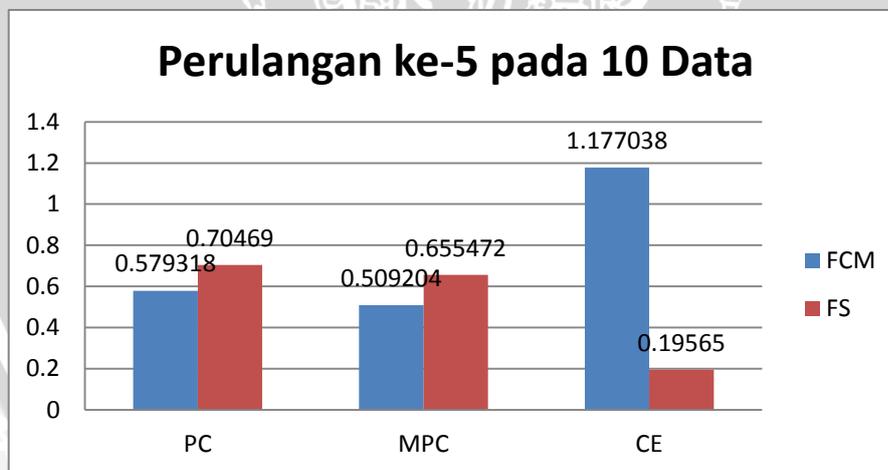
Dilihat dari grafik di atas, bisa dilihat bahwa nilai PC FCM dibandingkan dengan nilai PC FS dengan syarat diambil nilai yang paling besar adalah nilai PC pada FS, untuk nilai MPC pada MPC FCM dan MPC FS nilai yang paling besar adalah nilai MPC pada FS, sedangkan untuk nilai CE dari CE FCM dan CE FS yang paling kecil adalah nilai CE pada FS.

Tabel 5.5 Tabel Perulangan ke-5 pada 10 Data

Perulangan Ke-5 pada 10 Data

Indeks	FCM	FS	Perbandingan
PC	0.579318	0.70469	FS
MPC	0.509204	0.655472	FS
CE	1.177038	0.19565	FS

Dari Tabel 5.5 bisa kita lihat bahwa pada uji coba 10 data untuk perulangan ke-5 didapatkan nilai PC, MPC, dan CE, dimana pada perbandingan nilai PC dan MPC antara *Fuzzy C-Means Clustering* dengan *Fuzzy Subtractive Clustering* diambil nilai max (*maximum*) sedangkan pada nilai CE diambil nilai min (*minimum*) dan didapatkan kesimpulan 1 artinya bahwa *Fuzzy Subtractive Clustering* lebih optimal pada 10 data tersebut. Dengan *inputan* parameter pada masing-masing metode yaitu FCM ($c=7$; $\text{MaxIter}=100$; $E=0.0001$) dan FS ($r=0.3$; $\text{accept ratio}=0.6$; $\text{reject ratio}=0.1$; terbentuk $c=7$). Untuk lebih jelasnya perbandingan ketiga indeks validitas dapat dilihat pada Gambar 5.5.



Gambar 5.5 Grafik Perulangan ke-5 pada 10 Data

Dilihat dari grafik di atas, bisa dilihat bahwa nilai PC FCM dibandingkan dengan nilai PC FS dengan syarat diambil nilai yang paling besar adalah nilai PC pada FS, untuk nilai MPC pada MPC FCM dan MPC FS nilai yang paling besar adalah nilai MPC pada FS, sedangkan untuk nilai CE dari CE FCM dan CE FS yang paling kecil adalah nilai CE pada FS.

5.5.2.2. Uji Coba 2

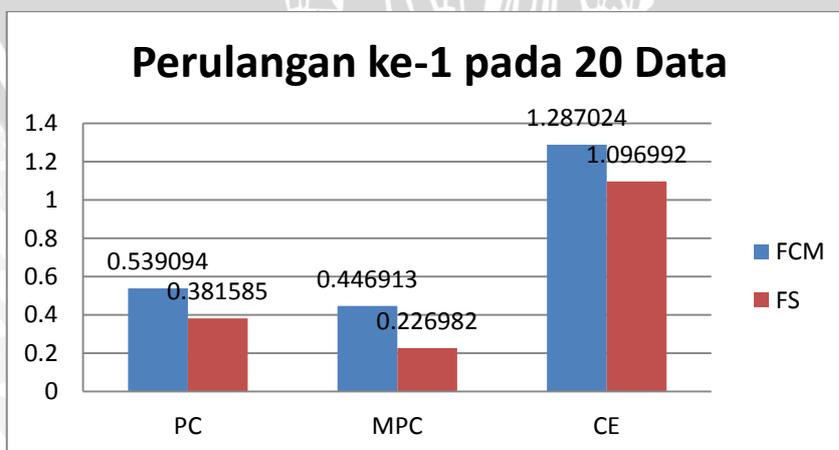
Pada uji coba 2, akan dilakukan proses uji coba sebanyak 20 data dan pada proses uji coba ini akan dilakukan juga proses perulangan 5 kali dengan 20 data yang berbeda yang fungsinya untuk mengetahui nilai pada masing-masing metode.

Tabel 5.6 Tabel Perulangan ke-1 pada 20 Data

Perulangan Ke-1 pada 20 Data

Indeks	FCM	FS	Perbandingan
PC	0.539094	0.381585	FCM
MPC	0.446913	0.226982	FCM
CE	1.287024	1.096992	FS

Dari Tabel 5.6 bisa kita lihat bahwa pada uji coba 20 data untuk perulangan ke-1 didapatkan nilai PC, MPC, dan CE, dimana pada perbandingan nilai PC dan MPC antara *Fuzzy C-Means Clustering* dengan *Fuzzy Subtractive Clustering* diambil nilai max (*maximum*) sedangkan pada nilai CE diambil nilai min (*minimum*) dan didapatkan kesimpulan 0 artinya bahwa *Fuzzy C-Means Clustering* lebih optimal pada 20 data tersebut. Dengan *inputan* parameter pada masing-masing metode yaitu FCM ($c=6$; $\text{MaxIter}=100$; $E=0.1$) dan FS ($r=0.8$; $\text{accept ratio}=0.5$; $\text{reject ratio}=0.4$; terbentuk $c=6$). Untuk lebih jelasnya perbandingan ketiga indeks validitas dapat dilihat pada Gambar 5.6.



