

**PREDIKSI TINGKAT RISIKO PENYAKIT JANTUNG
KORONER (PJK) MENGGUNAKAN METODE
FUZZY K-NEAREST NEIGHBOR (FK-NN)**

SKRIPSI

Sebagai salah satu syarat untuk memperoleh
gelar Sarjana dalam bidang Ilmu Komputer



Disusun oleh :
MASLIKHA PUSPASARI
NIM. 0810963013

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

MALANG

2013

LEMBAR PERSETUJUAN

PREDIKSI TINGKAT RISIKO PENYAKIT JANTUNG KORONER (PJK) MENGUNAKAN METODE *FUZZY K-NEAREST NEIGHBOR* (FK-NN)

SKRIPSI

Sebagai salah satu syarat untuk memperoleh
gelar Sarjana dalam bidang Ilmu Komputer



Disusun oleh :
MASLIKHA PUSPASARI
NIM. 0810963013

Telah diperiksa dan disetujui oleh:

Pembimbing I,

Pembimbing II,

Candra Dewi, S.Kom., MSc

NIP. 197711142003122001

Drs. Muh Arif Rahman, M.Kom

NIP. 196604231991111001

LEMBAR PENGESAHAN

**PREDIKSI TINGKAT RISIKO PENYAKIT JANTUNG KORONER (PJK)
MENGUNAKAN METODE *FUZZY K-NEAREST NEIGHBOR* (FK-NN)**

SKRIPSI

Sebagai salah satu syarat untuk memperoleh
gelar Sarjana dalam bidang Ilmu Komputer

Disusun Oleh:

MASLIKHA PUSPASARI

NIM.0810963013

Skripsi ini telah diuji dan dinyatakan lulus pada tanggal 14 Januari 2013

Penguji I

Penguji II

Dian Eka Ratnawati, S.Si., M.Kom
NIP. 197306192002122001

Suprpto, S.T., M.T.
NIP. 197107271966031001

Penguji III

Ahmad Afif Supianto, S.Si., M.Kom

Mengetahui
Ketua Program Studi Teknik Informatika

Drs. Marji, M.T.
NIP. 196708011992031001

PERNYATAAN ORISINALITAS SKRIPSI

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah SKRIPSI ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila ternyata didalam naskah SKRIPSI ini dapat dibuktikan terdapat unsur-unsur PLAGIASI, saya bersedia SKRIPSI ini digugurkan dan gelar akademik yang telah saya peroleh (SARJANA) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku. (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, Januari 2013

Mahasiswa,

Maslikha Puspasari
NIM. 0810963013

KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Allah SWT yang telah melimpahkan segala Rahmat, Karunia dan Hidayah-Nya sehingga Penulis dapat menyelesaikan skripsi dengan judul: **"Prediksi Tingkat Risiko Penyakit Jantung Koroner (PJK) Menggunakan Metode Fuzzy K - Nearest Neighbor (FK-NN)"**

Skripsi ini diajukan sebagai syarat ujian seminar skripsi dalam rangka untuk memperoleh gelar Sarjana Komputer di Program Studi Informatika dan Ilmu Komputer, Universitas Brawijaya Malang. Atas terselesaikannya skripsi ini, Penulis mengucapkan terima kasih kepada:

1. Candra Dewi, S.Kom., MSc. selaku Dosen Pembimbing Skripsi I.
2. Drs. Muh. Arif Rahman, M.Kom. selaku Dosen Pembimbing Skripsi II.
3. Drs. Marji, MT. selaku Ketua Program Studi Teknik Informatika Program Teknologi Informasi & Ilmu Komputer Universitas Brawijaya.
4. Nurul Hidayat, SPd. MSc. selaku Dosen Penasehat Akademik.
5. Ir. Sutrisno, MT, selaku Ketua Program Teknologi Informasi & Ilmu Komputer Universitas Brawijaya.
6. Segenap Bapak dan Ibu dosen yang telah mendidik dan mengajarkan ilmunya kepada Penulis selama menempuh pendidikan di Program Teknologi Informasi & Ilmu Komputer Universitas Brawijaya.
7. Segenap staf dan karyawan di Program Teknik Informatika Program Teknologi Informasi & Ilmu Komputer Universitas Brawijaya yang telah banyak membantu Penulis dalam pelaksanaan penyusunan skripsi ini.
8. Kedua orang tua Penulis, Hari Tjahjono dan Dahlina Siregar yang tidak pernah henti memberikan doa, cinta serta dukungan kepada Penulis.
9. Kedua adik Penulis, Lukhy Fitri Apriliana dan M. Reza Rivandana yang selalu memberi dukungan dan doa kepada Penulis.
10. Ardhy Wisdarianto, S.Kom, yang telah memberi dukungan baik dalam bentuk material maupun non material demi terselesaikannya skripsi ini.

11. Sahabat-sahabat tercinta Hendra Prayuda, S.Kom, Alfian Ardhi, S.Kom, Ardhiyan Syahrullah, S.Kom yang senantiasa memberikan semangat dan doa demi terselesaikannya skripsi ini.
12. Teman-teman Ilmu Komputer angkatan 2008 yang telah banyak memberikan bantuan dan pengalaman selama menjadi mahasiswa di Universitas Brawijaya.
13. Teman-teman seperjuangan skripsi Nisa, Resthy, Jihan, Uly, Cici, Mila, Fahrur, David, Nina, Ikhwan, Mas Sandro, Mas Kadek, Mbak Tebe dan temen-temen lainnya yang senantiasa saling mendoakan dan mendukung demi terselesaikannya skripsi ini.
14. Teman-teman KKN Tulungrejo 2012 yang telah banyak memberikan semangat dan doa.
15. Semua pihak yang telah membantu terselesaikannya skripsi ini yang tidak dapat kami sebutkan satu per satu.

Penulis menyadari bahwa skripsi ini masih banyak kekurangan dan jauh dari sempurna, karena keterbatasan materi dan pengetahuan yang dimiliki penulis. Maka, saran dan kritik yang membangun dari semua pihak sangat diharapkan demi penyempurnaan selanjutnya. Semoga skripsi ini dapat bermanfaat dan berguna bagi semua pihak, baik penulis maupun pembaca, dan semoga Allah SWT meridhoi dan dicatat sebagai ibadah. Amin.

Malang, Januari 2013

Penulis

ABSTRAK

Maslikha Puspasari. 2013. Prediksi Tingkat Risiko Penyakit Jantung Koroner (PJK) Menggunakan Metode Fuzzy K-Nearest Neighbor (FK-NN).

Dosen Pembimbing : Candra Dewi, S.Kom., MSc dan Drs. Muh.Arif Rahman, M.Kom

Penyakit Jantung Koroner (PJK) merupakan salah satu penyakit yang menjadi salah satu penyebab kematian tertinggi di berbagai negara, tidak luput negara Indonesia juga termasuk di dalamnya. Dengan meningkatnya angka kematian di berbagai negara ini, maka alat diagnosis dan terapi terus dikembangkan, guna menanggulangi dan mengurangi meningkatnya angka kematian yang disebabkan oleh PJK. Salah satu metode penanggulangan yang dapat digunakan adalah teknik data mining. Pada penelitian ini digunakan metode gabungan Data Mining dengan Logika Fuzzy, yaitu Fuzzy K-Nearest Neighbor (FK-NN). Dimana metode Fuzzy K-Nearest Neighbor dapat melakukan prediksi Penyakit Jantung Koroner sesuai dengan faktor-faktor yang ada. Dimana pada penelitian ini faktor PJK yang digunakan yaitu umur, HDL, trigliserida, LDL, kolesterol dan sistolik. FK-NN melakukan prediksi dengan mencari nilai tetangga terdekat kemudian menggunakan basis nilai keanggotaan data uji dari setiap kelas dengan mengambil nilai keanggotaan terbesar sebagai hasil akhir prediksi. Pengujian metode ini digunakan pada data latih PJK dengan jumlah berbeda. Dimana pada data latih dengan jumlah 40, 50, 60 dan 70 didapat akurasi tertinggi sebesar 66,67% pada data latih 70. Semakin banyak data latih yang digunakan, maka kemungkinan semakin banyaknya jarak *record* yang mendekati kelas data prediksi.

Kata Kunci: Penyakit Jantung Koroner (PJK), Data Mining, Logika Fuzzy, Fuzzy K-Nearest Neighbor (FK-NN)

ABSTRACT

Maslikha Puspasari. 2013. Prediction Coronary Heart Disease (CHD) with method of Fuzzy K-Nearest Neighbor (FK-NN).

Advisor: Candra Dewi, S.Kom., MSc and Drs. Muh.Arif Rahman, M.Kom.

Coronary Heart Disease (CHD) is one of the diseases cause of death in many countries, included in Indonesia. With the increased mortality in these countries, the diagnostic tools and therapies continue to be developed, in order to prevent and reduce the increasing number of deaths caused by CHD. One method of control that can be used is data mining techniques. This research used a combined method of Data Mining with Fuzzy Logic, namely Fuzzy K-Nearest Neighbor (FK-NN). Where the method of Fuzzy K-Nearest Neighbor able to predict coronary heart disease in accordance with the existing factors. Where to study CHD factors used were age, HDL, triglycerides, LDL cholesterol and systolic. FK-NN to predict the value of the nearest neighbor search then use the base membership value of each class of test data by taking the largest membership value as the final prediction. The test method is used to train the data with a number of different CHD. Where the training data by the number 40, 50, 60 and 70 obtained the highest accuracy of 66.67% on the data train 70. The more data train is used, then the possibility of record with a distance approaching the data class prediction increase.

Kata Kunci: Coronary Heart Disease (CHD), Data Mining, Fuzzy Logic, Fuzzy K-Nearest Neighbor (FK-NN)

DAFTAR ISI

LEMBAR PERSETUJUAN	ii
LEMBAR PENGESAHAN	iii
PERNYATAAN ORISINALITAS SKRIPSI.....	iv
KATA PENGANTAR.....	v
ABSTRAK	vii
ABSTRACT	viii
DAFTAR ISI.....	ix
DAFTAR TABEL	xii
DAFTAR GAMBAR.....	xiv
DAFTAR SOURCECODE	xvi
DAFTAR LAMPIRAN	xvii
BAB I PENDAHULUAN.....	1
1.1 LATAR BELAKANG.....	1
1.2 PERUMUSAN MASALAH	3
1.3 TUJUAN PENELITIAN	3
1.4 BATASAN PERMASALAHAN.....	4
1.5 MANFAAT PENELITIAN	4
1.6 SISTEMATIKA PENULISAN	4
BAB II TINJAUAN PUSTAKA.....	6
2.1. PENYAKIT JANTUNG KORONER (PJK).....	6
2.1.1. Pengertian Penyakit Jantung Koroner (PJK)	6
2.1.2. Faktor Risiko Penyakit Jantung Koroner (PJK)	6
2.1.3. Kelas Risiko Penyakit Jantung Koroner (PJK).....	10
2.2. LOGIKA <i>FUZZY</i>	10
2.2.1. Pengertian Logika Fuzzy	10
2.2.2. Crisp Set.....	11
2.2.3. Fuzzy Set.....	11
2.2.4. Fungsi Keanggotaan	12
2.3. DATA MINING.....	15

2.3.1.	Pengertian Data Mining	15
2.3.2.	Teknik Data Mining	16
2.3.3.	Klasifikasi	17
2.3.4.	K-Nearest Neighbor (KNN).....	18
2.3.4.1.	Proses K-Nearest Neighbor (KNN).....	19
2.3.4.2.	Fungsi Kombinasi.....	21
2.3.4.3.	Algoritma K-Nearest Neighbor	21
2.4.	<i>FUZZY K-NEAREST NEIGHBOR CLASSIFIER</i>	22
2.4.1.	Pengertian <i>Fuzzy K-Nearest Neighbor</i> (FK-NN)	22
2.4.2.	Algoritma <i>Fuzzy K-Nearest Neighbor</i> (FK-NN)	23
2.5.	AKURASI HASIL PENGUJIAN	25
BAB III METODOLOGI DAN PERANCANGAN		26
3.1.	STUDI LITERATUR	27
3.2.	PENGUMPULAN DATA	27
3.3.	ANALISA DAN PERANCANGAN SISTEM	27
3.3.1.	Deskripsi Sistem	27
3.3.2.	Perancangan Sistem	30
3.3.2.1.	Perancangan Sistem Prediksi	30
3.3.2.2.	Normalisasi nilai setiap atribut.....	31
3.3.2.2.1.	Hitung Max.....	31
3.3.2.2.2.	Hitung Min	31
3.3.2.3.	Klasifikasi K-Nearest Neighbor (K-NN)	34
3.3.2.3.1.	Euclidean Distance	34
3.3.2.3.2.	Weight	34
3.3.2.3.3.	Weight sejumlah K.....	38
3.3.2.4.	Transformasi ke data fuzzy	38
3.4.	PERHITUNGAN MANUAL	43
3.5.	PERANCANGAN ANTARMUKA	52
3.6.	PERANCANGAN UJI COBA	54
2.6.1.	Uji Pengaruh Nilai k dan Data Latih Terhadap Tingkat Akurasi	55
3.6.3.	Uji Pengaruh Kelas mayoritas dan kelas minoritas	57
BAB IV IMPLEMENTASI DAN PEMBAHASAN		59

5.1.	LINGKUNGAN IMPLEMENTASI	59
5.1.1.	Lingkungan implementasi perangkat keras	59
5.1.2.	Lingkungan implementasi perangkat lunak	59
5.2.	IMPLEMENTASI PROGRAM.....	60
5.2.1.	Struktur Data.....	60
5.2.2.	Implementasi Pelatihan.....	64
5.2.2.1.	Implementasi Baca File Data Latih dan data Uji.	64
5.2.2.2.	Implementasi Normalisasi	65
5.2.2.3.	Implementasi K-Nearest Neighbor.....	69
5.2.2.3.1.	Implementasi Euclidean Distance	70
5.2.2.3.2.	Implementasi <i>Weight</i>	71
5.2.2.3.3.	Implementasi <i>Weight</i> sejumlah k.....	72
5.2.3.	Implementasi Transformasi ke data <i>fuzzy</i>	72
5.3.	IMPLEMENTASI ANTARMUKA.....	78
5.3.1.	Form Prediksi FK-NN	78
5.3.2.	Antarmuka Pengujian FK-NN	81
5.4.	SISTEMATIKA PENGUJIAN	87
5.4.1.	Sistematika Pengujian Pengaruh Nilai k dan Data Latih.....	87
5.4.2.	Sistematika Pengujian Pengaruh Kelas Mayoritas dan Minoritas ...	87
5.5.	IMPLEMENTASI UJI COBA	88
5.5.1.	Implementasi Pengujian Pengaruh Nilai k dan Pengaruh Jumlah Data Latih	88
5.5.2.	Implementasi Pengujian Pengaruh Kelas Mayoritas dan Minoritas	93
5.6.	ANALISA HASIL	96
5.6.1.	Analisa Hasil Pengaruh Nilai k dan Pengaruh Jumlah Data Latih ..	96
5.6.2.	Analisa Hasil Pengaruh Kelas Mayoritas dan Minoritas	101
BAB V KESIMPULAN DAN SARAN		105
5.1.	KESIMPULAN	105
5.2.	SARAN	106
DAFTAR PUSTAKA.....		107
LAMPIRAN.....		111

DAFTAR TABEL

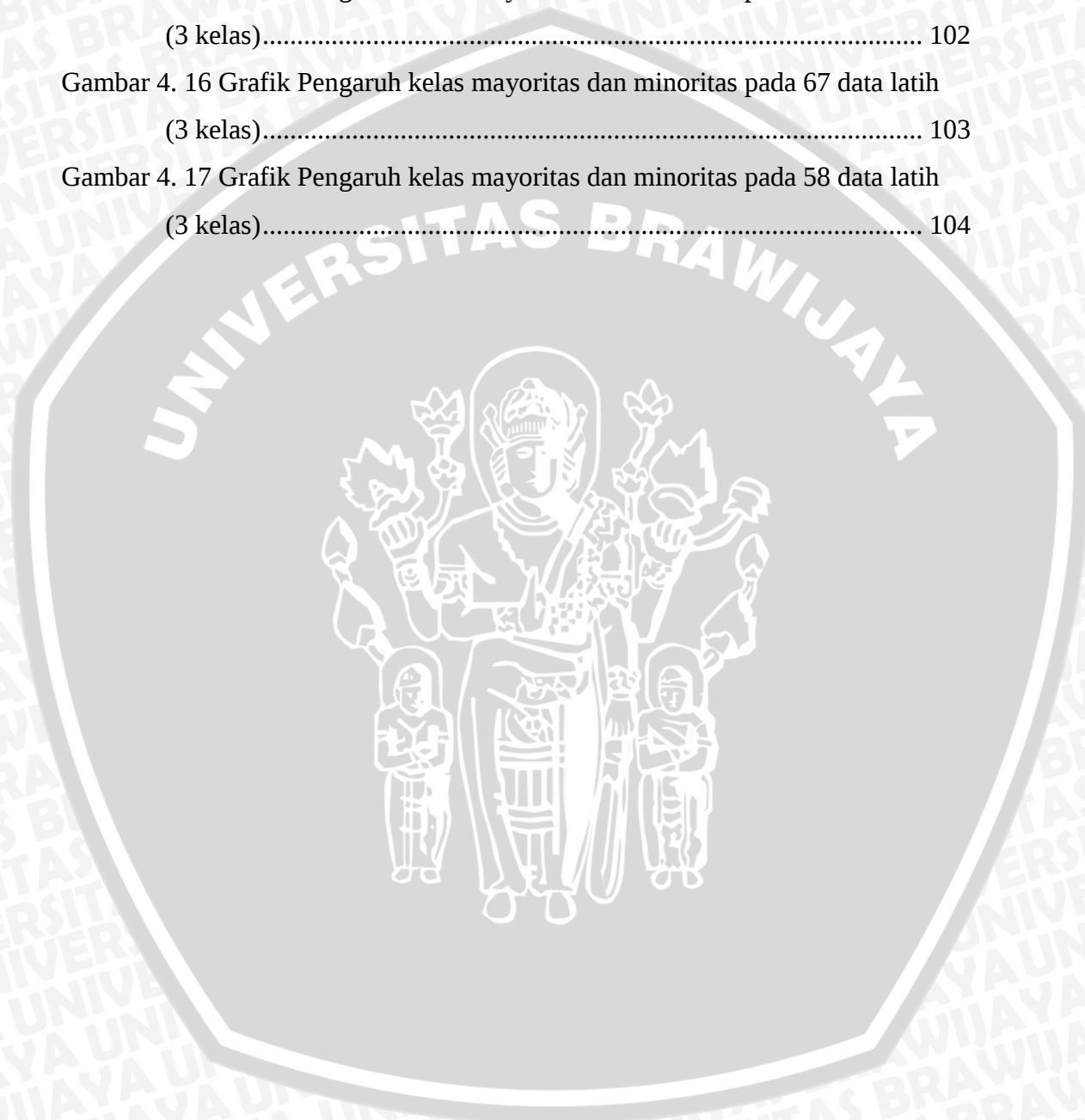
Tabel 2. 1 Klasifikasi Kadar LDL	8
Tabel 2. 2 Klasifikasi Kadar Kolesterol	8
Tabel 2. 3 Klasifikasi Kadar HDL	9
Tabel 2. 4 Klasifikasi Kadar Trigliserida	9
Tabel 2. 5 Klasifikasi Tekanan darah sistolik	10
Tabel 2. 6 Rentang nilai kelas risiko PJK	10
Tabel 3. 1 Komposisi Kategori Data Uji (30 data)	29
Tabel 3. 2 Komposisi Kategori Data Latih (40 data)	29
Tabel 3. 3 Komposisi Kategori Data Latih (50 data)	29
Tabel 3. 4 Komposisi Kategori Data Latih (60 data)	29
Tabel 3. 5 Komposisi Kategori Data Latih (70 data)	29
Tabel 3. 6 Data input pelatihan sistem	43
Tabel 3. 7 Data input uji sistem	44
Tabel 3. 8 Data yang akan dilakukan klasifikasi	44
Tabel 3. 9 Range Data Training	45
Tabel 3. 10 Data Pelatihan yang telah di normalisasi	45
Tabel 3. 11 Bobot setiap atribut data latih	46
Tabel 3. 12 Data Uji yang telah di normalisasi	46
Tabel 3. 13 Nilai jarak terdekat data uji dengan data pelatihan	46
Tabel 3. 14 Nilai weighted	47
Tabel 3. 15 Data setelah sorting dari yang terbesar	48
Tabel 3. 16 Seleksi 4 data dengan nilai terbesar	48
Tabel 3. 17 Target dalam linguistic membershipnya	49
Tabel 3. 18 Nilai Fuzzy untuk setiap target	49
Tabel 3. 19 Nilai <i>membership</i> akhir	50
Tabel 3. 20 Seleksi 1 data dengan nilai terbesar	50
Tabel 3. 21 Target dalam linguistic membership	51
Tabel 3. 22 Nilai Fuzzy untuk setiap target	51
Tabel 3. 23 Nilai <i>membership</i> akhir	52

Tabel 3. 24 Tabel Uji Pengaruh Nilai k dan Data Latih Terhadap Akurasi Sistem	55
Tabel 3. 25 Tabel Uji Pengaruh Jumlah Data Latih Terhadap Akurasi	56
Tabel 3. 26 Tabel Uji Pengaruh Kelas Mayoritas dan Kelas Minoritas.....	57
Tabel 4. 1 Pengujian Pengaruh nilai k pada 40 data latih	88
Tabel 4. 2 Pengujian Pengaruh nilai k pada 50 data latih	89
Tabel 4. 3 Pengujian Pengaruh nilai k pada 60 data latih	90
Tabel 4. 4 Pengujian Pengaruh nilai k pada 70 data latih	91
Tabel 4. 5 Pengujian Pengaruh nilai k dan jumlah data latih.....	92
Tabel 4. 6 Pengujian Pengaruh Kelas Mayoritas dan Minoritas pada 70 Data Latih dengan 5 Kelas.	93
Tabel 4. 7 Pengujian Pengaruh Kelas Mayoritas dan Minoritas pada 47 Data Latih dengan 3 Kelas.	94
Tabel 4. 8 Pengujian Pengaruh Kelas Mayoritas dan Minoritas pada 67 Data Latih dengan 3 Kelas	95
Tabel 4. 9 Pengujian Pengaruh Kelas Mayoritas dan Minoritas pada 58 Data Latih dengan 3 Kelas	95