Gambar 5.6 Grafik Perulangan ke-1 pada 20 Data

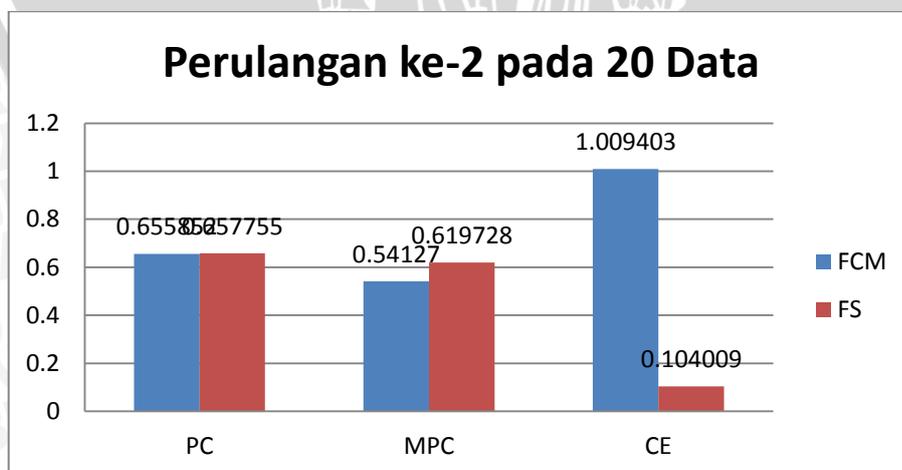
Dilihat dari grafik di atas, bisa dilihat bahwa nilai PC FCM dibandingkan dengan nilai PC FS dengan syarat diambil nilai yang paling besar adalah nilai PC pada FCM, untuk nilai MPC pada MPC FCM dan MPC FS nilai yang paling besar adalah nilai MPC pada FCM, sedangkan untuk nilai CE dari CE FCM dan CE FS yang paling kecil adalah nilai CE pada FS.

Tabel 5.7 Tabel Perulangan ke-2 pada 20 Data

Perulangan Ke-2 pada 20 Data

Indeks	FCM	FS	Perbandingan
PC	0.655852	0.657755	FS
MPC	0.54127	0.619728	FS
CE	1.009403	0.104009	FS

Dari Tabel 5.7 bisa kita lihat bahwa pada uji coba 20 data untuk perulangan ke-2 didapatkan nilai PC, MPC, dan CE, dimana pada perbandingan nilai PC dan MPC antara *Fuzzy C-Means Clustering* dengan *Fuzzy Subtractive Clustering* diambil nilai max (*maximum*) sedangkan pada nilai CE diambil nilai min (*minimum*) dan didapatkan kesimpulan 1 artinya bahwa *Fuzzy Subtractive Clustering* lebih optimal pada 20 data tersebut. Dengan *inputan* parameter pada masing-masing metode yaitu FCM ($c=4$; $MaxIter=100$; $E=0.001$) dan FS ($r=0.9$; $accept\ ratio=0.6$; $reject\ ratio=0.4$; terbentuk $c=4$). Untuk lebih jelasnya perbandingan ketiga indeks validitas dapat dilihat pada Gambar 5.7.



Gambar 5.7 Grafik Perulangan ke-2 pada 20 Data

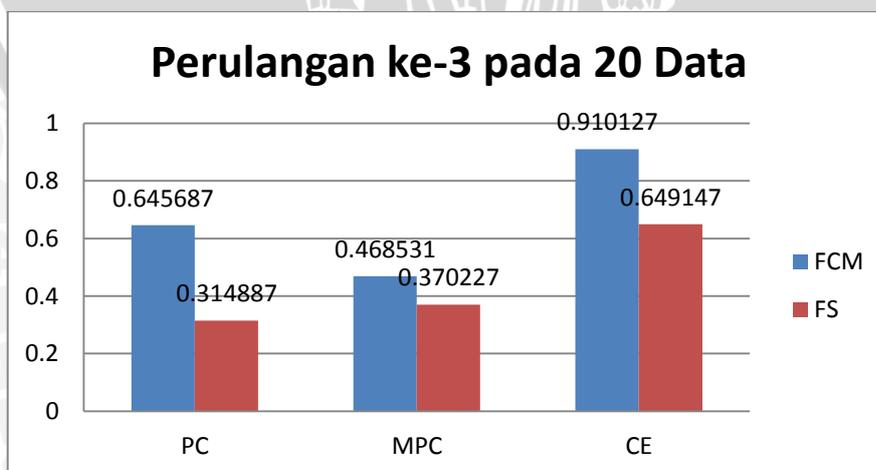
Dilihat dari grafik di atas, bisa dilihat bahwa nilai PC FCM dibandingkan dengan nilai PC FS dengan syarat diambil nilai yang paling besar adalah nilai PC pada FS, untuk nilai MPC pada MPC FCM dan MPC FS nilai yang paling besar adalah nilai MPC pada FS, sedangkan untuk nilai CE dari CE FCM dan CE FS yang paling kecil adalah nilai CE pada FS.

Tabel 5.8 Tabel Perulangan ke-3 pada 20 Data

Perulangan Ke-3 pada 20 Data

Indeks	FCM	FS	Perbandingan
PC	0.645687	0.314887	FCM
MPC	0.468531	0.370227	FCM
CE	0.910127	0.649147	FS

Dari Tabel 5.8 bisa kita lihat bahwa pada uji coba 20 data untuk perulangan ke-3 didapatkan nilai PC, MPC, dan CE, dimana pada perbandingan nilai PC dan MPC antara *Fuzzy C-Means Clustering* dengan *Fuzzy Subtractive Clustering* diambil nilai max (*maximum*) sedangkan pada nilai CE diambil nilai min (*minimum*) dan didapatkan kesimpulan 0 artinya bahwa *Fuzzy C-Means Clustering* lebih optimal pada 20 data tersebut. Dengan *inputan* parameter pada masing-masing metode yaitu FCM ($c=3$; MaxIter=100; $E=0.000001$) dan FS ($r=0.8$; *accept ratio*=0.6; *reject ratio*=0.2; terbentuk $c=3$). Untuk lebih jelasnya perbandingan ketiga indeks validitas dapat dilihat pada Gambar 5.8.



Gambar 5.8 Grafik Perulangan ke-3 pada 20 Data

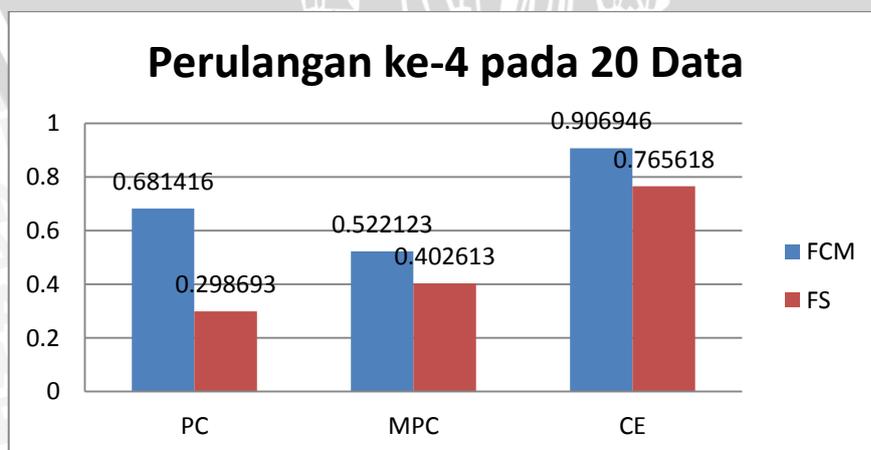
Dilihat dari grafik di atas, bisa dilihat bahwa nilai PC FCM dibandingkan dengan nilai PC FS dengan syarat diambil nilai yang paling besar adalah nilai PC pada FCM, untuk nilai MPC pada MPC FCM dan MPC FS nilai yang paling besar adalah nilai MPC pada FCM, sedangkan untuk nilai CE dari CE FCM dan CE FS yang paling kecil adalah nilai CE pada FS.

Tabel 5.9 Tabel Perulangan ke-4 pada 20 Data

Perulangan Ke-4 pada 20 Data

Indeks	FCM	FS	Perbandingan
PC	0.681416	0.298693	FCM
MPC	0.522123	0.402613	FCM
CE	0.906946	0.765618	FS

Dari Tabel 5.9 bisa kita lihat bahwa pada uji coba 20 data untuk perulangan ke-4 didapatkan nilai PC, MPC, dan CE, dimana pada perbandingan nilai PC dan MPC antara *Fuzzy C-Means Clustering* dengan *Fuzzy Subtractive Clustering* diambil nilai max (*maximum*) sedangkan pada nilai CE diambil nilai min (*minimum*) dan didapatkan kesimpulan 0 artinya bahwa *Fuzzy C-Means Clustering* lebih optimal pada 20 data tersebut. Dengan *inputan* parameter pada masing-masing metode yaitu FCM ($c=3$; MaxIter=100; $E=0.0001$) dan FS ($r=0.98$; $accept\ ratio=0.7$; $reject\ ratio=0.6$; terbentuk $c=3$). Untuk lebih jelasnya perbandingan ketiga indeks validitas dapat dilihat pada Gambar 5.9.



Gambar 5.9 Grafik Perulangan ke-4 pada 20 Data

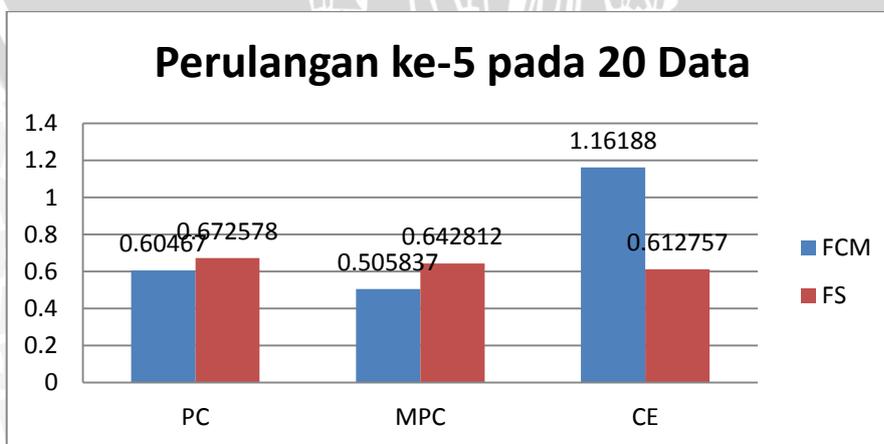
Dilihat dari grafik di atas, bisa dilihat bahwa nilai PC FCM dibandingkan dengan nilai PC FS dengan syarat diambil nilai yang paling besar adalah nilai PC pada FCM, untuk nilai MPC pada MPC FCM dan MPC FS nilai yang paling besar adalah nilai MPC pada FCM, sedangkan untuk nilai CE dari CE FCM dan CE FS yang paling kecil adalah nilai CE pada FS.

Tabel 5.10 Tabel Perulangan ke-5 pada 20 Data

Perulangan Ke-5 pada 20 Data

Indeks	FCM	FS	Perbandingan
PC	0.60467	0.672578	FS
MPC	0.505837	0.642812	FS
CE	1.16188	0.612757	FS

Dari Tabel 5.10 bisa kita lihat bahwa pada uji coba 20 data untuk perulangan ke-5 didapatkan nilai PC, MPC, dan CE, dimana pada perbandingan nilai PC dan MPC antara *Fuzzy C-Means Clustering* dengan *Fuzzy Subtractive Clustering* diambil nilai max (*maximum*) sedangkan pada nilai CE diambil nilai min (*minimum*) dan didapatkan kesimpulan 1 artinya bahwa *Fuzzy Subtractive Clustering* lebih optimal pada 20 data tersebut. Dengan *inputan* parameter pada masing-masing metode yaitu FCM ($c=3$; MaxIter=50; $E=0.1$) dan FS ($r=0.85$; *accept ratio*=0.5; *reject ratio*=0.4; terbentuk $c=3$). Untuk lebih jelasnya perbandingan ketiga indeks validitas dapat dilihat pada Gambar 5.10.



Gambar 5.10 Grafik Perulangan ke-5 pada 20 Data

Dilihat dari grafik di atas, bisa dilihat bahwa nilai PC FCM dibandingkan dengan nilai PC FS dengan syarat diambil nilai yang paling besar adalah nilai PC pada FS, untuk nilai MPC pada MPC FCM dan MPC FS nilai yang paling besar adalah nilai MPC pada FS, sedangkan untuk nilai CE dari CE FCM dan CE FS yang paling kecil adalah nilai CE pada FS.

5.5.2.3. Uji Coba 3

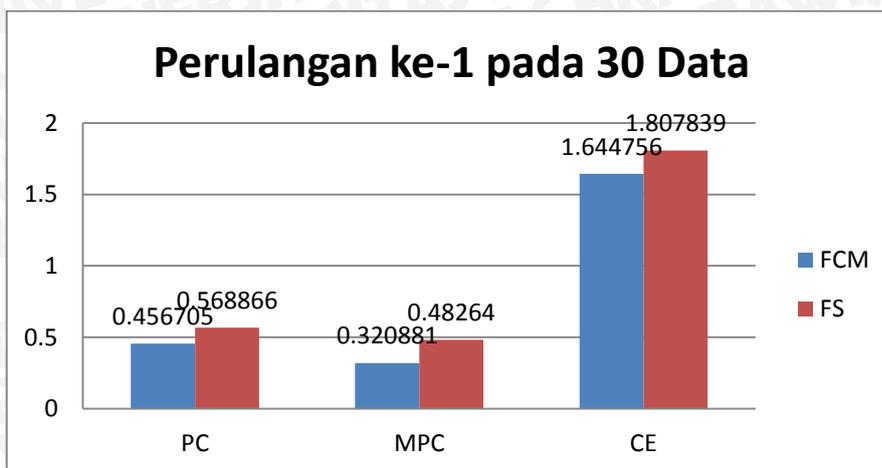
Pada uji coba 3, akan dilakukan proses uji coba sebanyak 30 data dan pada proses uji coba ini akan dilakukan juga proses perulangan 5 kali dengan 30 data yang berbeda yang fungsinya untuk mengetahui nilai pada masing-masing metode.