DAFTAR GAMBAR

Gambar 2. 1 Kurva Bentuk Trapesium-Segitiga.....	13
Gambar 2. 2 Fungsi Keanggotaan Tingkat Risiko	15
Gambar 3. 1 Diagram Alir Penelitian	26
Gambar 3. 2 Alur Proses Pengujian Sistem	30
Gambar 3. 3 Alur Proses Normalisasi setiap atribut.....	32
Gambar 3. 4 Alur Proses Hitung Max.....	33
Gambar 3. 5 Alur Proses Hitung Min	33
Gambar 3. 6 Alur Proses K-Nearest Neighbor	35
Gambar 3. 7 Alur Proses Euclidean Distance	36
Gambar 3. 8 Alur Proses Weight	37
Gambar 3. 9 Alur Proses Weight sejumlah k.....	38
Gambar 3. 10 Alur Proses Transformasi Ke Data Fuzzy.....	39
Gambar 3. 11 Alur Proses Transformasi Ke Data Fuzzy.....	40
Gambar 3. 12 Alur Proses Transformasi Ke Data Fuzzy.....	41
Gambar 3. 13 Alur Proses Transformasi Ke Data Fuzzy.....	42
Gambar 3. 14 Antarmuka sistem prediksi.....	52
Gambar 3. 15 Antarmuka sistem pengujian.....	53
Gambar 4. 1 Antarmuka Prediksi FK-NN.....	79
Gambar 4. 2 Antarmuka Proses prediksi dengan input data tunggal	80
Gambar 4. 3 Antarmuka Pengujian FK-NN.....	81
Gambar 4. 4 Antarmuka Pengujian pada tab data uji	82
Gambar 4. 5 Antarmuka Pengujian pada tab data latih.....	83
Gambar 4. 6 Antarmuka Pengujian pada tab weight k.....	84
Gambar 4. 7 Antarmuka Pengujian pada tab Membership	85
Gambar 4. 8 Antarmuka Pengujian pada tab Akurasi.....	86
Gambar 4. 9 Grafik Pengaruh nilai k terhadap 40 data latih.....	96
Gambar 4. 10 Grafik Pengaruh 50 Data Latih terhadap nilai k	97
Gambar 4. 11 Grafik Pengaruh 60 Data Latih terhadap nilai k	98
Gambar 4. 12 Grafik Pengaruh 70 Data Latih terhadap nilai k	99

Gambar 4. 13 Grafik Pengaruh Jumlah Data Latih.....	100
Gambar 4. 14 Grafik Pengaruh kelas mayoritas dan minoritas pada 70 data latih (5 kelas).....	101
Gambar 4. 15 Grafik Pengaruh kelas mayoritas dan minoritas pada 47 data latih (3 kelas).....	102
Gambar 4. 16 Grafik Pengaruh kelas mayoritas dan minoritas pada 67 data latih (3 kelas).....	103
Gambar 4. 17 Grafik Pengaruh kelas mayoritas dan minoritas pada 58 data latih (3 kelas).....	104

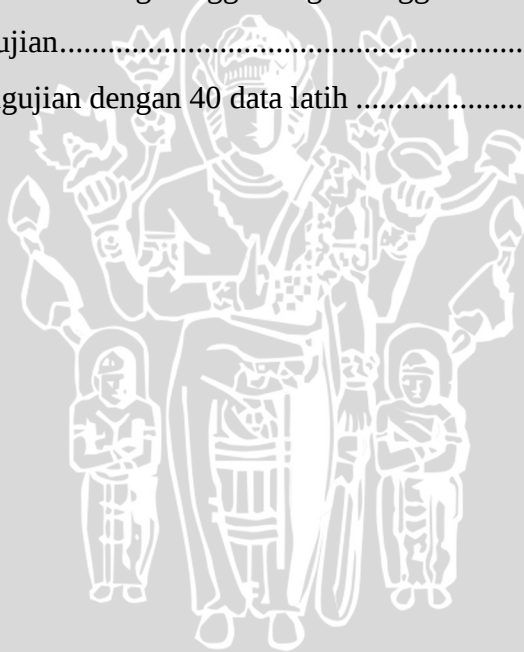


DAFTAR SOURCECODE

Sourcecode 4. 1 Kelas FormUtama.....	60
Sourcecode 4. 2 Kelas InputData.....	62
Sourcecode 4. 3 Kelas Euclidean.....	63
Sourcecode 4. 4 Kelas Weight.....	64
Sourcecode 4. 5 Kelas Membership.....	64
Sourcecode 4. 6 Baca File Data Latih.....	65
Sourcecode 4. 7 Perhitungan Max.....	66
Sourcecode 4. 8 Perhitungan Min.....	66
Sourcecode 4. 9 Perhitungan Normalisasi.....	66
Sourcecode 4. 10 Normalisasi setiap atribut.....	69
Sourcecode 4. 11 bobot setiap atribut (nilai tengah atribut kelas).....	69
Sourcecode 4. 12 Implementasi HitungEuclidean().....	70
Sourcecode 4. 13 Pemberian nilai Euclidean pada setiap data uji.....	71
Sourcecode 4. 14 Implementasi HitungWeight().....	71
Sourcecode 4. 15 Pemberian nilai Weight pada setiap data.....	72
Sourcecode 4. 16 Implementasi weight k.....	72
Sourcecode 4. 17 Impelementasi HitungNilaiMembership().....	73
Sourcecode 4. 18 Pemberian nilai Membership pada setiap data.....	75
Sourcecode 4. 19 Fungsi membership SangatRendah().....	76
Sourcecode 4. 20 Fungsi membership Rendah().....	76
Sourcecode 4. 21 Fungsi membership Sedang().....	77
Sourcecode 4. 22 Fungsi membership Tinggi().....	77
Sourcecode 4. 23 Fungsi membership SangatTinggi().....	78

DAFTAR LAMPIRAN

Lampiran 1 Data Pelatihan 70.....	111
Lampiran 2 Data Pelatihan 60.....	112
Lampiran 3 Data Pelatihan 50.....	114
Lampiran 4 Data Pelatihan 40.....	115
Lampiran 5 Data Pelatihan Sangat Rendah -Rendah-Sedang-Tinggi-Sangat Tinggi	116
Lampiran 6 Data Pelatihan Sangat Rendah -Rendah-Sedang.....	118
Lampiran 7 Data Pelatihan Rendah-Sedang-Tinggi	119
Lampiran 8 Data Pelatihan Sedang-Tinggi-Sangat Tinggi	121
Lampiran 9 Data Pengujian.....	122
Lampiran 10 Hasil Pengujian dengan 40 data latih	123



BAB I PENDAHULUAN

1.1 Latar Belakang

Penyakit Jantung Koroner (PJK) atau penyakit *kardiovaskular* merupakan salah satu penyebab kematian tertinggi di berbagai negara, tidak luput juga negara Indonesia juga termasuk di dalamnya. Penyakit Jantung Koroner juga telah ditetapkan oleh World Health Organization (WHO), termasuk dalam 10 macam penyakit mematikan yang dapat menyebabkan manusia kehilangan nyawa [DIR-06]. Dengan meningkatnya angka kematian diberbagai negara ini, maka alat diagnosis dan terapi terus dikembangkan, guna menanggulangi dan mengurangi meningkatnya angka kematian. Bahkan, sekarang di tahun 2000-an ini dapat dipastikan, kecenderungan penyebab kematian di Indonesia bergeser dari penyakit infeksi ke penyakit kardiovaskular biasa juga disebut sistem peredaran darah (antara lain Penyakit Jantung Koroner) dan degeneratif (Majid, 2007). Dimana 60% dari seluruh penyebab kematian yang berhubungan dengan penyakit jantung adalah penyakit jantung koroner (PJK) [SUP-08]. Dengan kata lain, lebih kurang satu diantara empat orang yang meninggal di Indonesia adalah akibat PJK.

Penyebab PJK secara pasti belum diketahui, meskipun demikian secara umum dikenal berbagai faktor yang berperan penting terhadap timbulnya PJK yang disebut sebagai faktor risiko PJK. Dikarenakan banyak faktor yang mempengaruhi terjadinya Penyakit Jantung Koroner sehingga usaha pencegahan harus dilakukan sedini mungkin. 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. Faktor resiko PJK yang utama adalah: *Hipertensi*, *Hiperkolesterolemia* dan merokok. Ketiga faktor ini saling mempengaruhi dan memperkuat resiko PJK akan tetapi dapat diperbaiki bila upaya pencegahan betul-betul dilaksanakan [ANW-04].

Pada umumnya untuk mendiagnosa PJK, seorang ahli khususnya para dokter akan melakukan beberapa proses yang akan diajukan kepada penderita

ataupun khalayak umum mencakup beberapa serangkaian pertanyaan maupun wawancara. Hal ini dilakukan guna mengetahui efek PJK dalam kegiatan sehari-hari. Setelah sesi ini selesai, para ahli akan melakukan pemeriksaan fisik secara lengkap kepada seseorang [PRI-94].

Data dari faktor risiko Penyakit Jantung Koroner (PJK) merupakan data yang masih bersifat samar dan terdapat unsur yang masih belum pasti didalamnya [WAH-11]. Untuk mengatasi hal ini, digunakannya metode *fuzzy* untuk mendata risiko penyakit jantung koroner ini. 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 [PRI-05].

Pada penelitian ini digunakan metode *fuzzy* dengan gabungan metode pada data mining untuk memprediksikan data. Pada data mining, terdapat beberapa metode untuk klasifikasi / prediksi diantaranya yaitu metode *K-Nearest Neighbor*, ID3, C45, Bayesian dan beberapa metode lainnya. Dimana dalam klasifikasi, terdapat target variabel kategori [KUS-09].

Metode *K-Nearest Neighbor* (K-NN) merupakan metode klasifikasi data yang cara kerjanya bersifat relative dengan cara yang lebih sederhana bila dibandingkan dengan metode klasifikasi pada data mining lainnya. Dimana metode ini melakukan klasifikasi kepada data baru yang masih belum diketahui masuk kedalam kelas mana, dengan menggunakan beberapa data dengan sejumlah k yang letaknya terdekat dengan data baru tersebut [JNI-96].

Metode K-NN dapat di gabungkan dengan beberapa metode lain, salah satu variasi dari penggunaan metode K-NN ini adalah *Fuzzy K-NN* dimana metode ini menggabungkan teknik *fuzzy* dengan teknik data mining [ZHA-09]. *Fuzzy K-NN* telah sebelumnya telah digunakan untuk memprediksi IRIS, Vertebratal Column, Liver, diabetes oleh [KEL-97] dan Web Clasificaion Document oleh [ZHA-09]. Dimana, dalam implementasinya algoritma *fuzzy K-NN* memberikan nilai keanggotaan kelas pada *vektor* sampel dan bukan menempatkan *vektor* pada kelas tertentu. Keuntungannya adalah nilai-nilai keanggotaan *vektor* seharusnya memberikan tingkat jaminan pada hasil klasifikasi. Dasar dari algoritma *Fuzzy K-*

NN adalah untuk menetapkan nilai keanggotaan sebagai fungsi jarak *vektor* dari K-NN dan keanggotaan tetangga mereka di kelas-kelas yang memungkinkan.

Berdasarkan uraian di atas, maka penelitian ini diberi judul “**Prediksi Tingkat Risiko Penyakit Jantung Koroner (PJK) Menggunakan Metode Fuzzy K-Nearest Neighbor (FK-NN)**”. Data yang digunakan untuk prediksi tingkat risiko dari dataset Penyakit Jantung Koroner (PJK) yaitu umur, kadar LDL, kadar HDL, kolesterol total, trigliserida dan tekanan darah sistolik. Dimana sebelumnya data ini telah digunakan sebelumnya oleh Wahyuni (2011), Ludviani (2011), Yeni (2011) dan Seviana (2012).

1.2 Perumusan Masalah

Rumusan masalah dalam skripsi ini adalah:

1. Bagaimana implementasi metode *Fuzzy K-Nearest Neighbor* (FK-NN) untuk klasifikasi pada dataset Penyakit Jantung Koroner (PJK) yang berupa data umur, kadar LDL, kadar HDL, kolesterol total, trigliserida dan tekanan darah sistolik?
2. Bagaimana pengaruh sejumlah nilai k (tetangga) serta jumlah data uji terhadap tingkat akurasi hasil klasifikasi dataset Penyakit Jantung Koroner (PJK) menggunakan metode *Fuzzy K-Nearest Neighbor* (FK-NN)?

1.3 Tujuan Penelitian

Tujuan dari skripsi ini adalah sebagai berikut:

1. Mengimplementasikan metode gabungan *Fuzzy K-Nearest Neighbor* (FK-NN) untuk mengetahui tingkat risiko pada dataset Penyakit Jantung Koroner (PJK).
2. Mengetahui pengaruh sejumlah nilai k (tetangga) serta jumlah data latih terhadap tingkat risiko hasil klasifikasi pada dataset Penyakit Jantung Koroner (PJK) menggunakan metode *Fuzzy K-Nearest Neighbor* (FK-NN).

1.4 Batasan Permasalahan

Pada skripsi ini, permasalahan dibatasi sebagai berikut:

1. Data yang digunakan untuk pengujian pada skripsi ini, didapatkan dari data hasil laboratorium pasien RSUD Saiful Anwar, RSI Unisma Malang, dan laboratorium cek fisik kesehatan.
2. Parameter Penyakit Jantung Koroner (PJK) yang digunakan berupa data umur, kadar LDL, kadar HDL, kolesterol total, trigliserida dan tekanan darah sistolik serta satu output hasil inferensi pada penelitian Wahyuni (2011).
3. Diagnosa risiko PJK pada sistem ditentukan berdasarkan diagnosa kelas risiko PJK pada penelitian yang dilakukan Wahyuni (2011), yaitu: sangat rendah, rendah, sedang, tinggi, dan sangat tinggi.
4. Nilai akurasi sistem dibandingkan dengan hasil inferensi pada penelitian Wahyuni (2011).
5. Diagnosa risiko PJK tidak bergantung pada Jenis Kelamin (laki-laki atau perempuan), tidak bergantung pada faktor-faktor seperti Diabetes Melitus, riwayat hidup, dan faktor-faktor lainnya. Diagnosa risiko PJK hanya berdasarkan faktor-faktor umur, kadar LDL, kadar HDL, kolesterol total, trigliserida dan tekanan darah sistolik.

1.5 Manfaat Penelitian

Manfaat yang dapat diambil dari skripsi ini yaitu dapat diklasifikasikannya Penyakit Jantung Koroner (PJK) berdasarkan data faktor risiko, dengan menggunakan *Fuzzy K-Nearest Neighbor* (FK-NN). Sehingga dapat diketahui termasuk dalam tingkatan apa Penyakit Jantung Koroner (PJK) yang diderita seseorang.

1.6 Sistematika Penulisan

1. BAB I : PENDAHULUAN

Bab ini berisi latar belakang penulisan, perumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, serta sistematika penulisan skripsi.

2. BAB II : TINJAUAN PUSTAKA

Bab ini berisi tentang prediksi tingkat risiko menggunakan klasifikasi *Fuzzy K-Nearest Neighbor* (FK-NN), serta diagnosis Penyakit Jantung Koroner secara umum.

3. BAB III : METODOLOGI DAN PERANCANGAN

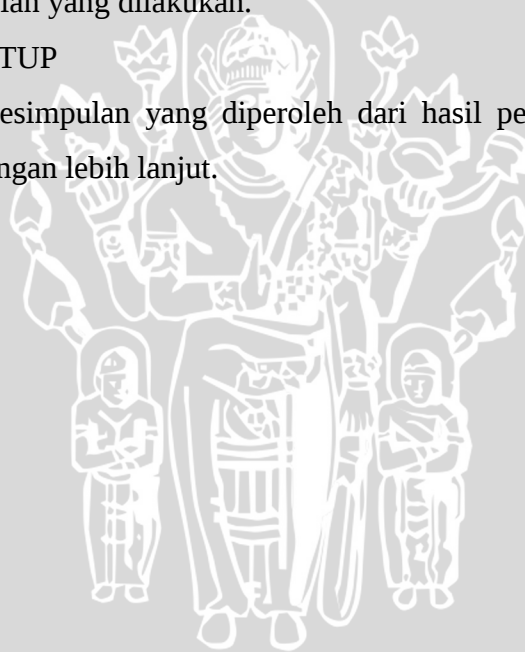
Bab ini berisi tentang metode-metode yang digunakan untuk prediksi tingkat risiko menggunakan klasifikasi *Fuzzy K-Nearest Neighbor* (FK-NN), pada studi kasus diagnosis Penyakit Jantung Koroner.

4. BAB IV : HASIL DAN PEMBAHASAN

Bab ini berisi pembahasan dari implementasi prediksi tingkat risiko menggunakan klasifikasi *Fuzzy K-Nearest Neighbor* (FK-NN) pada sistem dan hasil pengujian yang dilakukan.

5. BAB V : PENUTUP

Bab ini berisi kesimpulan yang diperoleh dari hasil pengujian dan saran untuk pengembangan lebih lanjut.



BAB II

TINJAUAN PUSTAKA

2.1. Penyakit Jantung Koroner (PJK)

2.1.1. Pengertian Penyakit Jantung Koroner (PJK)

Penyakit Jantung Koroner (PJK) merupakan penyakit jantung yang disebabkan karena adanya penyempitan arteri koronia akibat proses atau spasme atau kombinasi keduanya [ABD-07]. Bilamana penyempitan ini menjadi parah maka dapat terjadi serangan jantung. Adapun penyempitan pembuluh arteri ke otak dapat menimbulkan stroke. Otot jantung diberi oksigen dan nutrisi yang diangkut oleh darah melalui arteri-arteri koroner utama yang bercabang menjadi sebuah jaringan pembuluh lebih kecil yang efisien. Sedangkan arteri ke otak yang mengangkut substansi yang sama [SOE-10].

Penyakit jantung koroner (PJK) adalah penyakit jantung dan pembuluh darah yang disebabkan karena penyempitan arteri koroner. Penyempitan pembuluh darah terjadi karena proses aterosklerosis atau spasme atau kombinasi keduanya. Aterosklerosis yang terjadi karena timbunan kolesterol dan jaringan ikat pada dinding pembuluh darah secara perlahan-lahan, hal ini sering ditandai dengan keluhan nyeri pada dada [HAY-99].

2.1.2. Faktor Risiko Penyakit Jantung Koroner (PJK)

Pola timbulnya penyakit PJK menarik para ahli peneliti medis. Diantaranya dari Framingham Heart Study, USA, suatu institusi yang terkenal dalam penyakit kardiovaskuler, berpendapat bahwa PJK bukanlah penyakit manusia lanjut usia (manula) atau nasib buruk yang tidak dapat dihindari tetapi terdapat beberapa faktor yang mempengaruhi seseorang terkena PJK. Dalam hubungan ini dikenal adanya “Faktor Risiko PJK” yaitu kondisi yang berkaitan dengan meningkatnya risiko timbulnya PJK. Faktor risiko tersebut diantaranya adalah tekanan darah, merokok, lipid, diabetes mellitus, obesitas dan riwayat keluarga dengan penyakit jantung [SUP-08].

Faktor risiko penyakit jantung koroner ada yang membaginya dalam faktor risiko primer (independen) dan sekunder [KRI-02], yaitu:

1. Faktor risiko primer: Faktor ini dapat menyebabkan gangguan arteri berupa aterosklerosis tanpa harus dibantu oleh faktor lain (independen), termasuk faktor risiko primer, yaitu hiperlidemi, merokok dan hipertensi.
2. Faktor risiko sekunder: Faktor ini baru dapat menimbulkan kelainan arteri bila ditemukan faktor lain secara bersamaan, termasuk faktor risiko sekunder, yaitu diabetes melitus (DM), obesitas, stres, kurang olah raga, alkohol, dan riwayat keluarga.

Tiga faktor resiko utama yang saling terkait sebagai penyebab PJK yaitu kebiasaan merokok, kurang aktivitas fisik, makan tidak seimbang, kegemukan, diet rendah serat atau kurang buah dan sayur dan tinggi kalori/lemak hewani dan lain-lain terus meningkat.

Pendeteksian PJK dapat menggunakan beberapa faktor risiko penyebab PJK, seperti umur, jenis kelamin, pekerjaan, kadar LDL, kadar kolesterol total, kadar HDL, kadar trigleserida, tekanan darah sistolik dan tekanan darah diastolik [EFF-08]. Namun, dalam bab ini akan dibahas mengenai faktor-faktor risiko PJK yang menjadi batasan permasalahan pada penelitian ini. Faktor-faktor risiko penyakit Jantung Koroner tersebut meliputi:

1. Usia

Umur termasuk faktor risiko PJK karena menyebabkan perubahan di dalam jantung dan pembuluh darah. Seiring dengan pertambahan umur, kadar kolesterol seseorang akan meningkat. Tekanan darah juga akan meningkat sesuai umur karena arteri secara perlahan kehilangan elastisitas. Secara umum, semakin tua umur, risiko PJK semakin meningkat pula [ANW-04].

2. *Low Density Lipoprotein (LDL)*

LDL (*Low Density Lippoprotein*) adalah jenis lipoprotein yang mengangkut kolesterol dan trigliserida dari hati ke jaringan perifer. Kadar LDL dinyatakan dengan satuan mg/dl. Peningkatan kadar LDL berbanding lurus dengan peningkatan risiko PJK.

Semakin tinggi kadar LDL, risiko PJK semakin meningkat pula

[ANW-04]. Klasifikasi kadar LDL ditunjukkan pada tabel 2.1 [TJO-01]:

Tabel 2. 1 Klasifikasi Kadar LDL

Klasifikasi	Kadar Kolestorel
Normal	< 100
Mendekati Normal	100 – 129
Borderline high	130 – 159
Tinggi	160 – 189
Sangat tinggi	≥ 190

3. 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 makanan yang masuk ke dalam tubuh masih sesuai dengan kebutuhan, maka tubuh akan tetap dalam keadaan sehat, tetapi jika lebih maka akan mengendap di dalam pembuluh darah yang menyebabkan terjadinya 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]. Klasifikasi kadar kolesterol ditunjukkan pada tabel 2.2 [TJO-01].

Tabel 2. 2 Klasifikasi Kadar Kolesterol

Klasifikasi	Kadar Kolestorel
Normal	< 200
Borderline (waspada)	200 – 239
Tinggi	≥ 240

4. *High Density Lipoprotein* (HDL)

HDL (*High Density Lipoprotein*) merupakan jenis kolesterol yang bersifat baik atau menguntungkan, karena mengangkut kolesterol dari pembuluh darah kembali ke hati untuk dibuang sehingga mencegah penebalan dinding pembuluh darah atau mencegah terjadinya proses

aterosklerosis. Peningkatan kadar HDL berbanding terbalik dengan peningkatan risiko PJK. Semakin tinggi kadar HDL, semakin rendah tingkat risiko PJK [ANW-04]. Klasifikasi kadar HDL ditunjukkan pada tabel 2.3 [TJO-01].

Tabel 2. 3 Klasifikasi Kadar HDL

Klasifikasi	Kadar Kolestrol
Rendah	< 40
Tinggi	≥ 60

5. Trigliserida

Trigliserida merupakan komponen yang normal dari darah, baik datang dari diet atau dihasilkan oleh tubuh. Sebagian besar lemak yang dimakan berbentuk *trigliserida*. Makan makanan yang mengandung lemak akan meningkatkan kadar kolesterol. Lemak berasal dari buah-buahan seperti kelapa, durian dan alpukat tidak mengandung kolesterol, tetapi kadar trigliseridanya tinggi [SOE-04]. Klasifikasi kadar trigliserida ditunjukkan pada tabel 2.4 [TJO-01]:

Tabel 2. 4 Klasifikasi Kadar Trigliserida

Klasifikasi	Kadar Kolestrol
Normal	< 150
Borderline high	150 – 199
Tinggi	200 – 499
Sangat tinggi	≥ 500

6. Tekanan darah sistolik

Tekanan darah sistolik adalah tekanan saat jantung berdenyut atau berdetak. Untuk diagnosa risiko PJK, tekanan darah sistolik menunjukkan risiko yang lebih tinggi terjangkit penyakit jantung koroner daripada tekanan darah diastolik. Semakin tinggi kolesterol total, risiko PJK semakin meningkat pula [ANW-04]. Klasifikasi tekanan darah sistolik dijelaskan pada tabel 2.5 [TJO-01]:

Tabel 2. 5 Klasifikasi Tekanan darah sistolik

Klasifikasi	Kadar Kolestorel
Normal	< 130
Normal tinggi	130 – 139
Hipertensi	
1. Ringan	140 – 159
2. Sedang	160 – 179
3. Berat	180 – 209
4. Sangat Berat	≥ 210

2.1.3. Kelas Risiko Penyakit Jantung Koroner (PJK)

Risiko penyakit jantung koroner (PJK) terbagi dalam 5 kelas yaitu tingkat risiko PJK sangat rendah, rendah, sedang, tinggi, dan sangat tinggi. Rentang nilai risiko PJK untuk setiap kelas diadaptasi dari penelitian Wahyuni (2011).

Tabel 2. 6 Rentang nilai kelas risiko PJK

No	Rentang nilai risiko (z)	Kelas risiko PJK
1	Kurang dari 10 ($z \leq 10$)	Sangat rendah
2	10 – 20 ($10 < z \leq 20$)	Rendah
3	20 – 30 ($20 < z \leq 30$)	Sedang
4	30 – 40 ($30 < z \leq 40$)	Tinggi
5	Lebih dari 40 ($z > 40$)	Sangat tinggi

2.2. Logika Fuzzy

2.2.1. Pengertian Logika Fuzzy

Logika fuzzy merupakan salah satu komponen pembentuk *soft computing*. Logika *fuzzy* pertama kali diperkenalkan oleh Prof. Lotfi A. Zadeh pada tahun 1965. Dasar logika *fuzzy* adalah teori himpunan *fuzzy*. Pada teori himpunan *fuzzy*, peranan derajat keanggotaan sebagai penentu keberadaan elemen dalam suatu himpunan sangatlah penting. Nilai atau derajat keanggotaan atau *membership function* sebagai ciri utama dari penalaran dengan logika *fuzzy* tersebut [SRI-10].

Dalam penyajiannya variabel-variabel yang akan digunakan harus cukup menggambarkan nilai *fuzzy*, tetapi di lain pihak persamaan-persamaan yang dihasilkan dari variabel-variabel itu haruslah cukup sederhana sehingga komputasinya menjadi cukup mudah. Karena itu Profesor Lotfi A. Zadeh kemudian memperoleh ide untuk menyajikannya dengan menentukan “derajat keanggotaan”

(*membership function*) dari masing-masing variabelnya [SUD-08].

Dalam konsepnya logika *fuzzy* meniru cara berpikir manusia dengan menggunakan konsep sifat kesamaan suatu nilai. Dengan menggunakan logika *fuzzy* nilai yang dihasilkan bukan hanya iya (1, *high*, *on*) atau tidak (0, *low*, *off*) tetapi seluruh kemungkinan diantara 0 dan 1. Teknologi Fuzzy ini sudah sangat berkembang pesat dan banyak digunakan dalam pemakaian pengaturan lalu lintas, sistem tranmisi otomatis, industri, dan lain-lain.

2.2.2. Crisp Set

Himpunan *Crisp* (*Crisp Set*) A didefinisikan oleh item-item yang ada pada himpunan itu. Jika $A \ni a$, maka nilai yang berhubungan dengan a adalah 1. Namun, jika $A \ni a$, maka nilai yang berhubungan dengan a adalah 0. $A = \{x|P(x)\}$ menunjukkan bahwa A berisi item x dengan $P(x)$ benar. Jika X_A merupakan fungsi karakteristik A dan properti P , dapat dikatakan bahwa $P(x)$ benar, jika dan hanya jika $X_A(x) = 1$ [SUD-08].

2.2.3. Fuzzy Set

Himpunan *fuzzy* (*fuzzy set*) didasarkan pada gagasan untuk memperluas jangkauan fungsi karakteristik sedemikian hingga fungsi tersebut akan mencakup bilangan *real* pada interval $[0,1]$. Nilai keanggotaannya menunjukkan bahwa suatu item tidak hanya bernilai benar atau salah. Nilai 0 menunjukkan salah, nilai 1 menunjukkan benar, dan masih ada nilai-nilai yang terletak antara benar dan salah [SUD-08].

Pada himpunan tegas (*crisp*), nilai keanggotaan suatu item x dalam suatu himpunan A , yang sering ditulis dengan $\mu_A(x)$, memiliki dua kemungkinan (Kusumadewi, 2010), yaitu:

- 1) Satu (1), yang berarti bahwa suatu item menjadi anggota dalam suatu himpunan, atau
- 2) Nol (0), yang berarti bahwa suatu item tidak menjadi anggota dalam suatu himpunan.

Jika pada himpunan *crisp*, nilai keanggotaan hanya ada 2 kemungkinan, yaitu 0 atau 1, pada himpunan *fuzzy* nilai keanggotaannya terletak pada rentang 0

sampai 1. Apabila x memiliki nilai keanggotaan fuzzy $\mu_A(x) = 1$ berarti x menjadi anggota penuh pada himpunan A . Himpunan fuzzy memiliki 2 atribut, yaitu:

- 1) *Linguistik*, yaitu penamaan suatu grup yang mewakili suatu keadaan atau kondisi tertentu dengan menggunakan bahasa alami, seperti: MUDA, PAROBAYA, TUA
- 2) *Numerik*, yaitu suatu nilai (angka) yang menunjukkan ukuran dari suatu variabel, seperti: 40, 25, 50, dsb.

Ada beberapa hal yang perlu diketahui dalam memahami sistem fuzzy yaitu:

- a) Variabel fuzzy merupakan variabel yang akan dibahas dalam sistem fuzzy misalnya umur, temperatur, permintaan, dsb.
- b) Himpunan fuzzy merupakan suatu group yang mewakili suatu kondisi tertentu dalam variabel fuzzy, misalnya variabel umur dibagi atas 3 himpunan fuzzy yaitu muda, parobaya dan tua.
- c) Semesta Pembicaraan adalah keseluruhan nilai yang diperbolehkan untuk dioperasikan dalam suatu variabel fuzzy misalnya semesta pembicaraan variabel umur adalah 0 sampai 100.
- d) *Domain* adalah keseluruhan nilai yang diijinkan dalam semesta pembicaraan dan boleh dioperasikan dalam himpunan fuzzy misalnya domain umur muda 20–45, domain parobaya 25–65 dan domain tua 45–70 [SRI-10].

2.2.4. Fungsi Keanggotaan

Fungsi keanggotaan (*membership function*) adalah suatu kurva yang menunjukkan pemetaan titik-titik input data ke dalam nilai keanggotaannya (sering disebut dengan derajat keanggotaan) yang memiliki interval antara 0 sampai 1. Salah satu cara yang dapat digunakan untuk mendapatkan nilai keanggotaan adalah dengan melakukan pendekatan fungsi [SRI-10].

Dalam teori himpunan fuzzy suatu objek dapat menjadi anggota dari banyak himpunan dengan derajat keanggotaan yang berbeda dalam masing-masing himpunan. Derajat keanggotaan menunjukkan nilai keanggotaan suatu objek pada suatu himpunan yang nilainya berkisar antara 0 sampai 1 [COX-05].

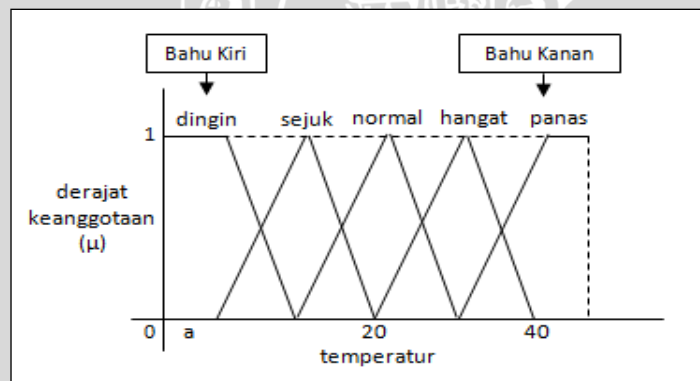
Asumsikan X adalah kumpulan elemen yang ditandai oleh x , maka

himpunan *fuzzy* (*fuzzy set*) A pada X didefinisikan sebagai berikut:

$$A = \{(x, \mu_A(x)) | x \in X\} \quad (2-1)$$

dimana $\mu_A(x)$ adalah nilai keanggotaan (*membership value*) dari x pada A yang memetakan X ke ruang keanggotaan [0,1]. Untuk membantu mencari nilai derajat keanggotaan suatu elemen dalam himpunan *fuzzy* digunakan fungsi keanggotaan (*fuzzy membership function*). Fungsi keanggotaan tersebut dapat memiliki beberapa bentuk yang berbeda asalkan memetakan ke [0,1] [JIN-04].

Beberapa tipe fungsi keanggotaan yang umum antara lain trapesium dan segitiga. Daerah yang terletak di tengah-tengah suatu variabel yang direpresentasikan dalam bentuk segitiga, pada sisi kanan dan kirinya akan naik dan turun. Tetapi terkadang salah satu sisi dari variabel tersebut tidak mengalami perubahan. Himpunan *fuzzy* 'bahu', bukan segitiga, digunakan untuk mengakhiri variabel suatu daerah *fuzzy*. Bahu kiri bergerak dari benar ke salah, demikian juga bahu kanan bergerak dari salah ke benar. Sebagai contoh, himpunan *fuzzy* pada variabel TEMPERATUR dengan daerah bahunya.



Gambar 2. 1 Kurva Bentuk Trapesium-Segitiga

Kurva trapezium-segitiga ini merupakan gabungan dari kurva trapezium dan kurva segitiga. Contohnya, fungsi keanggotaan segitiga ditentukan oleh tiga parameter (a, b, c) dan untuk nilai x yang diberikan dengan nilai a, b, dan c diketahui maka fungsi keanggotaan x dapat dihitung sebagai:

$$\mu(x) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ \frac{c-x}{c-b}, & b \leq x \leq c \\ 0, & x \geq c \end{cases} \quad (2-2)$$

untuk fungsi keanggotaan trapezium sendiri memiliki 4 parameter (a, b, c, d) dan diketahui fungsi keanggotaan x sebagai berikut:

$$\mu[x] = \begin{cases} 0; x \leq a \text{ atau } x \geq d \\ \frac{(x-a)}{(b-a)}; a \leq x \leq b \\ 1; & b \leq x \leq c \\ \frac{d-x}{d-c}; c \leq x \leq d \end{cases} \quad (2-3)$$

Peubah *linguistik* merupakan peubah verbal yang dapat digunakan untuk mewakili deskripsi pemikiran manusia berupa kumpulan kata (*linguistic term*) yang bersesuaian dengan derajat keanggotaan dalam suatu himpunan. Contoh deskripsi ini termasuk "tinggi", "rendah", "sangat rendah", "sangat tinggi", dan lain-lain. Sebuah variabel *linguistik* X memiliki seperangkat term L(X) yang elemen-elemennya adalah label dari nilai-nilai *linguistik* X. Sebagai contoh, variabel *linguistik* bernama suhu, L (suhu) = {sangat rendah, rendah, sedang, tinggi, sangat tinggi} [JIN-04]. Tingkat risiko penyakit jantung koroner diketahui fungsi keanggotaan x sebagai berikut [WAH-11]:

Tingkat Risiko Penyakit Jantung

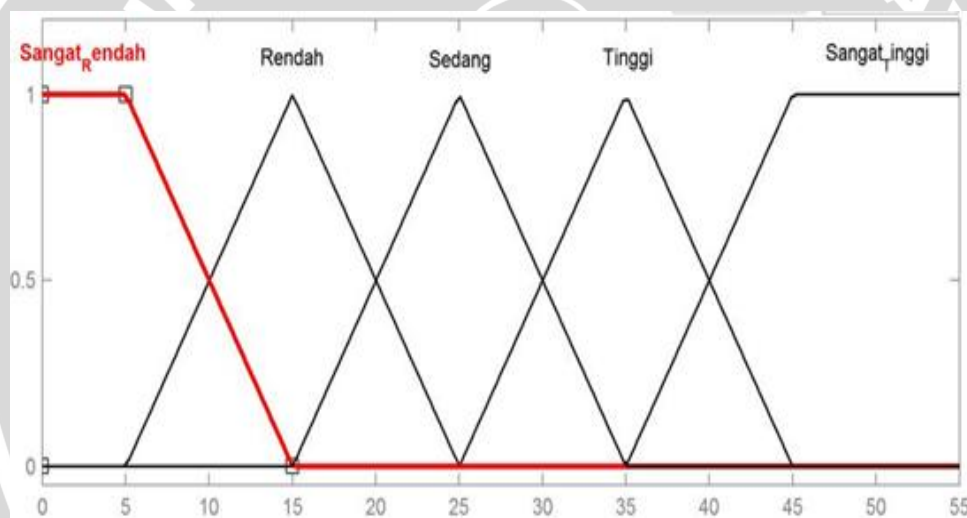
$$\mu_{\text{SANGAT RENDAH}}(x) = \begin{cases} 1, & 0 \leq x < 5 \\ \frac{(5-x)}{10}, & 5 \leq x < 15 \\ 0, & x \geq 15 \end{cases}$$

$$\mu_{\text{RENDAH}}(x) = \begin{cases} 0, & x \leq 5, x \geq 25 \\ \frac{x-5}{10}, & 5 \leq x < 15 \\ \frac{25-x}{10}, & 15 \leq x < 25 \end{cases}$$

$$\mu_{\text{SEDANG}}(x) = \begin{cases} 0, & x \leq 15, x \geq 35 \\ \frac{x-15}{10}, & 15 \leq x < 25 \\ \frac{35-x}{10}, & 25 \leq x < 35 \end{cases}$$

$$\mu_{\text{TINGGI}}(x) = \begin{cases} 0, & x \leq 25, x \geq 45 \\ \frac{x-25}{10}, & 25 \leq x < 35 \\ \frac{45-x}{10}, & 35 \leq x < 45 \end{cases}$$

$$\mu_{\text{SANGAT TINGGI}}(x) = \begin{cases} 0, & x \leq 35 \\ \frac{x-35}{10}, & 35 \leq x < 45 \\ 1, & x \geq 45 \end{cases}$$



Gambar 2. 2 Fungsi Keanggotaan Tingkat Risiko

2.3. Data Mining

2.3.1. Pengertian Data Mining

Data mining adalah suatu istilah yang digunakan untuk menguraikan penemuan pengetahuan di dalam database. Data Mining merupakan suatu proses yang menggunakan teknik statistika, matematika, kecerdasan buatan dan *machine learning* untuk mengidentifikasi informasi yang bermanfaat dan pengetahuan yang terakit dari berbagai database besar [TUR-05]. Data Mining merupakan teknologi yang sangat berguna untuk membantu perusahaan-perusahaan

menemukan informasi yang sangat penting dari gudang data mereka. Dengan data mining dapat meramalkan *trend* dan sifat-sifat perilaku bisnis yang sangat berguna untuk mendukung pengambilan keputusan penting. Analisis yang diotomatisasi yang dilakukan oleh data mining melebihi yang dilakukan oleh sistem pendukung keputusan tradisional yang sudah banyak digunakan. Data Mining mengeksplorasi basis data untuk menemukan pola-pola yang tersembunyi, mencari informasi untuk memprediksi yang mungkin saja terlupakan oleh para pelaku bisnis karena terletak di luar ekspektasi mereka [KUS-07].

Metode data mining ini dapat dibagi menjadi beberapa kelompok berdasarkan tugas yang dapat dilakukan, diantaranya yaitu metode deskripsi, estimasi, prediksi, klasifikasi, pengklusteran dan asosiasi [LAR-05].

Pada penelitian ini digunakan metode data mining dengan prediksi menggunakan metode gabungan klasifikasi *k-Nearest neighbor* dengan metode *fuzzy*.

Terdapat beberapa fungsionalitas *data mining*, sebagai berikut [HAN-00]:

1. Analisis asosiasi

Menemukan aturan (*rule*) asosiasi yang menunjukkan kondisi nilai atribut yang sering ada bersamaan dalam satu kumpulan data.

2. Klasifikasi

Fungsi pembelajaran yang memetakan (mengklasifikasi) sebuah item data ke dalam salah satu dari beberapa kelas yang sudah didefinisikan.

3. *Clustering*

Melakukan pengelompokan data tanpa berdasarkan kelas data tertentu.

4. *Pendeteksian* perubahan dan deviasi

Berfokus pada penemuan perubahan yang paling signifikan di dalam data dari nilai-nilai yang telah diukur sebelumnya.

2.3.2. Teknik Data Mining

Menurut Daniel Larose (2005) dalam Algoritma Data Mining [KUS-09], jenis pekerjaan yang dapat dilakukan dengan teknik data mining yaitu:

1. Deskripsi

Terkadang peneliti dan analisis secara sederhana ingin mencoba mencari cara untuk menggambarkan pola dan kecenderungan yang terdapat dalam

data. Sebagai contoh, petugas pengumpulan suara tidak dapat menemukan keterangan atau fakta bahwa siapa yang tidak cukup profesional akan sedikit didukung dalam pemilihan presiden.

2. Estimasi

Estimasi hampir sama dengan klasifikasi, kecuali variabel target estimasi lebih ke arah numerik daripada ke arah kategori.

3. Prediksi

Prediksi hampir sama dengan klasifikasi dan estimasi, kecuali bahwa dalam prediksi nilai dari hasil akan ada di masa mendatang. Beberapa metode dan teknik yang digunakan dalam klasifikasi dan estimasi dapat pula digunakan (untuk keadaan yang tepat) untuk prediksi.

4. Klasifikasi

Dalam klasifikasi terdapat target variabel kategori. Sebagai contoh, klasifikasi dalam penelitian adalah mendiagnosis penyakit seorang pasien untuk mendapatkan termasuk kategori penyakitnya.

5. Pengklusteran

Kluster adalah kumpulan *record* yang memiliki kemiripan satu dengan yang lainnya dan memiliki ketidakmiripan dengan *record-record* dalam *kluster* lain. Pengklusteran berbeda dengan klasifikasi yaitu tidak adanya variabel target dalam pengklusteran.

6. Asosiasi

Tugas asosiasi dalam data mining adalah menemukan atribut yang muncul bersama dalam satu waktu.

2.3.3. Klasifikasi

Menurut Han dan Kamber (2000) klasifikasi adalah proses untuk menemukan suatu kumpulan model atau fungsi yang menggambarkan dan membedakan kelas data atau konsep, agar dapat menggunakan model tersebut untuk memprediksi kelas objek yang kelas labelnya tidak diketahui.

Klasifikasi merupakan suatu teknik yang dapat memberikan klasifikasi pada data baru dengan memanipulasi data yang telah diklasifikasi dan menggunakan hasilnya untuk memberikan sejumlah aturan. Aturan-aturan tersebut digunakan

pada data-data baru untuk diklasifikasi. Teknik ini menggunakan *supervised induction*, yang memanfaatkan kumpulan pengujian dari *record* yang terklasifikasi untuk menentukan kelas data baru [KUS-07].

Terdapat beberapa metode klasifikasi, antara lain *decision tree*, Bayesian, *fuzzy*, *neural network*, *support vektor machine* (SVM) dan *k-nearest neighbor*. Dibandingkan dengan beberapa metode klasifikasi lain yang rata-rata hanya dapat menetapkan satu kelas untuk sampel masukan, metode *fuzzy* klasifikasi memiliki kelebihan, dimana metode ini dapat menetapkan sebuah “vektor *fuzzy*” yang menunjukkan kekuatan *relative* dari masing-masing kelas. Contoh dari sistem klasifikasi *fuzzy* adalah mesin cuci *fuzzy*. Beberapa variabel/parameter mesin cuci ditentukan berdasarkan jumlah dan jenis pakaian. Keluaran atau informasi dari sistem klasifikasi ini digunakan untuk menentukan jenis *spin-dry* serta lembut atau kasar gesekan pakaian yang optimal [ALB-95].