Tabel 5.11 Tabel Perulangan ke-1 pada 30 Data

Perulangan Ke-1 pada 30 Data

Indeks	FCM	FS	Perbandingan
PC	0.456705	0.568866	FS
MPC	0.320881	0.48264	FS
CE	1.644756	1.807839	FCM

Dari Tabel 5.11 bisa kita lihat bahwa pada uji coba 30 data untuk perulangan ke-1 didapatkan nilai PC, MPC, dan CE, dimana pada perbandingan nilai PC dan MPC antara *Fuzzy C-Means Clustering* dengan *Fuzzy Subtractive Clustering* diambil nilai max (*maximum*) sedangkan pada nilai CE diambil nilai min (*minimum*) dan didapatkan kesimpulan 1 artinya bahwa *Fuzzy Subtractive Clustering* lebih optimal pada 30 data tersebut. Dengan *inputan* parameter pada masing-masing metode yaitu FCM ($c=5$; $MaxIter=100$; $E=0.0001$) dan FS ($r=0.97$; $accept\ ratio=0.2$; $reject\ ratio=0.1$; terbentuk $c=5$). Untuk lebih jelasnya perbandingan ketiga indeks validitas dapat dilihat pada Gambar 5.11.



Gambar 5.11 Grafik Perulangan ke-1 pada 30 Data

Dilihat dari grafik di atas, bisa dilihat bahwa nilai PC FCM dibandingkan dengan nilai PC FS dengan syarat diambil nilai yang paling besar adalah nilai PC pada FS, untuk nilai MPC pada MPC FCM dan MPC FS nilai yang paling besar adalah nilai MPC pada FS, sedangkan untuk nilai CE dari CE FCM dan CE FS yang paling kecil adalah nilai CE pada FCM.

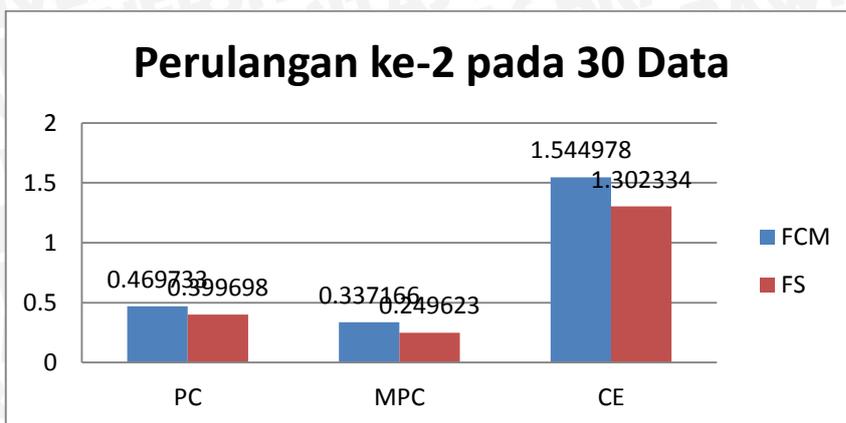
Tabel 5.12 Tabel Perulangan ke-2 pada 30 Data

Perulangan Ke-2 pada 30 Data

Indeks	FCM	FS	Perbandingan
PC	0.469733	0.399698	FCM
MPC	0.337166	0.249623	FCM
CE	1.544978	1.302334	FS

Dari Tabel 5.12 bisa kita lihat bahwa pada uji coba 30 data untuk perulangan ke-2 didapatkan nilai PC, MPC, dan CE, dimana pada perbandingan nilai PC dan MPC antara *Fuzzy C-Means Clustering* dengan *Fuzzy Subtractive Clustering* diambil nilai max (*maximum*) sedangkan pada nilai CE diambil nilai min (*minimum*) dan didapatkan kesimpulan 0 artinya bahwa *Fuzzy C-Means Clustering* lebih optimal pada 30 data tersebut. Dengan *inputan* parameter pada masing-masing metode yaitu FCM ($c=5$; $MaxIter=50$; $E=0.001$) dan FS ($r=0.9$; $accept\ ratio=0.5$; $reject\ ratio=0.3$; terbentuk $c=5$). Untuk lebih jelasnya perbandingan ketiga indeks validitas dapat dilihat pada Gambar 5.12.





Gambar 5.12 Grafik Perulangan ke-2 pada 30 Data

Dilihat dari grafik di atas, bisa dilihat bahwa nilai PC FCM dibandingkan dengan nilai PC FS dengan syarat diambil nilai yang paling besar adalah nilai PC pada FCM, untuk nilai MPC pada MPC FCM dan MPC FS nilai yang paling besar adalah nilai MPC pada FCM, sedangkan untuk nilai CE dari CE FCM dan CE FS yang paling kecil adalah nilai CE pada FS.

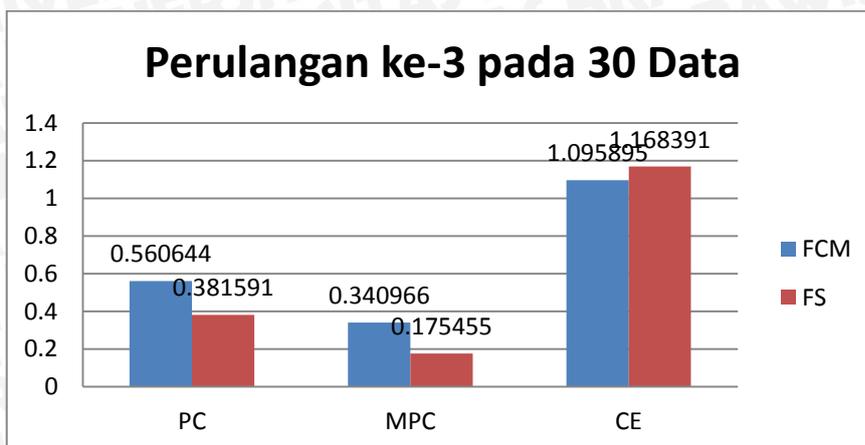
Tabel 5.13 Tabel Perulangan ke-3 pada 30 Data

Perulangan Ke-3 pada 30 Data

Indeks	FCM	FS	Perbandingan
PC	0.560644	0.381591	FCM
MPC	0.340966	0.175455	FCM
CE	1.095895	1.168391	FCM

Dari Tabel 5.13 bisa kita lihat bahwa pada uji coba 30 data untuk perulangan ke-3 didapatkan nilai PC, MPC, dan CE, dimana pada perbandingan nilai PC dan MPC antara *Fuzzy C-Means Clustering* dengan *Fuzzy Subtractive Clustering* diambil nilai max (*maximum*) sedangkan pada nilai CE diambil nilai min (*minimum*) dan didapatkan kesimpulan 0 artinya bahwa *Fuzzy C-Means Clustering* lebih optimal pada 30 data tersebut. Dengan *inputan* parameter pada masing-masing metode yaitu FCM ($c=3$; $MaxIter=100$; $E=0.00001$) dan FS ($r=0.96$; $accept\ ratio=0.6$; $reject\ ratio=0.1$; terbentuk $c=3$). Untuk lebih jelasnya perbandingan ketiga indeks validitas dapat dilihat pada Gambar 5.13.





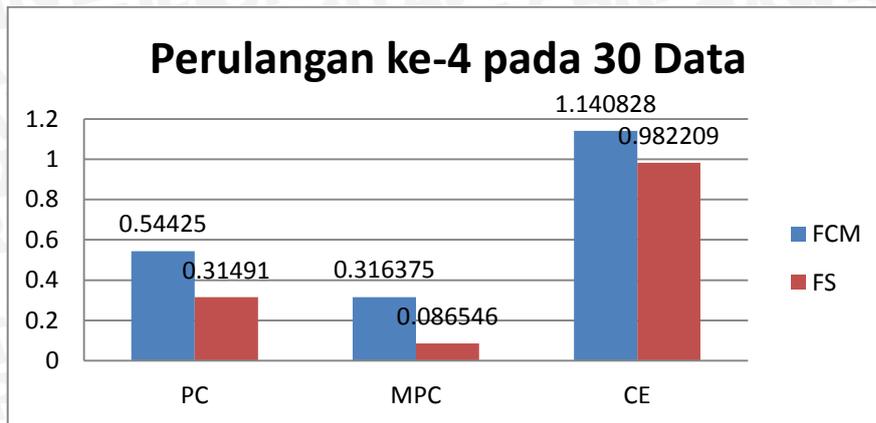
Gambar 5.13 Grafik Perulangan ke-3 pada 30 Data

Dilihat dari grafik di atas, bisa dilihat bahwa nilai PC FCM dibandingkan dengan nilai PC FS dengan syarat diambil nilai yang paling besar adalah nilai PC pada FCM, untuk nilai MPC pada MPC FCM dan MPC FS nilai yang paling besar adalah nilai MPC pada FCM, sedangkan untuk nilai CE dari CE FCM dan CE FS yang paling kecil adalah nilai CE pada FCM.

Tabel 5.14 Tabel Perulangan ke-4 pada 30 Data

Perulangan Ke-4 pada 30 Data			
Indeks	FCM	FS	Perbandingan
PC	0.54425	0.31491	FCM
MPC	0.316375	0.086546	FCM
CE	1.140828	0.982209	FS

Dari Tabel 5.14 bisa kita lihat bahwa pada uji coba 30 data untuk perulangan ke-4 didapatkan nilai PC, MPC, dan CE, dimana pada perbandingan nilai PC dan MPC antara *Fuzzy C-Means Clustering* dengan *Fuzzy Subtractive Clustering* diambil nilai max (*maximum*) sedangkan pada nilai CE diambil nilai min (*minimum*) dan didapatkan kesimpulan 0 artinya bahwa *Fuzzy C-Means Clustering* lebih optimal pada 30 data tersebut. Dengan *inputan* parameter pada masing-masing metode yaitu FCM ($c=3$; $MaxIter=50$; $E=0.001$) dan FS ($r=0.87$; $accept\ ratio=0.7$; $reject\ ratio=0.68$; terbentuk $c=3$). Untuk lebih jelasnya perbandingan ketiga indeks validitas dapat dilihat pada Gambar 5.14.



Gambar 5.14 Grafik Perulangan ke-4 pada 30 Data

Dilihat dari grafik di atas, bisa dilihat bahwa nilai PC FCM dibandingkan dengan nilai PC FS dengan syarat diambil nilai yang paling besar adalah nilai PC pada FCM, untuk nilai MPC pada MPC FCM dan MPC FS nilai yang paling besar adalah nilai MPC pada FCM, sedangkan untuk nilai CE dari CE FCM dan CE FS yang paling kecil adalah nilai CE pada FS.

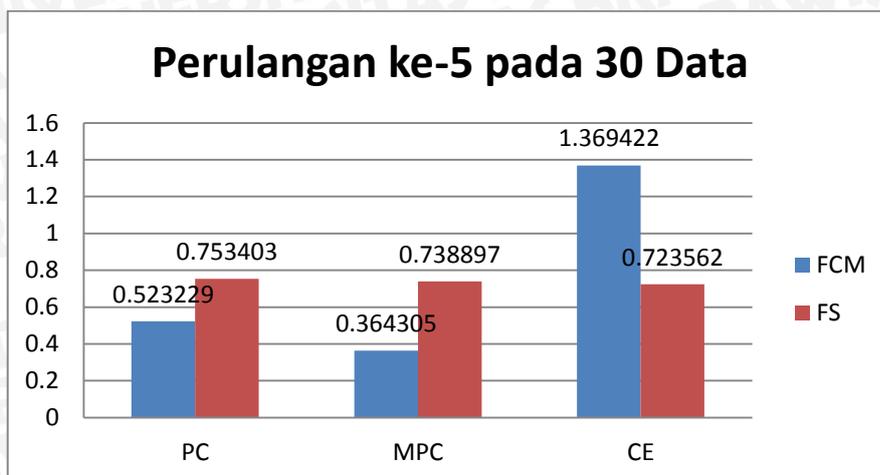
Tabel 5.15 Tabel Perulangan ke-5 pada 30 Data

Perulangan Ke-5 pada 30 Data

Indeks	FCM	FS	Perbandingan
PC	0.523229	0.753403	FS
MPC	0.364305	0.738897	FS
CE	1.369422	0.723562	FS

Dari Tabel 5.15 bisa kita lihat bahwa pada uji coba 30 data untuk perulangan ke-5 didapatkan nilai PC, MPC, dan CE, dimana pada perbandingan nilai PC dan MPC antara *Fuzzy C-Means Clustering* dengan *Fuzzy Subtractive Clustering* diambil nilai max (*maximum*) sedangkan pada nilai CE diambil nilai min (*minimum*) dan didapatkan kesimpulan 1 artinya bahwa *Fuzzy Subtractive Clustering* lebih optimal pada 30 data tersebut. Dengan *inputan* parameter pada masing-masing metode yaitu FCM ($c=3$; $MaxIter=50$; $E=0.000001$) dan FS ($r=0.87$; $accept\ ratio=0.8$; $reject\ ratio=0.68$; terbentuk $c=3$). Untuk lebih jelasnya perbandingan ketiga indeks validitas dapat dilihat pada Gambar 5.15.