2.3.4. K-Nearest Neighbor (KNN)

Algoritma *K-Nearest Neighbor* (KNN) merupakan metode pengklasifikasian data yang bekerja relatif dengan cara lebih sederhana dibandingkan dengan metode pengklasifikasi data lainnya. Algoritma ini mengklasifikasi data baru yang belum diketahui kelasnya dengan memilih data sejumlah k yang letaknya terdekat dari data baru tersebut. *Class* terbanyak dari data terdekat sejumlah k tersebut dipilih sebagai *class* yang diprediksikan untuk data yang baru. Nilai k umumnya ditentukan dalam jumlah ganjil untuk menghindari munculnya jumlah jarak yang sama dalam proses pengklasifikasian [JNI-96]. Dimana aturan keputusan yang didapat memberikan prosedur *non-parametrik* sederhana untuk tugas dari pola kelas yang sudah ada untuk pola input berdasarkan label kelas diwakili oleh nilai k terdekat (misalnya, dalam arti Euclidean) tetangga vektor [KEL-85].

Ketepatan algoritma KNN ini sangat dipengaruhi oleh ada atau tidaknya fitur-fitur yang tidak relevan, atau jika bobot fitur tersebut tidak setara dengan relevansinya terhadap klasifikasi. Riset terhadap algoritma ini sebagian besar membahas bagaimana memilih dan memberi bobot terhadap fitur, agar performa klasifikasi menjadi lebih baik. Algoritma K-NN ini memiliki konsistensi yang kuat.

K-Nearest Neighbor (K-NN) adalah suatu metode yang menggunakan algoritma *supervised* dimana hasil dari sampel uji yang baru diklasifikasikan berdasarkan mayoritas dari kategori pada K-NN. Tujuan dari algoritma ini adalah mengklasifikasi objek baru berdasarkan atribut dan sampel latih. pengklasifikasian tidak menggunakan model apapun untuk dicocokkan dan hanya berdasarkan pada memori. Klasifikasi menggunakan *voting* terbanyak di antara klasifikasi dari K objek. Algoritma K-NN menggunakan klasifikasi ketetanggaan sebagai nilai prediksi dari sample uji yang baru. Dekat atau jauhnya tetangga biasanya dihitung berdasarkan jarak *Euclidean*.

Algoritma metode K-NN sangatlah sederhana, bekerja dengan berdasarkan pada jarak terpendek dari sample uji ke sample latih untuk menentukan K-NN nya. Setelah mengumpulkan K-NN, kemudian diambil mayoritas dari K-NN untuk dijadikan prediksi dari sample uji.

2.3.4.1. Proses K-Nearest Neighbor (KNN)

Langkah awal yang dilakukan pada metode ini adalah menghitung jarak antara *record* baru dengan tiap *record* pada data latih. Namun sebelum dihitung jaraknya, setiap record dilakukan normalisasi terlebih dahulu. Normalisasi yang digunakan pada penelitian ini adalah *Min-max normalization*. Untuk menghitung nilai normalisasi ditunjukkan oleh persamaan (2-4).

$$V[i][j]' = (V[i][j] - \min[j]) / \text{Range}[j] \quad (2-4)$$

Dimana:

$\text{Range}[j] = \max[j] - \min[j]$

$V[i][j]'$ = hasil normalisasi yang nilainya berkisar antara 0 dan 1

$V[i][j]$ = nilai atribut yang akan dinormalisasi

$\min[j]$ = nilai minimum dari suatu atribut j

$\max[j]$ = nilai maksimum dari suatu atribut j

Selanjutnya, atribut yang telah di normalisasi tersebut digunakan pada proses klasifikasi. Awalnya dicari nilai bobot tiap atribut dari setiap kelas yang ada. Menghitung bobot dengan keanggotaan kelas yang ada pada data latih. Persamaan 2.4 untuk pencarian rata-rata bobot tiap atribut:

$$w = (x_1 + x_2 + \dots + x_n) / n \quad (2-5)$$

dimana : w = bobot tiap inputan

x_n = data inputan ke- n

n = jumlah data

Jika dikehendaki bisa juga memakai median dari *cluster* tersebut. Jika rata-rata (*mean*) bukan satu-satunya ukuran yang bisa dipakai.

Setelah itu, dihitung jarak terdekat antara data uji dengan data latih terlebih dahulu. Untuk menghitung jarak digunakan fungsi jarak *Euclidean* yang ditunjukkan oleh persamaan (2-6).

$$X_1 = (X_{11}, X_{12}, \dots, X_{1n})$$

$$X_2 = (X_{21}, X_{22}, \dots, X_{2n})$$

$$d(x_1, x_2) = \sqrt{\sum_{r=1}^n (a_r(x_1) - a_r(x_2))^2} \quad (2-6)$$

di mana x_1 dan x_2 merupakan dua *record* dengan n atribut. Persamaan (2-6) menghitung jarak antara x_1 dan x_2 , dengan tujuan untuk menentukan perbedaan antara nilai-nilai atribut pada *record* x_1 dan x_2 . Setelah diketahui jarak antar *record*, kemudian diambil sebanyak k tetangga terdekat untuk memprediksi label kelas dari *record* baru menggunakan label kelas tetangga. Prediksi tersebut diketahui dengan fungsi kombinasi yang dibahas pada subbab 2.3.4.2.

2.3.4.2. Fungsi Kombinasi

Untuk memberikan keputusan klasifikasi bagi *record* baru, dilakukan kombinasi terhadap *record* yang serupa, dengan fungsi kombinasi. Terdapat dua jenis fungsi kombinasi *unweighted voting* dan *weighted voting*. Pada *unweighted voting*, label kelas untuk *record* yang baru, dipilih berdasarkan label kelas terbanyak (mayoritas) yang dimiliki tetangga. Sedangkan *weighted voting* dilakukan dengan memberikan bobot terhadap beberapa tetangga yang dekat dengan *record* yang baru. Pemberian bobot ini dapat memberikan pengaruh lebih dalam menentukan label kelas. *Weighted* ditunjukkan oleh persamaan (2-7).

$$\text{Weight}(y) = \frac{1}{\text{DISTANCE}(y,a)^2} \quad (2-7)$$

Dimana:

y = *neighbor*

a = *new Record*

DISTANCE(neighbor, newRecord) merupakan jarak antara *record* yang baru dengan tetangganya. Persamaan (2-7) dapat menghitung bobot dari semua tetangganya, dengan menjumlahkan bobot dari tetangga yang memiliki label kelas yang sama. Label kelas dari *record* yang baru adalah label kelas dari *record* yang jumlah bobot tetangganya paling besar [MOR-09].

2.3.4.3. Algoritma K-Nearest Neighbor

Berikut ini adalah algoritma *K-Nearest Neighbor* menurut James M Keller (1985):

$W = \{x_1, x_2, \dots, x_n\}$

BEGIN

Input y , of unknown classification

Set K , $1 \leq K \leq n$

Initialize $i = 1$

DO UNTIL (K -nearest neighbors found)

 Compute distance from y to x_i

```

IF ( $i \leq K$ ) THEN
    Include  $x_i$  in the set of K-nearest neighbor
ELSE IF ( $x_i$  closer to  $y$  than any previous nearest neighbor) THEN
    Delete the farthest of the K-nearest neighbors
    Include  $x_i$  in the set of K-nearest neighbors
END IF
Increment  $i$ 
END DO UNTIL
Determine the majority class represented in the set of K-Nearest neighbor
IF (a tie exists) THEN
    Compute sum of distance of neighbor in each class which tied
    IF (no tie occurs) THEN
        Classify  $y$  in the class of minimum sum
    ELSE
        Classify  $y$  in the class of last minimum found
    END IF
ELSE
    Classify  $y$  in the majority class
END IF
END

```

2.4. Fuzzy K-Nearest Neighbor Classifier

2.4.1. Pengertian Fuzzy K-Nearest Neighbor (FK-NN)

Fuzzy K-Nearest Neighbor merupakan metode klasifikasi yang menggabungkan teknik fuzzy klasifikasi dengan *k-Nearest Neighbor classifier*. Algoritma *Fuzzy K-Nearest Neighbor* memberikan nilai keanggotaan kelas pada vektor sampel dan bukan menempatkan vektor pada kelas tertentu. Keuntungannya adalah nilai-nilai keanggotaan vektor seharusnya memberikan tingkat jaminan pada hasil klasifikasi. FK-NN (*fuzzy k-nearest neighbor*) merupakan salah satu proses yang penting dari ekstraksi ciri. Proses ini menghitung derajat keanggotaan fuzzy dengan menghitung jarak antara vektor antar kelas maupun dalam kelas. Dalam prosesnya jarak antar vektor dihitung dengan menggunakan fungsi jarak *Euclidean distance*. Setelah jaraknya diketahui maka selanjutnya adalah mengetahui k tetangga yang berdekatan, proses ini biasa juga disebut dengan KNN (*K-Nearest Neighbour*). Setelah itu kita hitung nilai keanggotaan fuzzy vektor tersebut, oleh sebab itu metode ini dinamakan FK-NN. Teori himpunan fuzzy mengeneralisasi teori K-NN klasik dengan mendefinisikan nilai keanggotaan sebuah data pada masing-masing kelas [PRA-12].

Menurut Han, J. H., Pengklasifikasi *K-Nearest Neighbor* (K-NN) biasanya digunakan dalam pengenalan pola. Sebuah sampel input ditetapkan ke dalam kelas yang diwakili oleh mayoritas dari *K-Nearest Neighbors*. Bagaimanapun, sekali sebuah input sampel dimasukkan ke dalam sebuah kelas, tak ada petunjuk dari kekuatannya dari keanggotaan dalam kelas itu. *Fuzzy* pengklasifikasi K-NN, sebuah kombinasi dari *fuzzy logic* dan K-NN *classifier*, dirancang untuk menyelesaikan masalah di atas. Terdiri dari dua langkah: penamaan *fuzzy* yang menghitung vektor *fuzzy* dari sampel pengujian dan klasifikasi *fuzzy* yang menghitung vektor *fuzzy* dari sampel input. Dalam klasifikasi *fuzzy*, kita menentukan keanggotaan *fuzzy* μ_{uc} untuk sebuah sampel input χ_u pada setiap kelas c sebagai sebuah kombinasi linear dari vektor *fuzzy* dari sampel pengujian *K-Nearest Neighbor*:

$$\mu_{NC} = \frac{\sum_{i=1}^k W_i \mu_{ic}}{\sum_{i=1}^k W_i} \quad (2-8)$$

Dimana μ_{ic} adalah keanggotaan *fuzzy* dari sampel pengujian χ_i dalam kelas c , χ_i adalah satu dari sampel-sampel *K-Nearest Neighbor*, dan w_i adalah berat kebalikannya sebanding ke jarak d antara χ_i dan χ_u :

$$w_i = d_{iu}^{-2} \quad (2-9)$$

2.4.2. Algoritma *Fuzzy K-Nearest Neighbor* (FK-NN)

Dasar dari algoritma ini adalah untuk menetapkan nilai keanggotaan sebagai fungsi jarak vektor dari *K-Nearest Neighbor* dan keanggotaan tetangga mereka di kelas-kelas yang memungkinkan. Berikut ini adalah algoritma *fuzzy* K-NN [KEL-85]:

```
BEGIN
  Input x, of unknown classification
  Set K,  $1 \leq K \leq n$ 
  Initialize i = 1
  DO UNTIL (K-nearest neighbors to x found)
    Compute distance from x to  $x_i$ 
```

```

IF (i ≤ K) THEN
    Include xi in the set of K-nearest neighbor
ELSE IF (xi closer to x than any previous nearest neighbor) THEN
    Delete the farthest of the K-nearest neighbors
    Include xi in the set of K-nearest neighbors
END IF
END DO UNTIL
Set i = 1
DO UNTIL (x assigned membership in all classes)
    Compute ui(x) using (2.10)
    Increment i
END DO UNTIL
END

```

Berikut ini merupakan rumus (2-10), yang merupakan fungsi rumus logika *fuzzy* untuk menentukan nilai keanggotaan kelas:

$$u_i(x) = \frac{\sum_{j=1}^K U_{ij} \left(\frac{1}{|x-x_j|^{m-1}} \right)}{\sum_{j=1}^K \left(\frac{1}{|x-x_j|^{2/(m-1)}} \right)} \quad (2-10)$$

Dimana U_{ij} adalah keanggotaan *fuzzy* dari sampel pengujian x , x_j adalah satu dari sampel-sampel *Fuzzy K-Nearest Neighbor*, dan m adalah berat kebalikannya sebanding ke jarak di antara x_j dan x .

Seperti yang terlihat pada (2-10), keanggotaan ditugaskan dari x dipengaruhi oleh kebalikan dari jarak dari tetangga terdekat dan keanggotaan kelas tersebut. Jarak terbalik berfungsi untuk bobot keanggotaan vektor, jika jarak lebih dekat ataupun jarak lebih jauh dari vektor dalam pertimbangan. Data yang diujikan dapat diberikan keanggotaan kelas dalam beberapa cara. Pertama, dapat diberikan keanggotaan lengkap di kelas yang sama dan non keanggotaan di kelas lainnya. Alternatif lainnya adalah untuk menetapkan keanggotaan sampel berdasarkan jarak dari rata-rata kelas atau berdasarkan jarak dari kelas data uji itu sendiri dengan kelas lainnya yang memiliki jarak terdekat dengan data uji. Namun dalam penelitian ini, untuk menghitung nilai keanggotaan *fuzzy* digunakan metode trapezium-segitiga.

Variabel (m) menentukan berapa banyak jarak diberi bobot saat menghitung kontribusi masing-masing tetangga dengan nilai keanggotaan. Jika m adalah dua,

maka kontribusi dari setiap titik tetangga tertimbang dan jarak relatif mereka dari sudut yang diklasifikasikan berpengaruh kurang. Sebagai m mendekati satu, tetangga dekat yang berbobot jauh lebih berat daripada mereka lebih jauh, yang memiliki efek mengurangi jumlah poin yang berkontribusi terhadap nilai keanggotaan dari titik yang diklasifikasikan [KEL-85].

2.5. Akurasi Hasil Pengujian

Akurasi merupakan seberapa dekat suatu angka hasil pengukuran terhadap angka. Dalam penelitian ini akurasi tingkat risiko PJK dihitung dari jumlah prediksi yang benar dibagi dengan jumlah data uji. Tingkat akurasi diperoleh dengan perhitungan sesuai dengan persamaan (2-11) [NUG-06].

$$\text{Tingkat akurasi} = \frac{\sum \text{data uji benar}}{\sum \text{total data uji}} \quad (2-11)$$

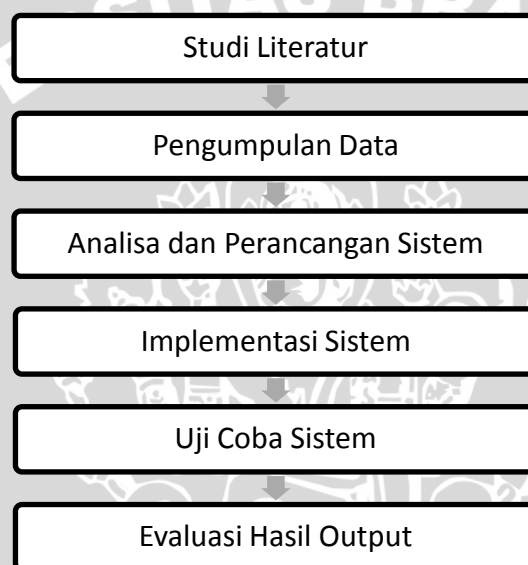
$$\text{Akurasi (\%)} = \frac{\sum \text{data uji benar}}{\sum \text{total data uji}} \times 100\% \quad (2-12)$$

Jumlah prediksi benar adalah jumlah *record* data uji yang diprediksi kelasnya menggunakan metode klasifikasi dan hasilnya sama dengan kelas sebenarnya. Sedangkan jumlah total prediksi adalah jumlah keseluruhan *record* yang diprediksi kelasnya (seluruh data uji).

BAB III

METODOLOGI DAN PERANCANGAN

Pada bab metodologi dan perancangan berisi mengenai pembahasan metodologi yang digunakan dalam penelitian serta perancangan sistem untuk memprediksi tingkat risiko PJK dengan menggunakan *Fuzzy K-nearest Neighbor*. Secara umum, tahapan dari penelitian ini adalah ditunjukkan pada gambar 3.1.



Gambar 3. 1 Diagram Alir Penelitian

Dari gambar 3.1, secara lebih rinci tahapan dari penelitian adalah sebagai berikut:

1. Studi Literatur

Mempelajari literatur yang berhubungan dengan penyakit jantung koroner dan metode *Fuzzy K-Nearest Neighbor* (FK-NN) dengan membaca buku maupun *browsing*.

2. Pengumpulan Data

Pada tahap ini, data yang digunakan berupa dataset rekam medik pasien PJK.

3. Analisa dan Perancangan Sistem

Menganalisa dan melakukan perancangan sistem dengan metode *Fuzzy K-Nearest Neighbor* (FK-NN).

4. Implementasi Sistem

Membangun perangkat lunak berdasarkan analisis dan perancangan yang telah dilakukan (implementasi).

5. Uji Coba Sistem

Melakukan uji coba terhadap perangkat lunak.

6. Evaluasi hasil Ouput

Mengevaluasi output yang dihasilkan oleh sistem.

3.1. Studi Literatur

Dalam penelitian ini dibutuhkan studi literatur untuk merealisasikan tujuan dan penyelesaian masalah. Teori-teori mengenai mengenai penyakit jantung koroner, tingkat risiko penyakit jantung koroner, himpunan *fuzzy*, metode *K-Nearest Neighbor* dan metode gabungan *Fuzzy K-nearest Neighbor* (FK-NN) sebagai dasar penelitian yang diperoleh dari buku, jurnal dan browsing dari internet serta literatur lain yang berkaitan seperti yang telah dijelaskan pada bab 2. Kemudian data yang diperoleh diubah sehingga dapat digunakan untuk analisis. Setelah dianalisis maka dapat diimplementasikan ke dalam program.

3.2. Pengumpulan Data

Data yang digunakan pada penelitian ini merupakan data dari penelitian diagnosa Penyakit Jantung Koroner (PJK) yang dilakukan oleh Wahyuni (2011), dimana data ini diperoleh dari data rekam medik pasien RSUD Saiful Anwar, RSI Unisma Malang dan laboratorium cek fisik kesehatan. Pada dataset Penyakit Jantung Koroner (PJK) yang digunakan adalah: kadar LDL, umur, kadar HDL, kolesterol total, kadar trigliserida dan tekanan darah sistolik. Dimana data yang diperoleh sebanyak 100 record, tanpa adanya data yang memiliki missing value.

3.3. Analisa Dan Perancangan Sistem

3.3.1. Deskripsi Sistem

Sistem yang dibuat merupakan perangkat lunak yang mengimplementasikan algoritma *Fuzzy K-Nearest Neighbor* (FK-NN) untuk prediksi tingkat risiko penyakit jantung koroner. Dimana pada sistem ini akan diketahui tingkat risiko penyakit

jantung koroner yang diderita pasien. Hasil prediksi tersebut berupa nilai risiko seseorang terhadap penyakit jantung koroner. Hasil nilai yang diperoleh dapat disimpulkan, yaitu semakin tinggi nilai risikonya, maka semakin tinggi pula tingkat risiko seseorang terkena penyakit jantung koroner dan begitu pula sebaliknya.

Pada tahap awal, sistem memiliki data-data faktor risiko jantung koroner. Data faktor risiko jantung koroner yang didapatkan semuanya sudah berupa data numerik, sehingga bisa langsung dilanjutkan untuk proses pembelajaran dengan metode *Fuzzy K-Nearest Neighbor* (FK-NN).

Dataset ini terdiri dari 6 atribut (termasuk atribut kelas). Atribut-atribut tersebut meliputi:

1. Usia : usia penderita pada saat pemeriksaan
2. LDL : *Low Density Lipoprotein*
3. Kolesterol total
4. HDL : *High Density Lipoprotein*
5. Trigliserida
6. Tekanan darah sistolik

Dataset Tingkat Risiko Penyakit Jantung Koroner (PJK) ini terdiri dari 100 record dan tidak terdapat *missing value*. Atribut-atribut dalam *dataset* mempunyai nilai rentang yang berbeda. Dari 100 record tersebut, terdapat rincian sebagai berikut:

1. Sangat rendah sebanyak 2 *record*
2. Rendah sebanyak 15 *record*
3. Sedang sebanyak 49 *record*
4. Tinggi sebanyak 31 *record*
5. Sangat tinggi sebanyak 3 *record*

Perangkat lunak ini akan menguji keakuratan hasil prediksi tingkat risiko penyakit jantung koroner terhadap data sebenarnya. Selain itu, parameter uji juga berkaitan dengan nilai k (tetangga) yang berpengaruh terhadap tingkat akurasi.

Perancangan data uji yang dilakukan menggunakan variasi data uji dengan data latih. Dimana dalam tiap pengujian digunakan komposisi jumlah data latih yang berubah yaitu, 40, 50, 60 dan 70 data latih, sedangkan data uji dengan jumlah tetap 30 data. Komposisi jumlah atribut tiap data latih dan data uji ditampilkan pada tabel berikut:

Tabel 3. 1 Komposisi Kategori Data Uji (30 data)

Kategori	Jumlah data
Sangat Rendah	1
Rendah	4
Sedang	14
Tinggi	10
Sangat Tinggi	1

Tabel 3. 2 Komposisi Kategori Data Latih (40 data)

Kategori	Jumlah data
Sangat Rendah	1
Rendah	5
Sedang	20
Tinggi	12
Sangat Tinggi	2

Tabel 3. 3 Komposisi Kategori Data Latih (50 data)

Kategori	Jumlah data
Sangat Rendah	1
Rendah	6
Sedang	26
Tinggi	15
Sangat Tinggi	2

Tabel 3. 4 Komposisi Kategori Data Latih (60 data)

Kategori	Jumlah data
Sangat Rendah	1
Rendah	8
Sedang	30
Tinggi	19
Sangat Tinggi	2

Tabel 3. 5 Komposisi Kategori Data Latih (70 data)

Kategori	Jumlah data
Sangat Rendah	1
Rendah	11
Sedang	35
Tinggi	21
Sangat Tinggi	2

3.3.2. Perancangan Sistem

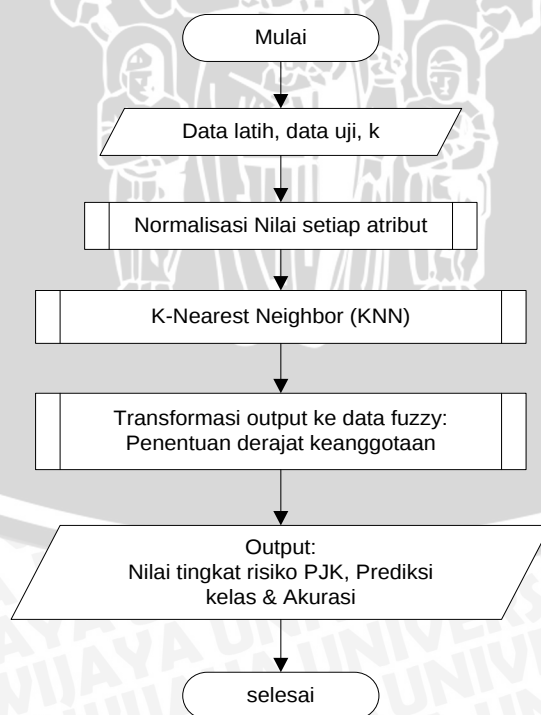
Pada sub bab ini, akan dijelaskan mengenai perancangan proses secara keseluruhan dari sistem ini. Secara umum, proses dari sistem ini adalah terbagi dari dua proses yaitu rancangan prediksi sistem dan pengujian sistem.

3.3.2.1. Perancangan Sistem Prediksi

Tahap prediksi sistem FK-NN untuk prediksi tingkat risiko PJK, langkah-langkahnya adalah sebagai berikut:

1. Menginputkan data faktor risiko PJK (umur, kadar LDL, kadar kolesterol, kadar HDL, kadar trigliserida dan tekanan darah sistolik) dan input nilai k . Pada proses prediksi hanya satu data saja yang akan diuji.
2. Melakukan perhitungan normalisasi untuk nilai setiap atribut.
3. Melakukan perhitungan k -NN.
4. Transformasi output data ke dalam data *fuzzy*.
5. Didapatkanlah hasil prediksi kelas PJK dan nilai tingkat risiko.

Proses prediksi sistem FK-NN digambarkan dalam diagram alir seperti pada gambar 3.1.



Gambar 3. 2 Alur Proses Pengujian Sistem

Berdasarkan Gambar 3.2, proses pada sistem meliputi tiga proses utama, yaitu:

1. Normalisasi nilai setiap atribut

Merupakan proses transformasi berupa normalisasi terhadap nilai-nilai atribut untuk membakukan skala pengaruh yang ada pada atribut, terhadap hasil. Proses ini dijelaskan pada subbab 3.3.2.1.

2. *K-Nearest Neighbor*

Merupakan proses klasifikasi pada *dataset*, dimana dataset ini di kelaskan ke dalam 5 kelas yaitu sangat rendah, rendah, normal, tinggi dan sangat tinggi. Merupakan proses klasifikasi data uji menggunakan KNN dengan penambahan bobot dan *voting* terhadap *k* tetangga terdekat. Proses ini dijelaskan pada subbab 3.3.2.2.

3. Transformasi ke data *fuzzy*

Proses ini mengubah nilai kuantitatif pada setiap atribut dalam *dataset* menjadi derajat keanggotaan *fuzzy* beserta linguistiknya berdasarkan fungsi keanggotaan atribut. Proses ini dijelaskan pada subbab 3.3.2.3.

3.3.2.2. Normalisasi nilai setiap atribut

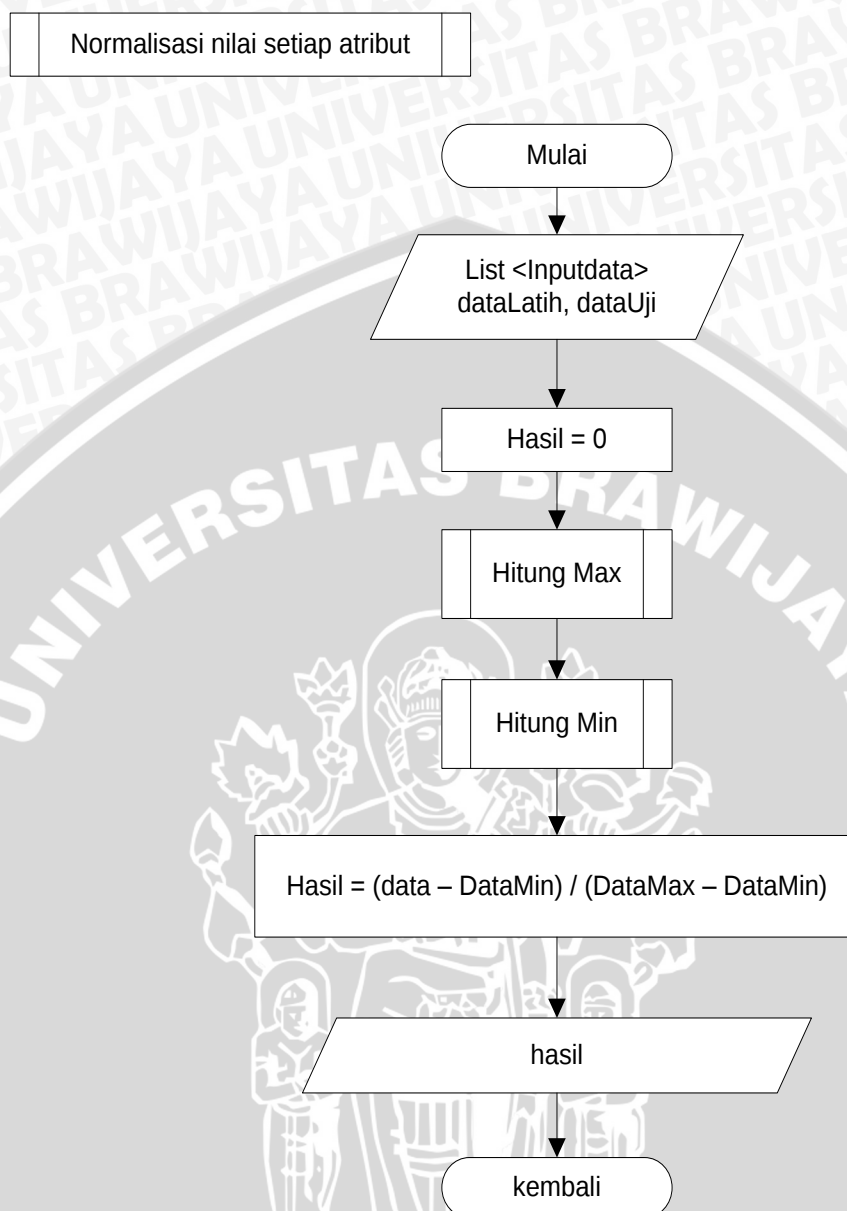
Proses ini merupakan normalisasi terhadap nilai-nilai atribut untuk membakukan skala pengaruh yang ada pada atribut, terhadap hasil. Untuk mencari hasil yang telah dinormalisasikan digunakan rumus (2.4).Selanjutnya, atribut yang telah di normalisasi tersebut digunakan pada proses klasifikasi. Pada proses normalisasi terdapat proses hitung nilai max dan hitung nilai min. Alur proses dari sistem ditampilkan oleh Gambar 3.3.

3.3.2.2.1. Hitung Max

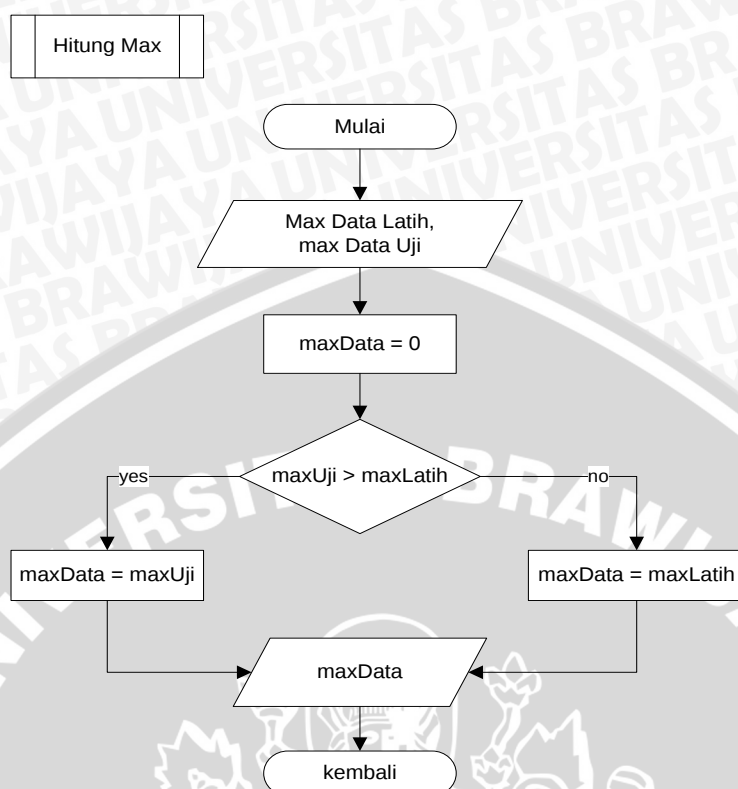
Proses hitung max merupakan proses mencari nilai *maksimum* data setiap atribut. Alur proses dari sistem ditampilkan oleh Gambar 3.4.

3.3.2.2.2. Hitung Min

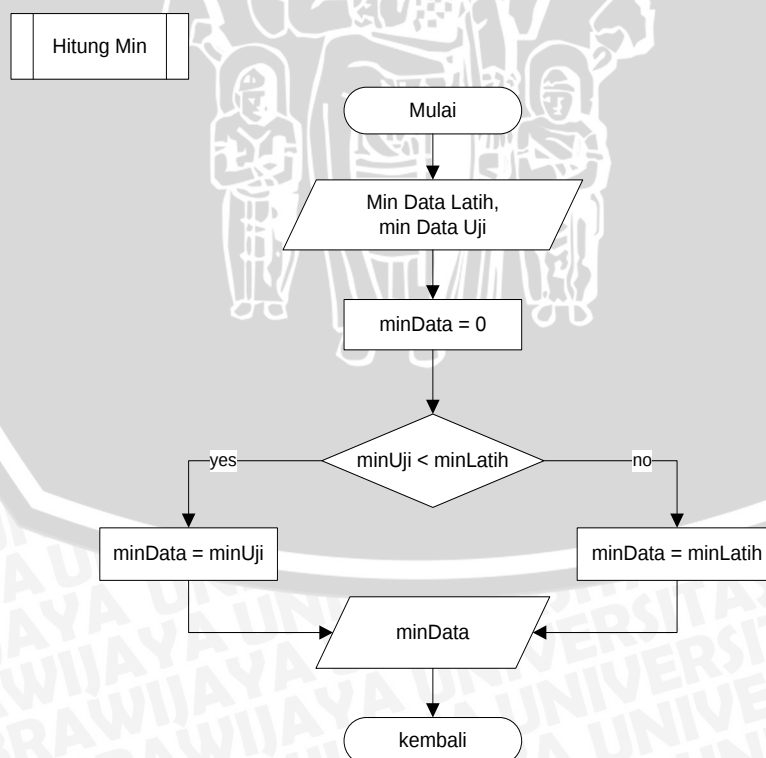
Proses hitung min merupakan proses mencari nilai *minimum* data setiap atribut. Alur proses dari sistem ditampilkan oleh Gambar 3.5.



Gambar 3. 3 Alur Proses Normalisasi setiap atribut



Gambar 3. 4 Alur Proses Hitung Max



Gambar 3. 5 Alur Proses Hitung Min

3.3.2.3. Klasifikasi K-Nearest Neighbor (K-NN)

Proses ini mengklasifikasikan dataset ke dalam 5 kelas sesuai dengan data Latih. Dimana setiap kelas memiliki anggota yang terdapat dalam *range* yang sama sesuai perhitungan data. Pada Klasifikasi ini juga dihitung *distance* atau jarak similarity dengan menggunakan *Euclidean Distance*, *weighted vote* dan *weighted* sejumlah k.

Langkah-langkah dalam proses klasifikasi dengan K-Nearest Neighbor adalah sebagai berikut:

1. Masukan (input) berupa data latih dan data uji yang telah dinormalisasi.
2. Proses untuk menghitung *distance* (jarak) antara *record* data uji dengan nilai *record* kelas pada data uji dengan *record* kelas pada data latih.
3. Proses *weighted vote*, dimana perhitungan nilai *weighted* sesuai dengan persamaan 2.7.
4. Proses *weight* sejumlah k, dimana pada proses ini diambil k *record* dengan nilai *weighted* terbesar yang akan digunakan sebagai tetangga.
5. Keluaran (output) berupa prediksi kelas dari data uji.

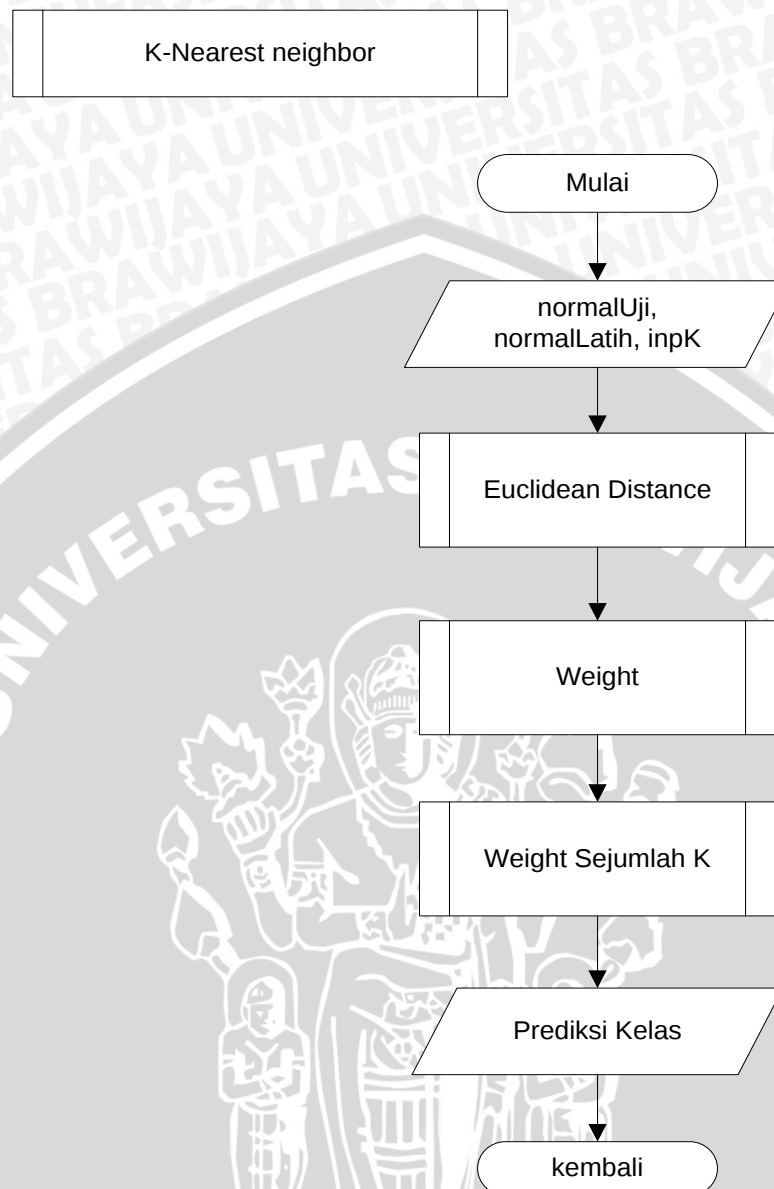
Flowchart untuk proses K-Nearest Neighbor (K-NN) ditampilkan oleh Gambar 3.6.

3.3.2.3.1. Euclidean Distance

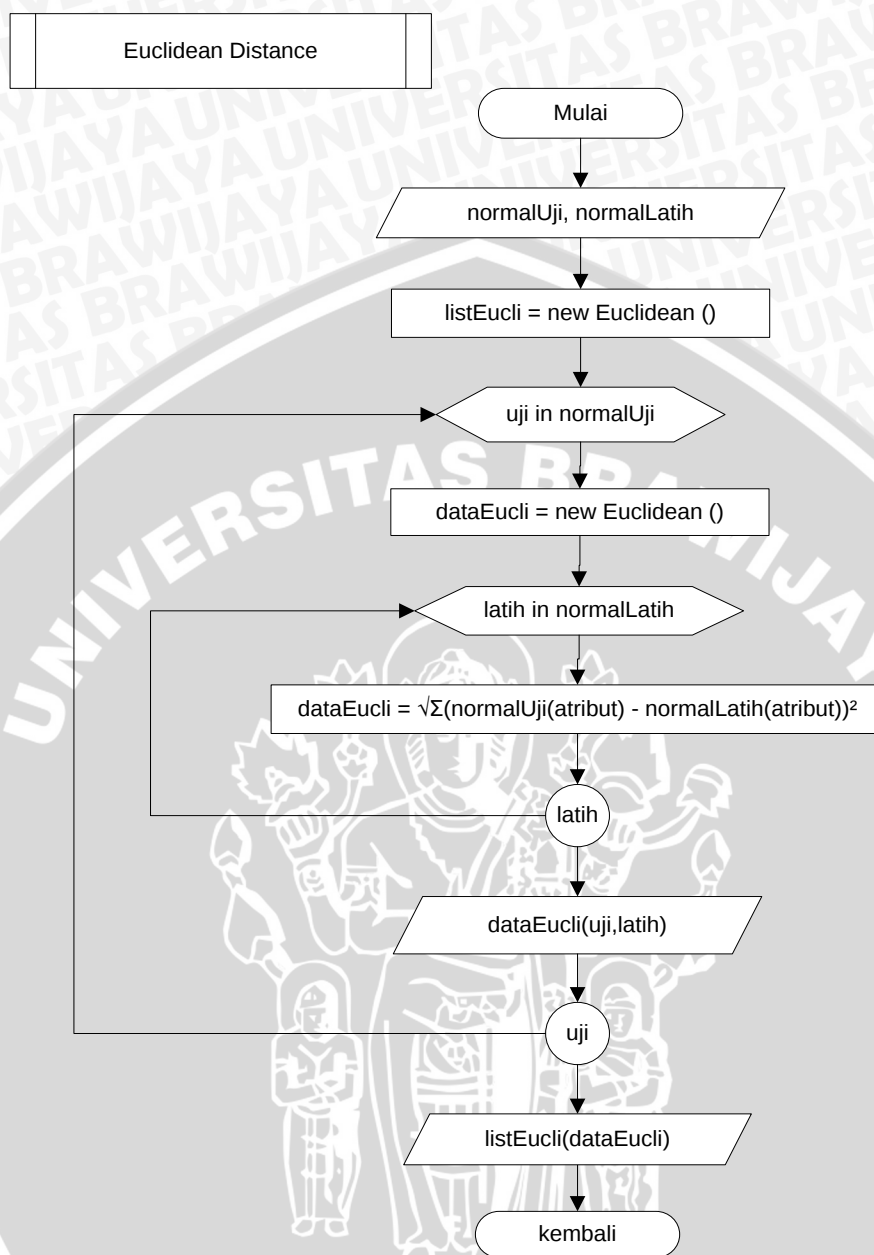
Proses ini untuk menghitung *distance* (jarak) antara *record* data uji dengan *record* data latih. Dimana untuk mendapatkan nilai jarak terdekat, antara data latih dan data uji, dihitung dari setiap atribut yaitu atribut umur, LDL, Kolesterol, LDL, *Trigliserida* dan *Sistolik*. *Flowchart* untuk proses *Euclidean Distance* ditampilkan oleh Gambar 3.7.

3.3.2.3.2. Weight

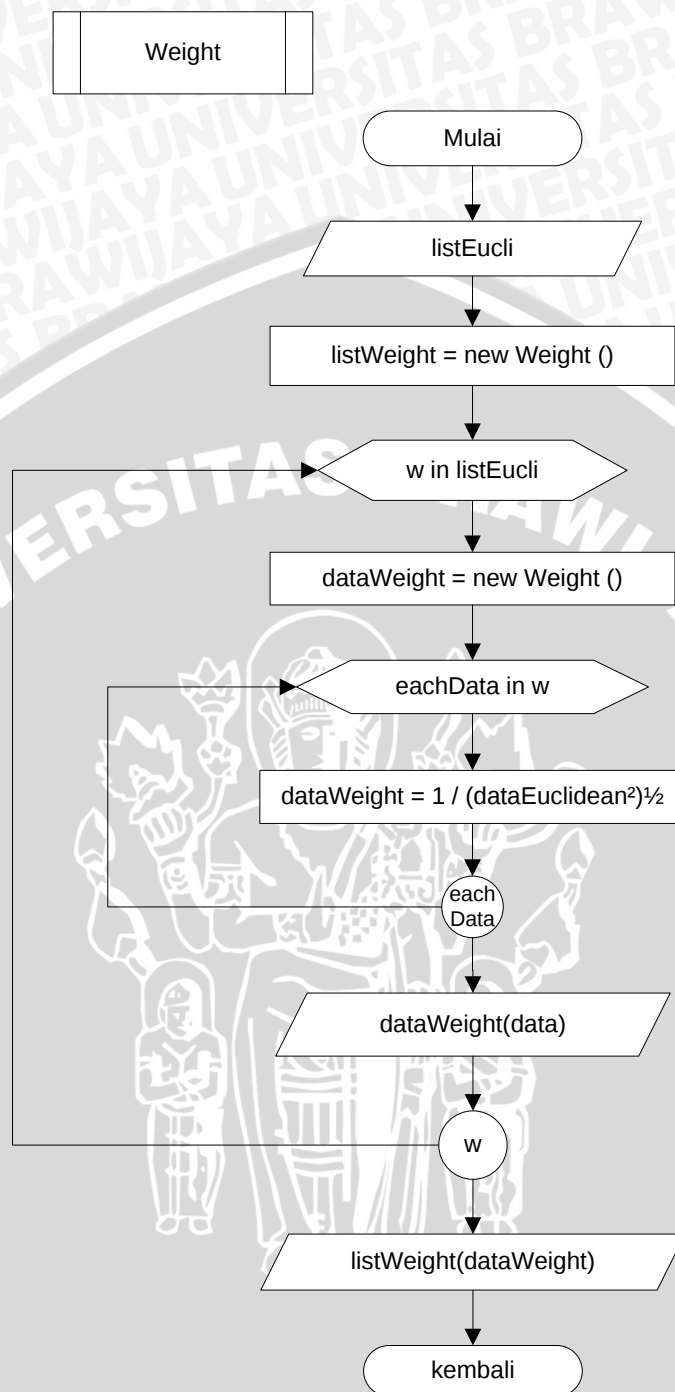
Proses ini untuk meminimalisir nilai dari *Euclidean Distance*. Dimana untuk mendapatkan nilai *weighted*, dapat digunakan persamaan 2.7. *Flowchart* untuk proses *Weight* ditampilkan oleh Gambar 3.8.