Gambar 5.15 Grafik Perulangan ke-5 pada 30 Data

Dilihat dari grafik di atas, bisa dilihat bahwa nilai PC FCM dibandingkan dengan nilai PC FS dengan syarat diambil nilai yang paling besar adalah nilai PC pada FS, untuk nilai MPC pada MPC FCM dan MPC FS nilai yang paling besar adalah nilai MPC pada FS, sedangkan untuk nilai CE dari CE FCM dan CE FS yang paling kecil adalah nilai CE pada FS.

5.2. Analisa Hasil

Seperti yang telah dilakukan pada proses pengujian data, didapatkan hasil perbandingan dari masing-masing uji coba menggunakan 10 data, 20 data, dan 30 data. Untuk masing-masing percobaan dilakukan sebanyak 5 kali perulangan dan pada setiap perulangan akan dibandingkan hasil indeks validitasnya. Pada analisa hasil berikut akan dibahas hasil perbandingan antara *Fuzzy C-Means Clustering* dan *Fuzzy Subtractive Clustering* pada kasus penyakit jantung coroner (PJK). Ditunjukkan pada Tabel 5.16 dan Gambar 5.16.

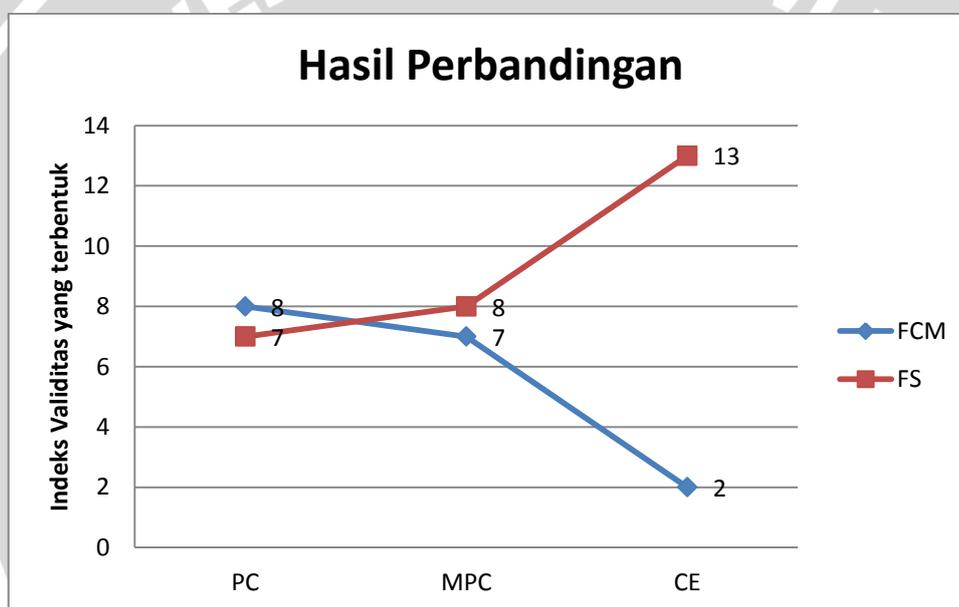
Tabel 5.16 Tabel Hasil Perbandingan

Hasil Perbandingan

Data	Perulangan ke-	Indeks Validitas		
		PC	MPC	CE
10 data	1	FCM	FS	FS
	2	FCM	FCM	FS
	3	FS	FS	FS
	4	FS	FS	FS
	5	FS	FS	FS
20 data	1	FCM	FCM	FS
	2	FS	FS	FS
	3	FCM	FCM	FS
	4	FCM	FCM	FS
	5	FS	FS	FS
30 data	1	FS	FS	FCM
	2	FCM	FCM	FS
	3	FCM	FCM	FCM
	4	FCM	FCM	FS
	5	FS	FS	FS

Pada Tabel 5.16 di atas merupakan hasil dari perbandingan nilai indeks validitas antara metode *Fuzzy C-Means Clustering* dan *Fuzzy Subtractive Clustering*. Saat melakukan proses perbandingan, jumlah cluster dari kedua metode tersebut harus sama, jika tidak bisa mendapatkan jumlah hasil *cluster* yang sama, selisih dari keduanya tidak boleh beda jauh. Dan untuk melakukan proses persamaan jumlah *cluster* terutama pada metode FS, dapat merubah parameter jari-jari (r). Pada metode FS ini jumlah *cluster* harus disamakan dengan metode FCM karena pada jumlah *cluster* pada metode FS terbentuk setelah proses iterasi berhenti, sedangkan pada metode FCM jumlah *cluster* sudah ditentukan dari awal. Pada uji coba 10 data dapat dilihat bahwa nilai PC banyak terdapat pada metode FS, untuk nilai MPC banyak terdapat pada metode FS, dan untuk nilai CE banyak terdapat pada metode FS juga. Lalu kita lihat pada uji coba 20 data, dimana nilai PC banyak terdapat pada metode FCM, untuk nilai MPC banyak terdapat pada metode FCM, dan untuk CE banyak terdapat pada metode FS. Selanjutnya dapat kita lihat lagi dari uji coba 30 data, dari uji coba tersebut

didapatkan nilai PC yang terbanyak adalah pada metode FCM, nilai MPC banyak terdapat pada metode FCM, dan untuk nilai CE banyak terdapat pada metode FS. Dari sini kita belum bisa menentukan metode mana yang terbaik karena mungkin saja bisa terjadi kesamaan jumlah dalam nilai indeks validitas pada kedua metode tersebut. Untuk mengetahui hasil perbandingan antara *Fuzzy C-Means Clustering* dan *Fuzzy Subtractive Clustering*, kita dapat melihat dari perbandingan jumlah nilai indeks validitas pada kedua metode terutama pada indeks CE, karena CE berfungsi untuk membantu pengambilan suatu keputusan dari suatu masalah. Untuk lebih jelas dapat dilihat pada Gambar 5.16.



Gambar 5.16 Grafik Hasil Perbandingan

Dari keseluruhan uji coba data penyakit jantung coroner didapatkan grafik seperti di atas, hasil perbandingan PC pada metode FCM adalah 8, sedangkan hasil perbandingan PC pada metode FS adalah 7. Lalu untuk hasil perbandingan MPC pada metode FCM adalah 7 dan pada metode FS adalah 8. Dan selanjutnya hasil perbandingan CE pada metode FCM adalah 2 dan pada metode FS hasil perbandingan CE adalah 13. Untuk menyimpulkan bahwa dari salah satu kedua metode tersebut itu merupakan metode yang bagus digunakan untuk melakukan proses *clustering* adalah dengan melihat hasil perbandingan CE,

sedangkan untuk menentukan nilai CE diambil nilai indeks validitasnya yang terkecil.