Gambar 3. 6 Alur Proses K-Nearest Neighbor



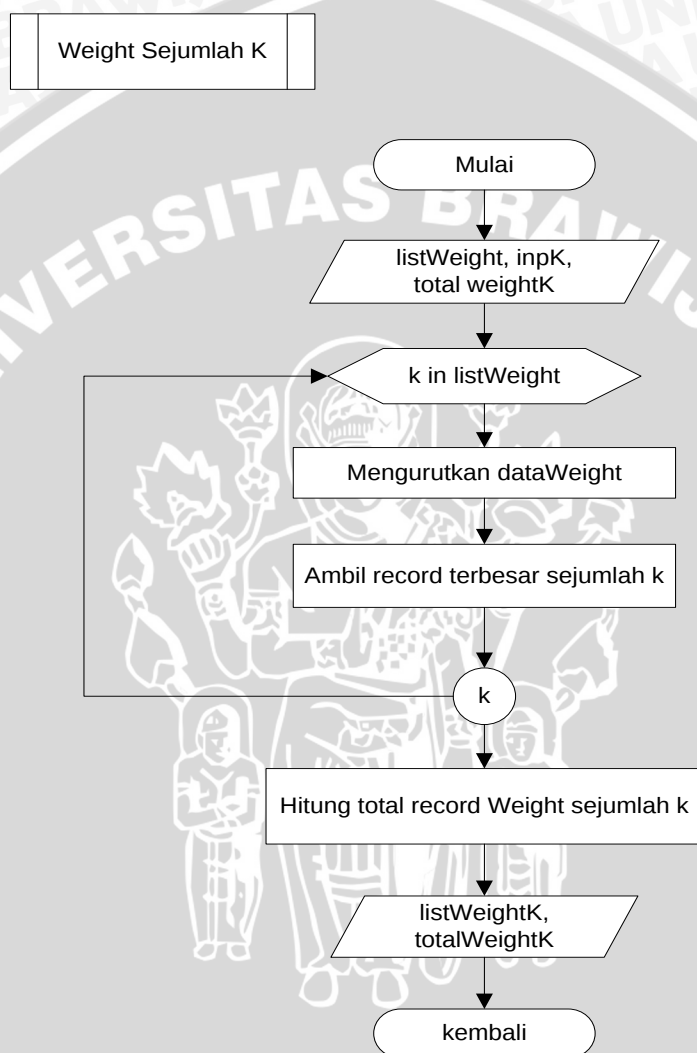
Gambar 3. 7 Alur Proses Euclidean Distance



Gambar 3. 8 Alur Proses Weight

3.3.2.3.3. Weight sejumlah K

Proses ini didapat dengan mengambil nilai *weight* sesuai dengan nilai *k* yang di inputkan *user*, setelah itu data disorting dan diambil data terbesar sejumlah nilai *k*. *Flowchart* untuk proses *Weight* sejumlah *K* ditampilkan oleh Gambar 3.9.

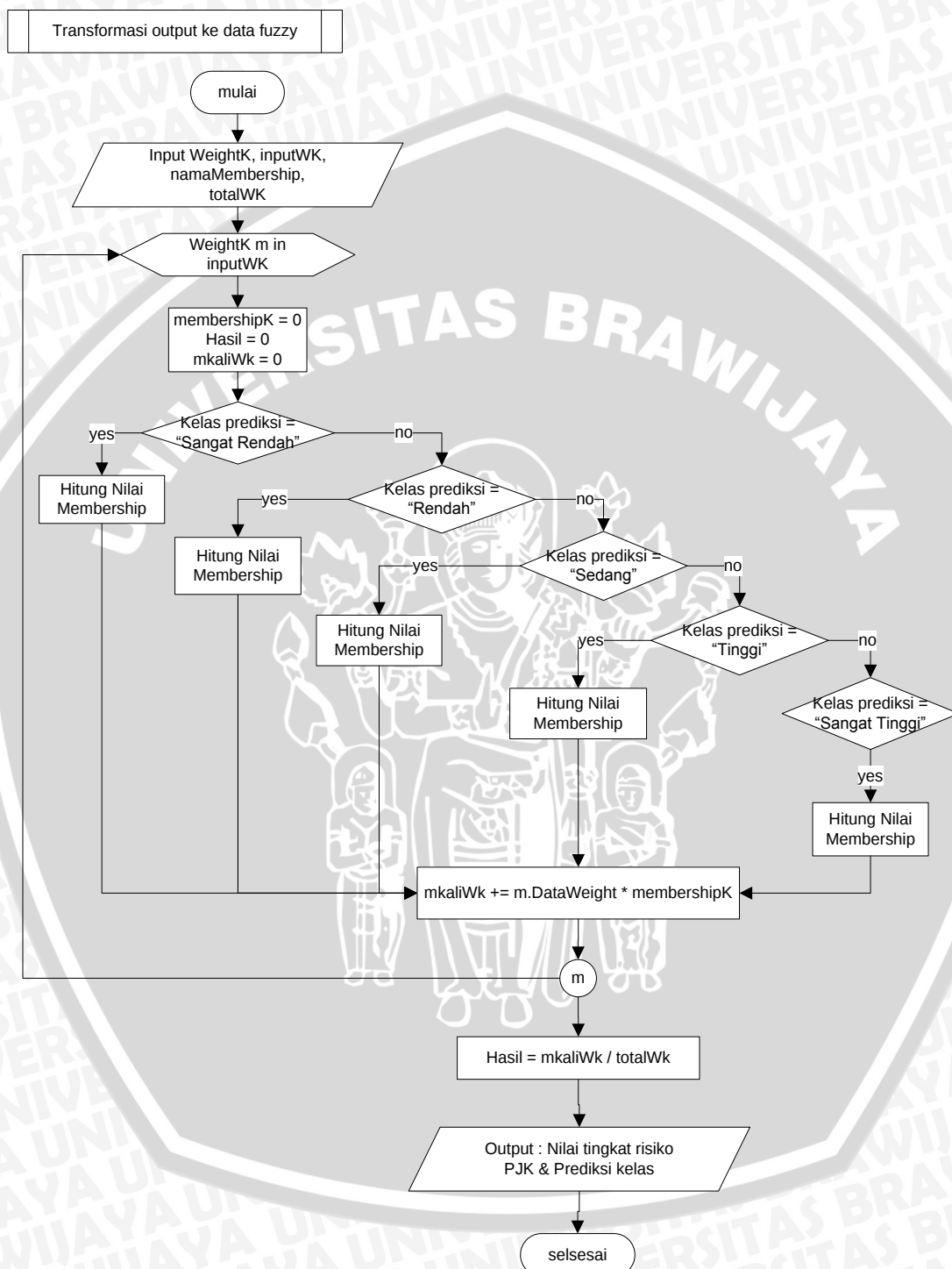


Gambar 3. 9 Alur Proses Weight sejumlah k

3.3.2.4. Transformasi ke data fuzzy

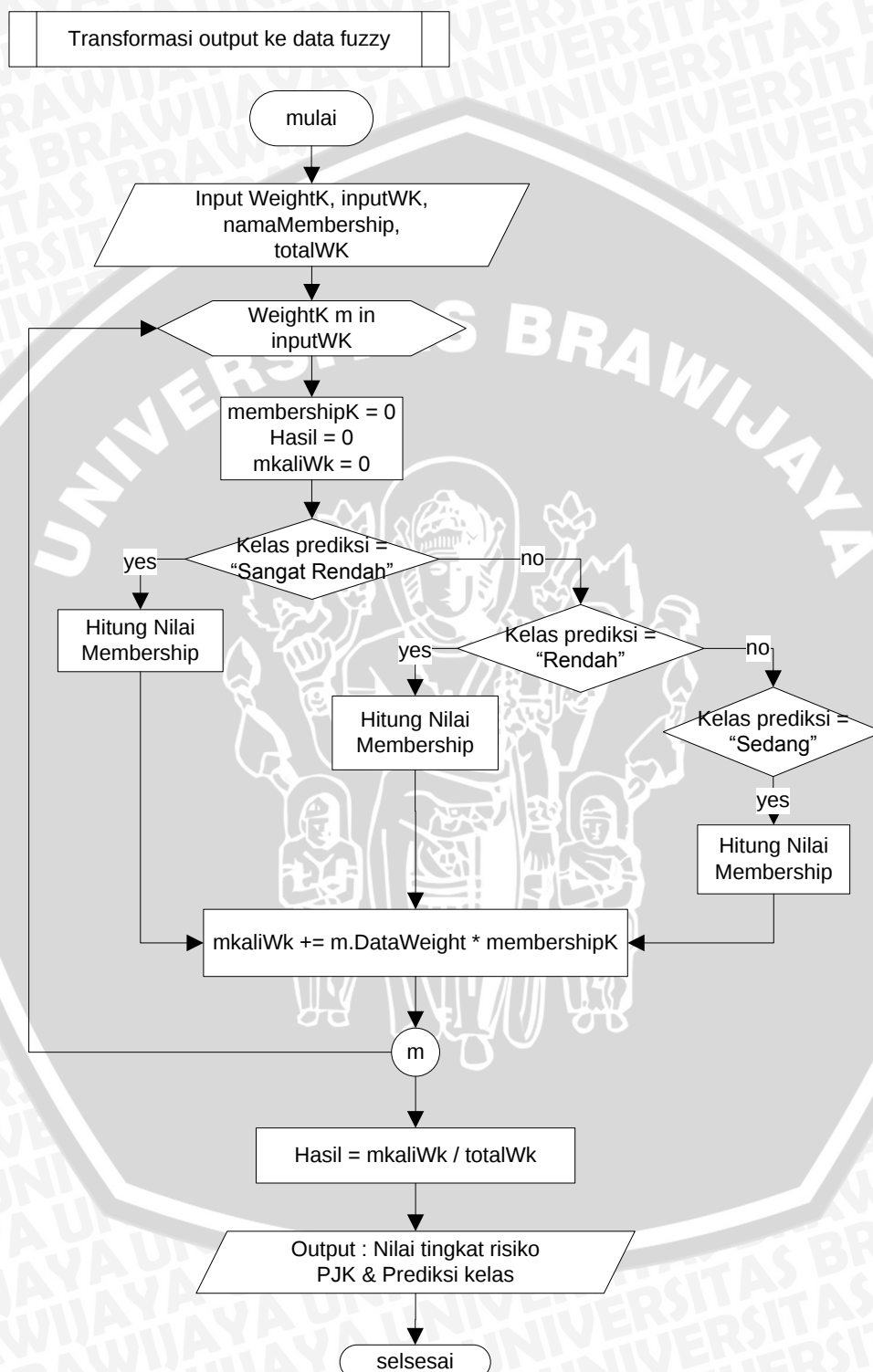
Proses ini mengubah nilai kuantitatif pada setiap atribut dalam *dataset* menjadi derajat keanggotaan *fuzzy* beserta *linguistiknya* berdasarkan fungsi keanggotaan atribut. Alur proses transformasi ke data *fuzzy* terdiri dari 4 macam, dimana setiap alurnya terdapat perbedaan *membership*. Alur proses untuk *membership* dengan 5

kelas (sangat rendah, rendah, sedang, tinggi dan sangat tinggi) dapat dilihat pada gambar 3.10.



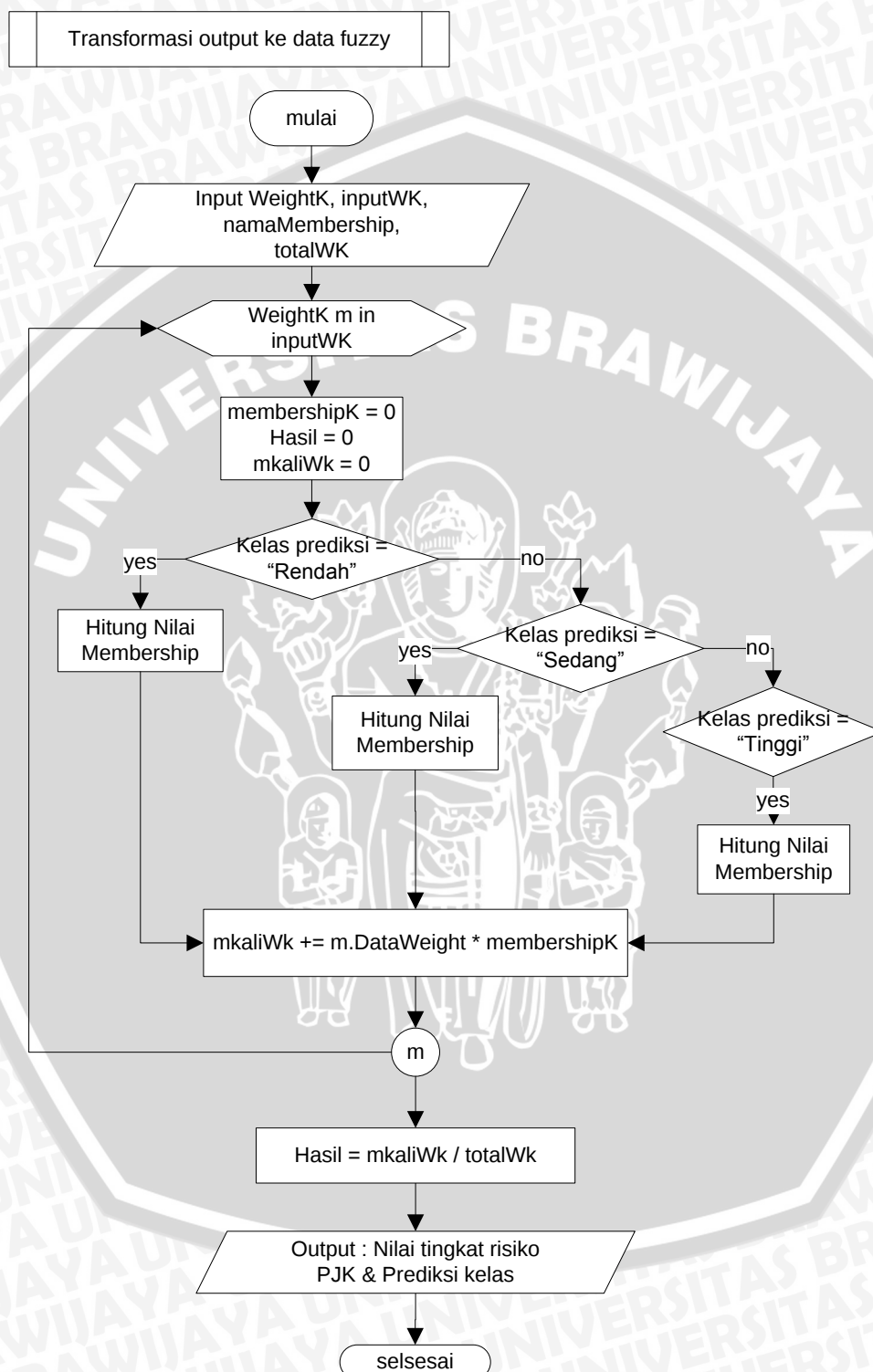
Gambar 3. 10 Alur Proses Transformasi Ke Data Fuzzy

Alur proses transformasi ke data *fuzzy* kedua, yaitu untuk *membership* dengan 3 kelas (sangat rendah, rendah, sedang) dapat dilihat pada gambar 3.11.



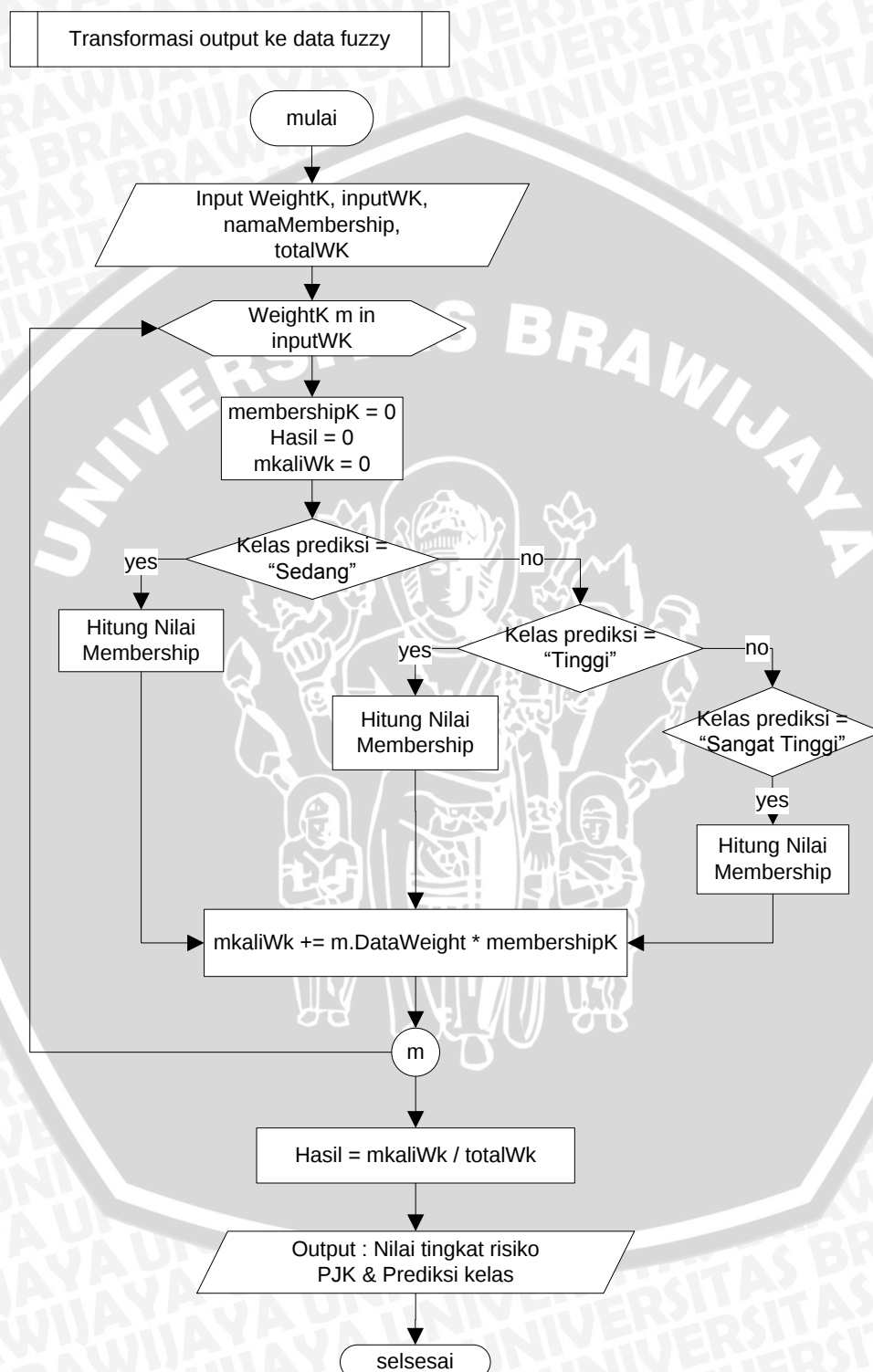
Gambar 3. 11 Alur Proses Transformasi Ke Data Fuzzy

Alur proses transformasi ke data *fuzzy* ketiga, yaitu untuk *membership* dengan 3 kelas (rendah, sedang, tinggi) dapat dilihat pada gambar 3.12.



Gambar 3. 12 Alur Proses Transformasi Ke Data Fuzzy

Alur proses transformasi ke data fuzzy keempat, yaitu untuk *membership* dengan 3 kelas (sedang, tinggi, sangat tinggi) dapat dilihat pada gambar 3.13.



Gambar 3. 13 Alur Proses Transformasi Ke Data Fuzzy

Atribut pada Tingkat risiko penyakit jantung koroner akan diubah ke dalam atribut *fuzzy*. Fungsi keanggotaan yang digunakan untuk mengubah atribut diperoleh dari jurnal penelitian Wahyuni (2011). Atribut tingkat risiko penyakit jantung koroner terdiri dari sangat rendah, rendah, sedang, tinggi dan sangat tinggi.

3.4. Perhitungan Manual

Pada subbab 3.4 akan menampilkan contoh perhitungan manual untuk proses prediksi tingkatan risiko bagi penderita Penyakit Jantung Koroner dengan sistem *Fuzzy K-Nearest Neighbor* (FK-NN) menggunakan 15 data pelatihan yang tercantum dalam tabel 3.1 dan data uji tercantum dalam tabel 3.2.

Tabel 3. 6 Data input pelatihan sistem

No.	Input						Target Output	
	Umur	LDL	Kol	HDL	Trigliserida	Sistolik		
1.	45	123	266	54	147	200	23	S
2.	67	167	255	39	220	140	31	T
3.	55	119	167	35	96	140	08	SR
4.	60	136	193	37	140	180	19	R
5.	61	98	256	33	269	150	29	S
6.	67	143	221	25	267	130	26	S
7.	57	115	218	66	274	150	20	S
8.	77	148	208	33	134	180	22	S
9.	67	94	182	75	149	160	19	R
10.	54	284	113	21	170	120	35	T
11.	53	160	200	39	160	150	32	T
12.	58	105	170	34	223	140	19	R
13.	77	186	168	66	177	130	22	S
14.	60	116	170	40	129	180	15	R
15.	65	137	214	13	302	140	25	S

Dari 15 data tersebut, memiliki 5 buah kelas sesuai dengan nilai target output yang mana merupakan tingkat risiko Penyakit Jantung Koroner. Dimana ke-5 kelas tersebut adalah tingkat risiko Sangat Rendah (SR), Rendah (R), Sedang (S), Tinggi (T) dan Sangat Tinggi (ST). Untuk *range* tingkat risiko dapat dilihat pada tabel 2.6.