Nilai PC pada metode FCM sedikit mengalami overlap karena semakin besar jumlah PC yang didapat maka jumlah *cluster* akan sedikit mengalami overlap, sedangkan semakin kecil jumlah PC maka jumlah *cluster* akan banyak mengalami overlap. Pada nilai MPC semakin besar jumlah MPC maka proses mengurangi perubahan jumlah *cluster* yang beragam sangat bagus. Sedangkan untuk nilai CE ini berbeda dengan nilai PC dan MPC karena nilai CE membantu pengambilan keputusan dalam suatu masalah. Jadi semakin kecil nilai CE maka tingkat kekaburan (*fuzziness*) semakin efektif, sehingga akan diketahui hasil perbandingan antara *Fuzzy C-Means Clustering* dan *Fuzzy Subtractive Clustering* pada kasus Risiko Penyakit Jantung Koroner (PJK) adalah metode *Fuzzy C-Means Clustering*.



BAB VI PENUTUP

6.1 Kesimpulan

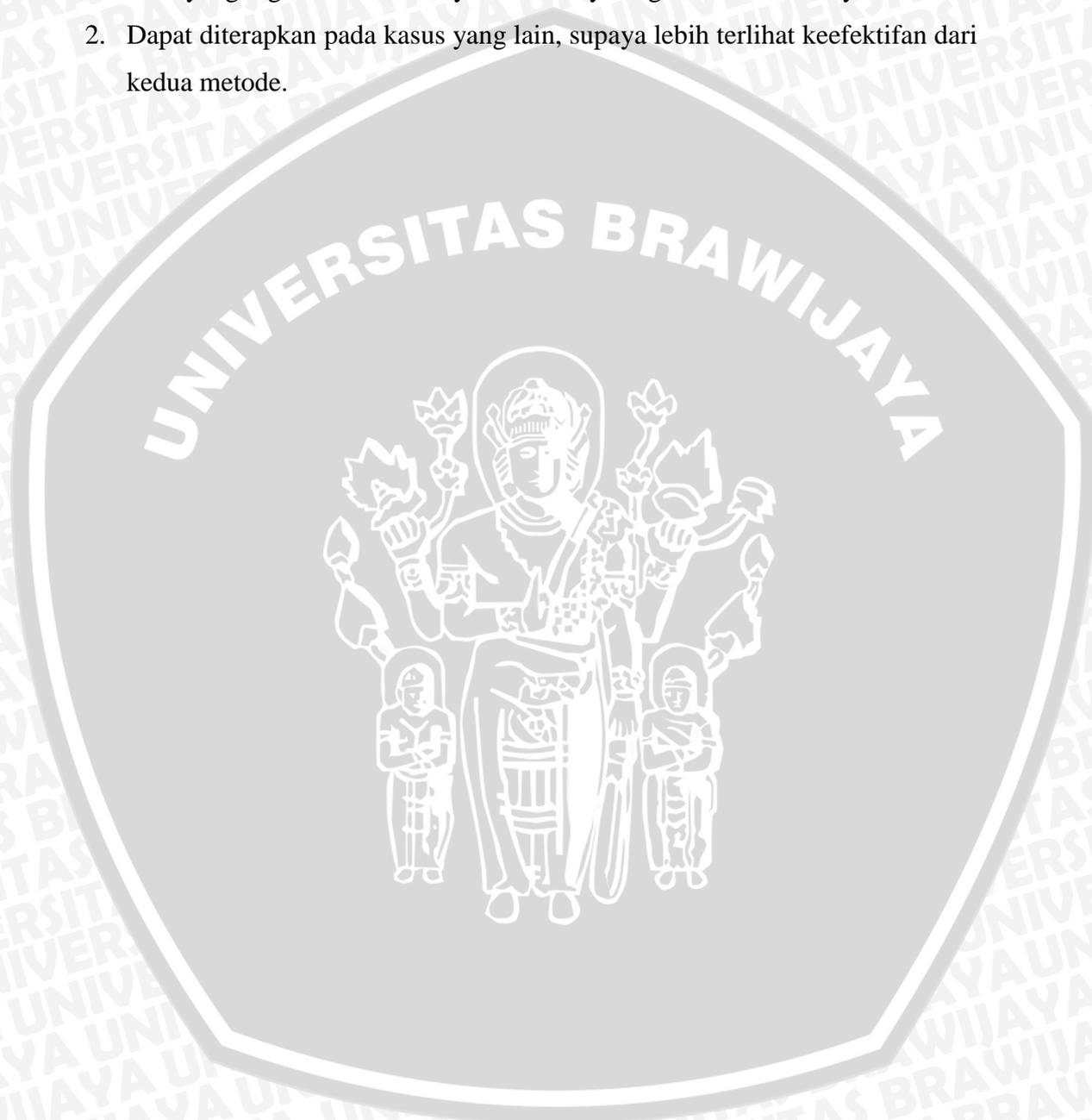
Berdasarkan hasil dari penelitian tentang perbandingan *Fuzzy C-Means Clustering* dan *Fuzzy Subtractive Clustering* pada kasus risiko Penyakit Jantung Koroner (PJK), dapat diambil kesimpulan :

1. Keefektifan *Fuzzy C-Means Clustering* pada kasus risiko penyakit jantung coroner (PJK) yang dilihat dari perbandingan indeks validitasnya yaitu nilai $PC = 8$; $MPC = 7$; $CE = 2$. Hasil perbandingan CE pada metode tersebut kecil maka tingkat kekaburan (*fuzziness*) pada metode rendah sehingga bagus digunakan untuk proses *clustering*.
2. Keefektifan *Fuzzy Subtractive Clustering* pada kasus risiko penyakit jantung coroner (PJK) yang dilihat dari perbandingan indeks validitasnya yaitu nilai $PC = 7$; $MPC = 8$; $CE = 13$. Hasil perbandingan CE pada metode tersebut sangat besar dibandingkan nilai CE pada metode FCM maka tingkat kekaburan (*fuzziness*) pada metode ini sangat tinggi sehingga kurang bagus jika digunakan untuk proses *clustering*.
3. Keefektifan kedua metode pada kasus risiko penyakit jantung coroner (PJK) jika dilihat dari hasil perbandingan indeks validitas kedua metode antara *Fuzzy C-Means Clustering* dan *Fuzzy Subtractive Clustering*. *Fuzzy C-Means Clustering* merupakan metode yang bagus digunakan untuk proses *clustering* karena semakin kecil nilai CE maka semakin bagus metode tersebut digunakan untuk proses *clustering*.

6.2 Saran

Saran yang bisa diberikan dari hasil penelitian yang digunakan sebagai bahan pengembangan penelitian yang terkait dengan penelitian ini, yaitu :

1. Data yang digunakan sebaiknya lebih banyak agar sebaran datanya akurat.
2. Dapat diterapkan pada kasus yang lain, supaya lebih terlihat keefektifan dari kedua metode.



Article I. DAFTAR PUSTAKA

- [ANW-04] Anwar, T., Bahri. 2004. "Penyakit Jantung Koroner dan Hipertensi".
<http://library.usu.ac.id/download/fk/gizi-bahri10.pdf> [9 February 2013].
- [DAV-03] Davidson, C., 2003, "Seri Kesehatan: Bimbingan Dokter pada Penyakit Jantung Koroner, Jakarta: Dian Rakyat.
- [ENA-08] Effendy, Nazrul, Subagja, dan Amir Faisal. 2008. "Prediksi Penyakit Jantung Koroner (PJK) Berdasarkan Faktor Risiko Menggunakan Jaringan Saraf Tiruan *Backpropagation*".
<http://journal.uui.ac.id/index.php/Snati/article/viewFile/725/680> [9 Februari 2013].
- [GAN-09] Gandha, Nico., 2009, "Hubungan Perilaku Literatur",
<http://www.lontar.ui.ac.id/file?file=digital/122845-S09038fk...Literatur.pdf> [2 mei 2013]
- [KOC-02] Koczy. 2002. "Fuzzy Rule Base Model Identification Techniques". Department of Telecommunication and Telematics, Budapest University of Technology and Economics. Hungary.
- [KOM-09] Kompas.com, 2009, "Awas, Jantung Koroner Mengintai Anda",
<http://kesehatan.kompas.com/read/2009/05/22/21044425/Awas.Jantung.Koroner.Mengintai.Anda> [13 Februari 2013].
- [KRI-02] Krismi, Arum. 2002. "Perbedaan Faktor Risiko Primer Pada Penderita Penyakit Jantung Koroner dan Kontrol di RSUP Dr. Sardjito Yogyakarta". Skripsi. Yogyakarta.
- [KUH-10] Kusumadewi, Sri., Hari Purnomo. 2010. "Aplikasi Logika Fuzzy Untuk Pendukung Keputusan". Jakarta: Graha Ilmu.
- [MAJ-07] Majid, Abdul, 2007, "Penyakit Jantung Koroner: Patofisiologi, Pencegahan, dan Pengobatan Terkini", Pidato Pengukuhan Jabatan Guru Besar Tetap Fakultas Kedokteran, USU.

- [NUG-11] Nugraheni, Yohana. 2011. "Data Mining Dengan Metode *Fuzzy* Untuk *Customer Relationship Management* (CRM) Pada Perusahaan Retail".
http://www.pps.unud.ac.id/thesis/pdf_thesis/unud-212-866836442-thesis_yohana.pdf [13 Februari 2013].
- [PRA-05] Priyono, Agus, dkk. 2005. "*Generation of Fuzzy Rules with Subtractive Clustering*". Jurnal Teknologi, 43 (D) Dis: 143-153. Malaysia.
- [SID-10] Sidhu, Ramandeep, S. 2010. "*A Subtractive Clustering Based Approach for Early Prediction of Fault Pronenes in Software Modules*". World Academy of Science, Engineering and Technology 67. India.
- [SOE-10] Soeharto, Imam. 2010. "Penyakit Jantung Koroner (PJK): Sebab, Mekanisme, dan Gejala".
<http://www.fkunhas.com/2010/07/penyakit-jantung-koroner-pjk-sebab.html> [9 Februari 2013].
- [WAH-11] Wahyuni, Kristin. 2011. "Diagnosis Penyakit Jantung Koroner (PJK) Berdasarkan Faktor Risiko Menggunakan Metode FCM". Skripsi. Universitas Brawijaya Malang.
- [WUY-05] Wu, K., Yang M. 2005. "*A Cluster Validity Index for Fuzzy Clustering*". Pattern Recognition Letter 26, pp. 1275-1291.

Lampiran

DATA PENYAKIT JANTUNG KORONER (PJK)

No.	Faktor PJK				
	Umur	LDL	Kol. Tot.	HDL	TD Sistolik
1	45	123	266	54	200
2	67	167	255	39	140
3	60	150	218	32	140
4	55	119	167	35	140
5	60	136	193	37	180
6	61	98	256	33	150
7	67	143	221	25	130
8	57	115	218	66	150
9	77	148	208	33	180
10	67	94	182	75	160
11	54	284	113	21	120
12	53	160	200	39	150
13	58	105	170	34	140
14	77	186	168	66	130
15	60	116	170	40	180
16	65	137	214	13	140
17	50	164	243	39	160
18	51	196	200	72	140
19	62	178	215	35	120
20	56	214	301	61	150
21	65	192	166	58	170
22	63	143	214	49	130
23	69	238	335	47	130
24	53	135	185	38	170
25	54	299	158	48	200
26	66	151	229	60	180
27	58	160	239	38	180
28	63	151	234	50	130
29	68	256	221	48	140
30	49	204	186	55	160
31	66	190	165	62	170
32	35	380	350	30	180
33	68	291	145	38	180
34	70	161	201	38	170
35	58	154	230	37	140
36	67	196	302	70	170
37	70	264	226	44	140

38	54	178	262	54	160
39	59	149	172	52	180
40	39	198	189	34	130
41	61	167	231	39	130
42	60	148	266	39	160
43	59	176	245	44	150
44	56	145	187	41	130
45	57	285	132	30	140
46	65	158	229	30	150
47	49	146	221	44	170
48	67	280	232	47	140
49	68	153	201	39	180
50	63	292	166	58	160
51	75	190	148	42	150
52	60	171	182	57	140
53	65	183	248	36	140
54	63	299	159	47	160
55	62	113	183	43	170
56	50	166	236	35	130
57	47	280	245	30	180
58	46	292	152	40	120
59	70	186	176	33	170
60	70	254	280	30	160
61	63	152	294	60	150
62	47	192	171	41	120
63	41	111	165	47	180
64	32	192	167	40	170
65	67	191	196	53	140
66	68	138	212	37	180
67	72	163	255	45	180
68	41	185	208	50	130
69	55	154	253	41	150
70	38	119	289	41	140
71	62	158	269	59	200
72	74	283	145	32	160
73	58	269	133	30	170
74	44	184	325	40	170
75	51	104	390	53	150
76	66	148	130	48	140
77	38	195	270	35	130
78	61	90	155	90	130
79	55	190	160	39	120

80	70	189	180	58	180
81	56	198	173	38	200
82	63	242	196	34	150
83	73	170	149	44	140
84	50	299	215	26	170
85	49	289	212	26	170
86	56	266	131	30	170
87	60	154	259	58	160
88	71	105	218	51	160
89	82	90	164	40	170
90	43	195	181	60	170
91	50	324	290	32	170
92	58	134	223	25	150
93	51	79	189	34	140
94	59	175	153	52	160
95	67	209	194	48	160
96	72	108	213	48	180
97	63	152	239	45	160
98	68	123	232	45	170
99	68	140	220	37	130
100	44	174	203	47	160