Tabel 3. 7 Data input uji sistem

No.	Input						Target Output
	Umur	LDL	Kol	HDL	Trigliserida	Sistolik	
1.	51	196	200	72	159	140	?

Langkah-langkah perhitungan untuk algoritma *K-Nearest Neighbor Classification* adalah sebagai berikut:

- a. Langkah 1, setiap data latih d urutkan sesuai dengan target kelasnya, untuk mencari bobot dari setiap atributnya. Di buat tabel yang berisi data-data yang akan diklasifikasikan, seperti pada tabel 3.3.

Tabel 3. 8 Data yang akan dilakukan klasifikasi

Data Ke-	Umur	LDL	Kol	HDL	Trigli	Sis	Target
3.	55	119	167	35	96	140	08 SR
4.	60	136	193	37	140	180	19 R
9.	67	94	182	75	149	160	19 R
12.	58	105	170	34	223	140	19 R
14.	60	116	170	40	129	180	15 R
1.	45	123	266	54	147	200	23 S
5.	61	98	256	33	269	150	29 S
6.	67	143	221	25	267	130	26 S
7.	57	115	218	66	274	150	20 S
8.	77	148	208	33	134	180	22 S
13.	77	186	168	66	177	130	22 S
15.	65	137	214	13	302	140	25 S
2.	67	167	255	39	220	140	31 T
10.	54	284	113	21	170	120	35 T
11.	53	160	200	39	160	150	32 T

- b. Kemudian dilakukan normalisasi untuk setiap atribut, dimana untuk menghitung hasil normalisasi digunakan persamaan 2.4. Nilai *range* didapat dari nilai max data *training* dikurangi nilai min data *training* pada tabel 3.4. Dimana data *training* terdiri dari 15 *record*. Berikut ini adalah contoh, perhitungan normalisasi nilai *record* pertama untuk atribut umur.

$V = 45$ (nilai *record* pertama untuk atribut umur)

$\min [j] = 45$

$\max [j] = 77$

Setelah diketahui nilai *maksimum* dan *minimum*, kemudian dilakukan perhitungan menggunakan persamaan (3-1).

$$V' = \frac{(V - \min [j])}{\text{range } [j]}$$

$$= \frac{(45 - 45)}{(77 - 32)}$$

$$= 0$$

Dari perhitungan ini diperoleh nilai normalisasi untuk *record* pertama pada atribut umur, yaitu 0. Proses yang sama juga dilakukan untuk nilai *record* pada seluruh atribut. Hasil perhitungan normalisasi ditampilkan pada Tabel 3.5.

Tabel 3. 9 Range Data Training

	Umur	LDL	Kol	HDL	Trigli	Sis
Max	77	284	266	75	302	200
Min	45	94	113	13	96	120
Range	32	190	153	62	206	80

Tabel 3. 10 Data Pelatihan yang telah di normalisasi

No	Umur	LDL	Kol	HDL	Trigli	Sis
1.	0	0.152632	1	0.66129	0.247573	1
2.	0.6875	0.384211	0.928105	0.419355	0.601942	0.25
3.	0.3125	0.131579	0.352941	0.354839	0	0.25
4.	0.46875	0.221053	0.522876	0.387097	0.213592	0.75
5.	0.5	0.021053	0.934641	0.322581	0.839806	0.375
6.	0.6875	0.257895	0.705882	0.193548	0.830097	0.125
7.	0.375	0.110526	0.686275	0.854839	0.864078	0.375
8.	1	0.284211	0.620915	0.322581	0.184466	0.75
9.	0.6875	0	0.45098	1	0.257282	0.5
10.	0.28125	1	0	0.129032	0.359223	0
11.	0.25	0.347368	0.568627	0.419355	0.31068	0.375
12.	0.40625	0.057895	0.372549	0.33871	0.616505	0.25
13.	1	0.484211	0.359477	0.854839	0.393204	0.125
14.	0.46875	0.115789	0.372549	0.435484	0.160194	0.75
15.	0.625	0.226316	0.660131	0	1	0.25

Tabel 3. 11 Bobot setiap atribut data latih

Target	Umur	LDL	Kol	HDL	Trigli	Sis
Sangat Rendah	0.3125	0.131579	0.352941	0.354839	0	0.25
Rendah	0.507813	0.098684	0.429739	0.540323	0.311893	0.5625
Sedang	0.598214	0.219549	0.709617	0.458525	0.622746	0.428571
Tinggi	0.40625	0.577193	0.498911	0.322581	0.423948	0.208333
Sangat Tinggi						

Tabel 3. 12 Data Uji yang telah di normalisasi

Data Ke-	Umur	LDL	Kol	HDL	Trigli	Sis
1.	0.1875	0.536842	0.568627	0.951613	0.305825	0.25

- c. Kemudian, di cari terlebih dahulu jarak terdekat “*Euclidean distance*” dari data uji dengan data pelatihan. Dimana rumus untuk *Euclidean distance*, dapat dilihat pada persamaan (2-5). Berikut ini adalah contoh, perhitungan nilai jarak terdekat dari *record* pertama data latih dengan record uji.

$$d(x_1, x_2) = \sqrt{\sum_{r=1}^n w(a_r(x_1) - a_r(x_2))^2}$$

$$d = \sqrt{\frac{0.598214(0.1875 - 0)^2 + 0.219549(0.536842 - 0.152632)^2 + 0.709617(0.568627 - 1)^2 + 0.458525(0.951613 - 0.66129)^2 + 0.622746(0.305825 - 0.247573)^2 + 0.428571(0.25 - 1)^2}{0.598214 + 0.219549 + 0.709617 + 0.458525 + 0.622746 + 0.428571}}$$

$$= \sqrt{\frac{0.021031 + 0.032409 + 0.132047 + 0.038648 + 0.002113 + 0.241071}{0.598214 + 0.219549 + 0.709617 + 0.458525 + 0.622746 + 0.428571}}$$

$$d = 0.22507$$

- d. Selanjutnya dihitung lagi jarak antara *record* uji dan bobot dengan *record* data latih yang lain. Untuk hasil perhitungan nilai dari jarak terdekat dapat dilihat pada tabel 3.7.

Tabel 3. 13 Nilai jarak terdekat data uji dengan data pelatihan

Data Ke-	Nilai	Target
1.	0.22507	S
2.	0.22772	T
3.	0.29349	SR
4.	0.24696	R
5.	0.25022	S
6.	0.788269	S
7.	0.25953	S
8.	0.17291	S

9.	0.27715	R
10.	0.29621	T
11.	0.14032	T
12.	0.22217	R
13.	0.21897	S
14.	0.24662	R
15.	0.30479	S

- e. Selanjutnya dari data yang telah ditemukan nilai *Euclidean distance* dilakukan pencarian nilai *weight* (w) dengan menggunakan persamaan 2.7. Berikut ini adalah contoh, perhitungan nilai *weight* dari hasil *record* pertama perhitungan *Euclidean distance*.

$$\text{Weight} = \frac{1}{0.22507}$$

$$= 19.74$$

- f. Untuk hasil perhitungan nilai *weight* dari hasil *Euclidean Distance* untuk seluruh *record* dapat dilihat pada tabel 3.8.

Tabel 3. 14 Nilai weighted

Data Ke-	Nilai	Target
1.	19.74	S
2.	19.28	T
3.	11.61	SR
4.	16.37	R
5.	15.97	S
6.	14.85	S
7.	33.44	S
8.	13.02	S
9.	30.26	R
10.	11.39	T
11.	50.79	T
12.	20.26	R
13.	20.86	S
14.	16.44	R
15.	10.76	S

- g. Misal, pada saat awal di inputkan nilai $k=4$, maka pada data setelah mengalami proses *weighted* data tersebut di urutkan sesuai dengan nilai yang paling besar. Jika nilai di ambil dari *Euclidean distance* yang di ambil nilai yang terkecil, karena *weighted* merupakan nilai *invers* dari *Euclidean distance*, maka yang di ambil adalah nilai terbesar. Kemudian $k=4$ dimaksudkan, di ambil 4 nilai data yang terbesar. Data hasil *sorting* dilihat pada tabel 3.9 dan data setelah penentuan $k=4$ dapat dilihat pada tabel 3.10.

Tabel 3. 15 Data setelah sorting dari yang terbesar

Data Ke-	Nilai	Urutan terbesar	Target	
1.	19.74	6	S	23
2.	19.28	7	T	31
3.	11.61	13	SR	8
4.	16.37	9	R	19
5.	15.97	10	S	29
6.	14.85	11	S	26
7.	33.44	2	S	20
8.	13.02	12	S	22
9.	30.26	3	R	19
10.	11.39	14	T	35
11.	50.79	1	T	32
12.	20.26	5	R	19
13.	20.86	4	S	22
14.	16.44	8	R	15
15.	10.76	15	S	25

Tabel 3. 16 Seleksi 4 data dengan nilai terbesar

Data Ke-	Nilai	Urutan terbesar	Target	
11	50.79	1	T	32
7	33.44	2	S	20
9	30.26	3	R	19
13	20.86	4	S	22

Didapatkan 4 nilai data terbesar, dimana masing-masing data memiliki 2 nilai target S (Sedang), 1 nilai target T (Tinggi), 1 nilai target R (Rendah).

- h. Kemudian, dilakukan perhitungan *membership function* sesuai dengan target kondisi awal data latih. Misal untuk data ke-11 dengan nilai target 32. Jadi $x=32$, dan akan di masukkan ke dalam semua *linguistic*. Berikut ini adalah contoh, perhitungan nilai $x=32$ dalam *membership*.

Tabel 3. 17 Target dalam linguistic membershipnya

Membership	Range	Perhitungan	Hasil
$\mu_{\text{SANGAT RENDAH}}(x)$	$0 \leq x < 5$	1	-
	$5 \leq x < 15$	$\frac{(5-x)}{10}$	-
	$x \geq 15$	0	0
$\mu_{\text{RENDAH}}(x)$	$x \leq 5$ atau $x \geq 25$	0	-
	$5 \leq x < 15$	$\frac{(x-5)}{10}$	-
	$15 \leq x < 25$	$\frac{(25-23)}{10}$	0
$\mu_{\text{SEDANG}}(x)$	$x \leq 15$ atau $x \geq 35$	0	0
	$15 \leq x < 25$	$\frac{(x-15)}{10}$	0.8
	$25 \leq x < 35$	$\frac{(35-x)}{10}$	0.3
$\mu_{\text{TINGGI}}(x)$	$x \leq 25$ atau $x \geq 45$	0	0
	$25 \leq x < 35$	$\frac{(x-25)}{10}$	0.7
	$35 \leq x < 45$	$\frac{(45-x)}{10}$	-
$\mu_{\text{SANGAT TINGGI}}(x)$	$x \leq 35$	0	0
	$35 \leq x < 45$	$\frac{(x-35)}{10}$	-
	$x \geq 45$	1	-

Tabel 3. 18 Nilai Fuzzy untuk setiap target

Data ke-	data		target		SR	R	S	T	ST
11	50.79		T	32	0	0	0.3	0.7	0
7	33.44		S	20	0	0.5	0.5	0	0
9	30.26		R	19	0	0.6	0.4	0	0
13	20.86		S	22	0	0.3	0.7	0	0
total	135.35								

$$S = \frac{(0.3 * 50.79) + (0.5 * 33.44) + (0.4 * 30.26) + (0.7 * 20.86)}{135.35}$$

$$= 0.433417$$

Tabel 3. 19 Nilai *membership* akhir

SR	0
R	0.303929
S	0.433417
T	0.262657
ST	0

- i. Dari *membership* setiap *linguistic* tersebut akan dicari nilai yang paling besar. Dari perhitungan tersebut, didapat nilai terbesar pada *linguistic* S (sedang) dengan nilai 0.433417.
- j. Jadi, jika masukan tersebut dicari dengan nilai $k=4$ akan didapat hasil dari data uji dengan target S (sedang) dengan nilai FK-NN 0.433417.
- k. Jika di cari nilai data uji dengan $k=1$. Maka akan dicari 1 nilai terbesar dari *weighted*.

Tabel 3. 20 Seleksi 1 data dengan nilai terbesar

Data Ke-	Nilai	Target
11	50.79	T 32

Didapatkan 1 nilai data terbesar, pada target T (Tinggi)

- l. Kemudian, dilakukan perhitungan *membership function* sesuai dengan target kondisi awal data latih. Data ke-11 tersebut memiliki nilai target 32. Jadi $x=32$, dan akan di masukkan ke dalam semua *linguistic*. Berikut ini adalah contoh, perhitungan nilai $x=32$ dalam *membership*.

Tabel 3. 21 Target dalam linguistic membership

Membership	Range	Perhitungan	Hasil
$\mu_{\text{SANGATRENDAH}}(x)$	$0 \leq x < 5$	1	-
	$5 \leq x < 15$	$\frac{(5-x)}{10}$	-
	$x \geq 15$	0	0
$\mu_{\text{RENDAH}}(x)$	$x \leq 5$ atau $x \geq 25$	0	0
	$5 \leq x < 15$	$\frac{(x-5)}{10}$	-
	$15 \leq x < 25$	$\frac{(25-23)}{10}$	-
$\mu_{\text{SEDANG}}(x)$	$x \leq 15$ atau $x \geq 35$	0	-
	$15 \leq x < 25$	$\frac{(x-15)}{10}$	-
	$25 \leq x < 35$	$\frac{(35-x)}{10}$	0.3
$\mu_{\text{TINGGI}}(x)$	$x \leq 25$ atau $x \geq 45$	0	-
	$25 \leq x < 35$	$\frac{(x-25)}{10}$	0.7
	$35 \leq x < 45$	$\frac{(45-x)}{10}$	-
$\mu_{\text{SANGATTINGGI}}(x)$	$x \leq 35$	0	0
	$35 \leq x < 45$	$\frac{(x-35)}{10}$	-
	$x \geq 45$	1	-

Tabel 3. 22 Nilai Fuzzy untuk setiap target

Data ke-	data		target		SR	R	S	T	ST
11	50.79		T	32	0	0	0.3	0.7	0
total	50.79								

- m. Dari perhitungan tersebut akan dihitung nilai *membership* dari setiap *linguistic*-nya. Berikut ini adalah contoh, perhitungan nilai setiap *linguistic* dengan data uji-nya.

$$S = \frac{(50.79 * 0.3)}{0.3}$$

$$S = 50.79$$

Tabel 3. 23 Nilai *membership* akhir

SR		0
R		0
S		0.3
T		0.7
ST		0

- n. Dari *membership* setiap *linguistic* tersebut akan dicari nilai yang paling besar. Dari perhitungan tersebut, didapat nilai terbesar pada *linguistic* T (tinggi) dengan nilai 0.7.
- o. Jadi, jika masukan tersebut dicari dengan nilai $k=1$ akan didapat hasil dari data uji dengan target T (tinggi) dengan nilai FK-NN 0.7.

3.5. Perancangan Antarmuka

Antarmuka (*interface*) untuk sistem, terdiri dari 2 form. Dimana form untuk prediksi dan form untuk pengujian. Form prediksi ditunjukkan pada gambar 3.14 dan form pengujian ditunjukkan pada gambar 3.15.

Gambar 3. 14 Antarmuka sistem prediksi

Pada gambar 3.14, antarmuka sistem terdiri dari:

1. *Textbox* yang digunakan untuk menginputkan data latih, dari file berupa .txt.
2. Tombol *browse* digunakan untuk mencari data yang akan diinputkan ke dalam sistem.
3. Tombol input data digunakan untuk memasukkan data dan menampilkan data ke dalam *datagridview*.
4. *Textbox* yang digunakan untuk menginputkan data uji yang meliputi 6 atribut.
5. *Combobox* untuk memilih inputan nilai k.
6. Tombol proses digunakan untuk melakukan proses prediksi.
7. Tab FK-NN untuk menampilkan hasil akhir prediksi FK-NN dalam bentuk *datagridview*, dengan menampilkan nama *membership* dan nilai *memberhip*.
8. *LabelText* yang digunakan untuk menampilkan hasil prediksi yang berupa kategori.
9. *LabelText* yang digunakan untuk menampilkan hasil prediksi yang berupa nilai *membership*.

The screenshot shows a software interface for testing the FK-NN model. It has several input sections and a results display area.

- Data Uji (Testing Data):** Includes an 'Input Data Uji' field with a value of 1, a '2' button, and an 'Input Data' button with a value of 3.
- Data Latih (Training Data):** Includes an 'Input Data Latih' field with a value of 4, a '5' button, and an 'Input Data' button with a value of 6.
- Input Nilai k:** A dropdown menu showing 'Input k: 1' and a value of 7.
- Akurasi (Accuracy):** A section with radio buttons for 'SR-R-S-T-ST', 'SR-R-S', 'R-S-T', and 'S-T-ST'. The 'R-S-T' option is selected, indicated by a green '8'.
- Keterangan (Description):** A list of membership levels: 'SR : Sangat Rendah', 'R : Rendah', 'S : Sedang', 'T : Tinggi', and 'ST : Sangat Tinggi'. The 'S : Sedang' option is selected, indicated by a green '9'.
- Proses:** A button labeled 'Proses' with a green '10'.
- Results:** A large area displaying the number '11' in green. Below it, there are labels for 'Akurasi : 12 %', 'Benar : 13', 'Salah : 14', and a 'Save to Excel' button with a green '15'.

Gambar 3. 15 Antarmuka sistem pengujian

Pada gambar 3.15, antarmuka sistem terdiri dari:

1. *Textbox* yang digunakan untuk menginputkan data uji, dari file berupa .txt.
2. Tombol *browse* digunakan untuk mencari data uji yang akan diinputkan ke dalam sistem.
3. Tombol input data digunakan untuk memasukkan data uji dan menampilkan data ke dalam *datagridview*.
4. *Textbox* yang digunakan untuk menginputkan data latih, dari file berupa .txt.
5. Tombol *browse* digunakan untuk mencari data latih yang akan diinputkan ke dalam sistem.
6. Tombol input data digunakan untuk memasukkan data latih dan menampilkan data ke dalam *datagridview*.
7. *Combobox* untuk memilih inputan nilai k.
8. *Radiobutton* digunakan untuk memasukkan akurasi data.
9. *Textbox* untuk member keterangan dari *radiobutton* pada *groupbox* akurasi.
10. Tombol proses digunakan untuk melakukan proses prediksi.
11. *Datagridview* untuk menampilkan hasil system dan hasil sebenarnya.
12. *Textbox* untuk menampilkan nilai akurasi sistem.
13. *Textbox* untuk menampilkan berapa data uji yang memiliki hasil prediksi Benar.
14. *Textbox* untuk menampilkan berapa data uji yang memiliki hasil prediksi Salah.
15. Tombol *Save to Excel* untuk menyimpan hasil prediksi ke dalam data excel.

3.6. Perancangan Uji Coba

Setelah sistem selesai dibuat, langkah selanjutnya adalah melakukan pengujian terhadap sistem tersebut. Pengujian dilakukan untuk mengetahui tingkat akurasi dari hasil prediksi tingkat risiko Penyakit Jantung Koroner (PJK) menggunakan metode *Fuzzy K-Nearest Neighbor*. Pengujian tersebut dilakukan terhadap data Penyakit Jantung Koroner yang terdiri dari 5 tingkat risiko, yaitu tingkat risiko Sangat rendah, rendah, normal, tinggi dan sangat tinggi.

Terdapat 2 macam pengujian yang dilakukan dalam penelitian ini, yaitu:

1. Pengujian untuk mengetahui pengaruh nilai k (jumlah tetangga terdekat) dan jumlah data latih terhadap tingkat akurasi sistem.
2. Pengujian untuk mengetahui pengaruh jumlah data kelas mayoritas dan data kelas minoritas terhadap tingkat akurasi sistem.

2.6.1. Uji Pengaruh Nilai k dan Data Latih Terhadap Tingkat Akurasi

Pengujian dan analisis hasil prediksi Penyakit Jantung Koroner (PJK) dilakukan melalui perbandingan kesalahan antara prediksi PJK sistem dengan PJK pakar. Nilai akurasi sistem dapat diketahui melalui perhitungan akurasi dengan menggunakan persamaan (2-10) dan (2-11).

Tabel 3. 24 Tabel Uji Pengaruh Nilai k dan Data Latih Terhadap Akurasi Sistem

k	Prediksi Benar	Prediksi Salah	Akurasi (%)
k = 1			
k = 2			
k = 3			
k = 4			
k = 5			
k = 6			
k = 7			
k = 8			
k = 9			
RATA-RATA			

Keterangan:

1. k = nilai k yang akan diuji
2. Akurasi = tingkat akurasi dalam persen, yang dihitung dengan persamaan 2.12.
3. Akurasi Rata-rata = tingkat akurasi rata-rata dalam persen dari beberapa kali pengujian dengan nilai k yang sama.

Uji pengaruh jumlah data latih terhadap tingkat akurasi klasifikasi dilakukan pada dataset penyakit jantung koroner. Pada pengujian ini dimasukkan 4 jumlah data latih yang berbeda, yaitu 40, 50, 60 dan 70 data latih.

Tabel 3. 25 Tabel Uji Pengaruh Jumlah Data Latih Terhadap Akurasi

Data Latih	k	Akurasi (%)
40 Data Latih	k = 1	
	k = 2	
	k = 3	
	k = 4	
	k = 5	
	k = 6	
	k = 7	
	k = 8	
	k = 9	
50 Data Latih	k = 1	
	k = 2	
	k = 3	
	k = 4	
	k = 5	
	k = 6	
	k = 7	
	k = 8	
	k = 9	
60 Data Latih	k = 1	
	k = 2	
	k = 3	
	k = 4	
	k = 5	
	k = 6	
	k = 7	
	k = 8	
	k = 9	
70 Data Latih	k = 1	
	k = 2	
	k = 3	
	k = 4	
	k = 5	
	k = 6	
	k = 7	
	k = 8	
	k = 9	

Keterangan:

1. Jumlah Data Latih = jumlah data yang digunakan sebagai data latih
2. k = kolom untuk nilai k yang dimasukkan
3. Kolom nilai akurasi menunjukkan hasil akurasi dari 5 pengujian yang telah dilakukan. Perhitungan tingkat keakurasian ini didapatkan pada persamaan 2.11.

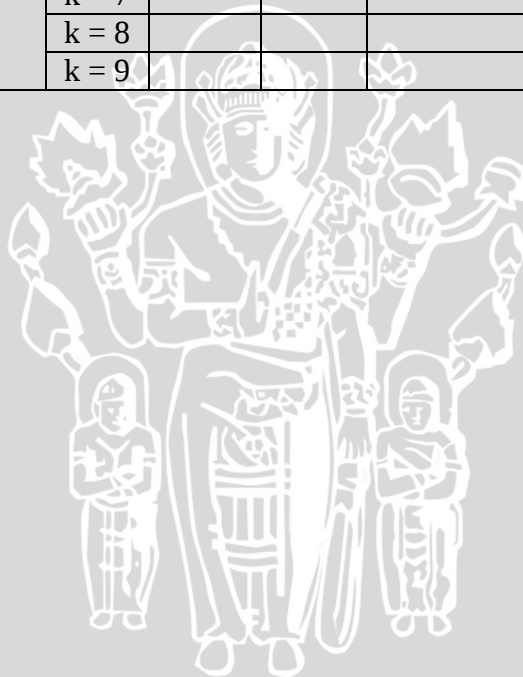
3.6.3. Uji Pengaruh Kelas mayoritas dan kelas minoritas

Uji pengaruh kelas mayoritas dan minoritas dimaksudkan untuk mengetahui pengaruh data latih yang terlalu dominan atau tidak. Pada pengujian ini, akan dibagi menjadi 4 kategori data latih. Dimana kategori pertama yaitu untuk data latih yang memiliki kelas Sangat Rendah – Rendah – Sedang. Untuk kategori kedua untuk data latih yang memiliki kelas Rendah – Sedang – Tinggi. Untuk kategori ketiga yang memiliki kelas Sedang – Tinggi – Sangat Tinggi. Dan untuk kategori terakhir memiliki semua kelas yaitu Sangat Rendah – Rendah - Sedang – Tinggi – Sangat Tinggi.

Tabel 3. 26 Tabel Uji Pengaruh Kelas Mayoritas dan Kelas Minoritas

Data Latih	k	Benar	Salah	Akurasi (%)
Sangat Rendah, Rendah, Sedang	k = 1			
	k = 2			
	k = 3			
	k = 4			
	k = 5			
	k = 6			
	k = 7			
	k = 8			
	k = 9			
Rendah, Sedang, Tinggi	k = 1			
	k = 2			
	k = 3			
	k = 4			
	k = 5			
	k = 6			
	k = 7			
	k = 8			
	k = 9			

Sedang, Tinggi, Sangat Tinggi	k = 1			
	k = 2			
	k = 3			
	k = 4			
	k = 5			
	k = 6			
	k = 7			
	k = 8			
	k = 9			
Sangat Rendah, Rendah, Sedang, Tinggi, Sangat Tinggi	k = 1			
	k = 2			
	k = 3			
	k = 4			
	k = 5			
	k = 6			
	k = 7			
	k = 8			
	k = 9			



BAB IV

IMPLEMENTASI DAN PEMBAHASAN

5.1. Lingkungan Implementasi

Implementasi perangkat lunak ini berupa aplikasi pemrograman yang menerapkan metode *Fuzzy K-Nearest Neighbor* (FK-NN) dalam memprediksi tingkat risiko penyakit jantung koroner. Terdapat beberapa atribut dalam data tingkat risiko penyakit jantung koroner, diantaranya umur, LDL, kolesterol, HDL, trigliserida dan sistolik. Setiap atribut ini memiliki nilai *linguistic* yang berbeda-beda. Adapun lingkungan implementasi akan dijelaskan ke dalam subbab lingkungan implementasi perangkat keras dan perangkat lunak.

5.1.1. Lingkungan implementasi perangkat keras

Perangkat keras yang digunakan untuk melaksanakan penelitian tentang prediksi tingkat risiko Penyakit Jantung Koroner (PJK) menggunakan *Fuzzy K-Nearest Neighbor* (FK-NN) adalah:

1. *Prosesor Intel Atom™ Processor N450 (1.66 GHz, 512 KB Cache)*
2. *Memori 1 GB*
3. *Harddisk 250 GB*
4. *Monitor*
5. *Keyboard*
6. *Mouse*

5.1.2. Lingkungan implementasi perangkat lunak

Perangkat lunak yang digunakan untuk melaksanakan penelitian tentang prediksi tingkat risiko Penyakit Jantung Koroner (PJK) menggunakan *Fuzzy K-Nearest Neighbor* (FK-NN) adalah:

1. Sistem operasi yang digunakan yaitu Windows 7 Ultimate Home
2. Microsoft Visual C# 2008 *Express Edition* sebagai *software Development* dalam implementasi perancangan sistem
3. Notepad dan *Microsoft office excel* 2007 sebagai media penyimpanan data.

5.2. Implementasi Program

Berdasarkan analisa dan perancangan proses yang telah dipaparkan pada Bab III, maka pada bab ini akan dijelaskan proses-proses implementasinya.

5.2.1. Struktur Data

Struktur data digunakan untuk menyimpan data-data yang dibutuhkan oleh sistem. Pada sistem ini digunakan struktur data berupa kelas-kelas yang digunakan untuk menyimpan variabel-variabel yang digunakan dalam sistem FK-NN.

Kelas *FormUtama* digunakan untuk menyimpan data yang akan digunakan pada proses FK-NN. Dimana terdapat *list* dataUji, normalUji, dataLatih dan normalLatih yang bertipe *InputData*. Kelas *FormUtama* akan direpresentasikan pada *sourcecode* 4.1.

```

1. public partial class FormUtama : KryptonForm
2. {
3.
4.     List<InputData> dataUji;
5.     List<InputData> normalUji;
6.
7.     List<InputData> dataLatih;
8.     List<InputData> normalLatih;

```

Sourcecode 4. 1 Kelas FormUtama

Kelas *Inputdata* merupakan kelas yang digunakan menyimpan data *input* dari data uji dan data latih yang terdiri dari beberapa atribut. Kelas *Inputdata* diimplementasikan pada *sourcecode* 4.2.

```

1. class InputData
2. {
3.     private double umur;
4.     private double ldl;
5.     private double kolesterol;
6.     private double hdl;
7.     private double trigliserida;
8.     private double sistolik;
9.     private double nilai;
10.    private string kategori;
11.    private string namaData;
12.    private string nilaiK;
13.
14.    public string NamaData{
15.        get{return namaData;}
16.        set{namaData = value;}}
17.
18.    public string NilaiK{
19.        get{return nilaiK;}

```

```
20.         set{nilaiK = value;}}
21.
22.     public double Umur{
23.         get{return umur;}
24.         set{umur = value;}}
25.
26.     public double LDL{
27.         get{return ldl;}
28.         set{ldl = value;}}
29.
30.     public double Kolesterol{
31.         get{return kolesterol;}
32.         set{kolesterol = value;}}
33.
34.     public double HDL{
35.         get{return hdl;}
36.         set{hdl = value;}}
37.
38.     public double Trigliserida{
39.         get{return trigliserida;}
40.         set{trigliserida = value;}}
41.
42.     public double Sistolik{
43.         get{return sistolik;}
44.         set{sistolik = value;}}
45.
46.     public double Nilai{
47.         get{return nilai;}
48.         set{nilai = value;}}
49.
50.     public string Kategori{
51.         get{return kategori;}
52.         set{kategori = value;}}
53.     }
54.
55.     public double SangatRendah
56.     {
57.         get{
58.             double hasilSR = 0;
59.             if ((nilai >= 0) && (nilai < 5)){
60.                 hasilSR = 1;}
61.             else if ((nilai >= 5) && (nilai < 15)){
62.                 hasilSR = (nilai - 5.0) / 10.0;}
63.             else if (nilai >= 15){
64.                 hasilSR = 0;}
65.             return hasilSR;}
66.     }
67.
68.     public double Rendah{
69.         get{
70.             double hasilR = 0;
71.             if ((nilai >= 15) && (nilai < 25)){
72.                 hasilR = (25 - nilai) / 10.0;}
73.             else if ((nilai >= 5) && (nilai < 15)){
74.                 hasilR = (nilai - 5.0) / 10.0;}
75.             else if ((nilai <= 5) || (nilai >= 25)){
76.                 hasilR = 0;}
77.             return hasilR;}
78.     }
79.
80.     public double Sedang{
81.         get{
82.             double hasilS = 0;
83.             if ((nilai >= 25) && (nilai <= 35)){
```

```

84.         hasilS = (35 - nilai) / 10.0;}
85.     else if ((nilai >= 15) && (nilai < 25)){
86.         hasilS = (nilai - 15.0) / 10.0;}
87.     else if ((nilai <= 15)|| (nilai >= 35)){
88.         hasilS = 0;}
89.     return hasilS;}
90. }
91.
92. public double Tinggi{
93.     get{
94.         double hasilT = 0;
95.         if ((nilai >= 35) && (nilai < 45)){
96.             hasilT = (45 - nilai) / 10.0;}
97.         else if ((nilai >= 25) && (nilai < 35)){
98.             hasilT = (nilai - 25.0) / 10.0;}
99.         else if ((nilai <= 25)|| (nilai >= 45)){
100.             hasilT = 0;}
101.         return hasilT;}
102.     }
103.
104. public double SangatTinggi{
105.     get{
106.         double hasilST = 0;
107.         if (nilai >= 45){
108.             hasilST = 1;}
109.         else if ((nilai >= 35) && (nilai < 45)){
110.             hasilST = (nilai - 35) / 10.0;}
111.         else if (nilai <= 35){
112.             hasilST = 0;}
113.         return hasilST;}
114.     }
115. }

```

Sourcecode 4. 2 Kelas InputData

Kelas *Euclidean* digunakan untuk menyimpan data dari perhitungan jarak terdekat antara data latih dengan data uji. Penggunaan kelas *Euclidean* diimplementasikan pada sourcecode 4.3.

```

1. class Euclidean
2. {
3.     private string namaData;
4.     private double dataEucli;
5.
6.     public string>NamaData{
7.         get{return namaData;}
8.         set{namaData = value;}
9.     }
10.
11.     public double>DataEucli{
12.         get{return dataEucli;}
13.         set{dataEucli = value;}
14.     }
15.
16.     public double>Nilai{
17.         get;
18.         set;
19.     }
20.
21.     public string>Kategori{
22.         get;

```

```

23.         set;
24.     }
25.
26.     public double SangatRendah{
27.         get;
28.         set;
29.     }
30.
31.     public double Rendah{
32.         get;
33.         set;
34.     }
35.
36.     public double Sedang{
37.         get;
38.         set;
39.     }
40.
41.     public double Tinggi{
42.         get;
43.         set;
44.     }
45.
46.     public double SangatTinggi{
47.         get;
48.         set;
49.     }
50. }

```

Sourcecode 4. 3 Kelas Euclidean

Kelas *Weight* digunakan dalam sistem untuk menyimpan data hasil *weight*, pada *form* pelatihan maupun *form* pengujian. Kelas *weight* diimplementasikan pada *sourcecode* 4.4.

```

1.  class Weight
2.  {
3.      private string namaData;
4.      private double dataWeight;
5.
6.      public string>NamaData{
7.          get{return namaData;}
8.          set{namaData = value;}
9.      }
10.     public double>DataWeight{
11.         get{return dataWeight;}
12.         set{dataWeight = value;}
13.     }
14.     public double>Nilai{
15.         get;
16.         set;}
17.     public string>Kategori{
18.         get;
19.         set;}
20.     public double>SangatRendah{
21.         get;
22.         set;}
23.     public double>Rendah{
24.         get;
25.         set;}
26.     public double>Sedang{

```

```

27.         get;
28.         set;}
29.     public double Tinggi{
30.         get;
31.         set;}
32.     public double SangatTinggi{
33.         get;
34.         set;}
35.
36.     public override string ToString(){
37.         return String.Format("{0} ({1})", namaData,
38.                                dataWeight);
39.     }
40. }

```

Sourcecode 4. 4 Kelas Weight

Pada kelas *membership*, akan disimpan hasil akhir data sebelum masuk dalam proses akurasi. Dimana data pada kelas *membership*, merupakan data hasil perhitungan *weight* dengan sejumlah k yang dimasukkan kedalam setiap *linguistic* variabel *membership*. Implentasi kelas Membership ditunjukkan pada sourcecode 4.5.

```

1.     class Membership{
2.         private string namaData;
3.
4.         public string NamaData{
5.             get { return namaData; }
6.             set { namaData = value; }
7.         }
8.
9.         public String NamaMembership{
10.            get;
11.            set;}
12.
13.         public Double NilaiMembership{
14.             get;
15.             set;}
16.
17.         public override string ToString(){
18.             return String.Format("{0} ({1})", namaData,
19.                                    NamaMembership);
20.     }

```

Sourcecode 4. 5 Kelas Membership

5.2.2. Implementasi Pelatihan

5.2.2.1. Implementasi Baca File Data Latih dan data Uji.

Pada tahap awal sistem, proses pertama yang dilakukan adalah proses mengambil data latih dan data uji dengan membaca data dengan format .txt. Setelah itu, data yang telah dimasukkan tersebut disimpan dalam kelas *Inputdata* yaitu *dataPerLine*, dimana pada kelas tersebut berisi variabel-variabel untuk

penyimpanan setiap atribut penyakit jantung koroner. Pada proses pembacaan data, tiap data akan dibaca per baris dengan method `ReadLine()`. Adapun pembacaan data tiap attribute dengan memecah data berbaris berdasarkan karakter spasi dengan method `Split()`. Hasil pemecahan data per baris berupa array string dan disimpan pada variable `outputPerLine`. Selanjutnya data `outputPerLine` akan dijadikan inputan attribute dari `dataPerLine`. Tiap data uji selanjutnya disimpan pada sebuah list data bertipe input data pada variable `dataUji`. Implementasi baca data latih ditunjukkan pada *sourcecode* 4.6.

```

1. dataUji = new List<InputData>();
2.
3. if (File.Exists(dataUjiTextBox.Text)){
4.     int count = 0;
5.     StreamReader file = new
6.     StreamReader(dataUjiTextBox.Text);
7.     while ((file.ReadLine()) != null){
8.
9.         count++;
10.
11.         string[] outputPerLine = file.ReadLine().Split("
12.             ".ToCharArray());
13.
14.         InputData dataPerLine = new InputData();
15.         dataPerLine>NamaData = String.Format("Data Uji-{0}", count);
16.         dataPerLine.Umur = Convert.ToDouble(outputPerLine[0]);
17.         dataPerLine.LDL = Convert.ToDouble(outputPerLine[1]);
18.         dataPerLine.Kolesterol = Convert.ToDouble(outputPerLine[2]);
19.         dataPerLine.HDL = Convert.ToDouble(outputPerLine[3]);
20.         dataPerLine.Trigliserida = Convert.ToDouble(outputPerLine[4]);
21.         dataPerLine.Sistolik = Convert.ToDouble(outputPerLine[5]);
22.         dataPerLine.Nilai = Convert.ToDouble(outputPerLine[6]);
23.         dataPerLine.Kategori = (outputPerLine[7]);
24.         dataUji.Add(dataPerLine);

```

Sourcecode 4. 6 Baca File Data Latih

5.2.2.2. Implementasi Normalisasi

Dalam proses Normalisasi, awalnya dicari terlebih dahulu nilai *maksimum* dan nilai *minimum*. Untuk perhitungan Max, terdapat seleksi antara nilai data uji dengan data latih dan nantinya dipilih data yang mempunyai nilai yang paling tinggi dan disimpan pada variable `MaxData`. Variabel ini yang nantinya akan dijadikan nilai *return* dari method `Max` dalam mencari nilai maximum antara data latih dan data uji. Implementasi perhitungan max di tampilkan pada *sorcecode* 4.7.

```
1. private double Max (double maxLatih, double maxUji)
2. {
3.     double maxData = 0;
4.     if (maxUji > maxLatih){
5.         maxData = maxUji;}
6.     else{
7.         maxData = maxLatih;}
8.     return maxData;
9. }
```

Sourcecode 4. 7 Perhitungan Max

Perhitungan Min, terdapat seleksi antara nilai data uji dengan data latih untuk mendapatkan nilai yang paling kecil. Implementasi perhitungan min di ditampilkan pada *sourcecode* 4.8.

```
1. private double Min(double minLatih, double minUji)
2. {
3.     double minData = 0;
4.     if (minUji < minLatih){
5.         minData = minUji;}
6.     else{
7.         minData = minLatih;}
8.     return minData;
9. }
```

Sourcecode 4. 8 Perhitungan Min

Setelah didapatkan nilai *minimum* dan *maksimum* data, kemudian setiap data baru dapat di normalisasi. Proses normalisasi data dilakukan oleh method Normalisasi. Implementasi perhitungan normalisasi di ditampilkan pada *sourcecode* 4.9.

```
1. private double Normalisasi(double data, double DataMin, double
2. DataMax){
3.     double hasil = 0;
4.     hasil = (data - DataMin) / (DataMax - DataMin);
5.     return hasil;
6. }
```

Sourcecode 4. 9 Perhitungan Normalisasi

Untuk proses normalisasi setiap atributnya ditunjukkan pada *sourcecode* 4.10. Data normalisasi ini akan di simpan ke dalam *List*, untuk data latih disimpan ke dalam list *normalLatih* dan *normalUji*. Pada proses normalisasi data, untuk dapat mengambil data minimum akan dilakukan pengurutan data dengan fungsi *OrderBy()* dan data minimum merupakan data pertama dan dipilih dengan fungsi *First()*. Sedangkan untuk mencari nilai maksimum dengan menggunakan fungsi

OrderByDescending(). Selanjutnya tiap atribut data latih dan data uji akan dilakukan normalisasi data atribut dengan menggunakan fungsi normalisasi().

```

1. // Umur //
2. double minUUmur = ((InputData)dataUji.OrderBy(u =>
3.     u.Umur).First()).Umur;
4. double minLUmur = ((InputData)dataLatih.OrderBy(u =>
5.     u.Umur).First()).Umur;
6.
7. double maxUUmur = ((InputData)dataUji.OrderByDescending(u =>
8.     u.Umur).First()).Umur;
9.
10. double maxLUmur = ((InputData)dataLatih.OrderByDescending(u =>
11.     u.Umur).First()).Umur;
12.
13.
14. // LDL //
15. double minULDL = ((InputData)dataUji.OrderBy(l =>
16.     l.LDL).First()).LDL;
17. double minLLDL = ((InputData)dataLatih.OrderBy(l =>
18.     l.LDL).First()).LDL;
19. double maxULDL = ((InputData)dataUji.OrderByDescending(l =>
20.     l.LDL).First()).LDL;
21. double maxLLDL = ((InputData)dataLatih.OrderByDescending(l =>
22.     l.LDL).First()).LDL;
23.
24. // Kolesterol //
25. double minUKolesterol = ((InputData)dataUji.OrderBy(k =>
26.     k.Kolesterol).First()).Kolesterol;
27. double minLKolesterol = ((InputData)dataLatih.OrderBy(k =>
28.     k.Kolesterol).First()).Kolesterol;
29. double maxUKolesterol =
30.     ((InputData)dataUji.OrderByDescending(k
31. => k.Kolesterol).First()).Kolesterol;
32. double maxLKolesterol =
33.     ((InputData)dataLatih.OrderByDescending(
34. k => k.Kolesterol).First()).Kolesterol;
35.
36. // HDL //
37. double minUHdl = ((InputData)dataUji.OrderBy(h =>
38.     h.HDL).First()).HDL;
39. double minLHdl = ((InputData)dataLatih.OrderBy(h =>
40.     h.HDL).First()).HDL;
41.
42. double maxUHdl = ((InputData)dataUji.OrderByDescending(h =>
43.     h.HDL).First()).HDL;
44. double maxLHdl = ((InputData)dataLatih.OrderByDescending(h =>
45.     h.HDL).First()).HDL;
46.
47. // Triglicerida //
48. double minUTriglicerida = ((InputData)dataUji.OrderBy(t =>
49.     t.Triglicerida).First()).Triglicerida;
50. double minLTriglicerida = ((InputData)dataLatih.OrderBy(t =>
51.     t.Triglicerida).First()).Triglicerida;
52. double maxUTriglicerida =
53.     ((InputData)dataUji.OrderByDescending(t
54. =>
55. t.Triglicerida).First()).Triglicerida;
56. double maxLTriglicerida =
57.     ((InputData)dataLatih.OrderByDescending
58. (t =>
59. t.Triglicerida).First()).Triglicerida;
60.

```

```
61. // Sistolik //
62. double minUSistolik = ((InputData)dataUji.OrderBy(s =>
63.     s.Sistolik).First()).Sistolik;
64. double minLSistolik = ((InputData)dataLatih.OrderBy(s =>
65.     s.Sistolik).First()).Sistolik;
66. double maxUSistolik = ((InputData)dataUji.OrderByDescending(s
67.     => s.Sistolik).First()).Sistolik;
68. double maxLSistolik =
69.     ((InputData)dataLatih.OrderByDescending(s
70.     => s.Sistolik).First()).Sistolik;
71.
72. ///////////NORMALISASI DATA UJI//////////
73. normalUji = new List<InputData>();
74. foreach (var training in dataUji)
75. {
76.     InputData dataUNormal = new InputData();
77.     dataUNormal>NamaData = training>NamaData;
78.     dataUNormal.Umur = Normalisasi(training.Umur,
79.         Min(minLUmur, minUUmur),
80.         Max(maxLUmur, maxUUmur));
81.     dataUNormal.LDL = Normalisasi(training.LDL,
82.         Min(minLLDL, minULDL), Max(maxLLDL,
83.         maxULDL));
84.     dataUNormal.Kolesterol = Normalisasi(training.Kolesterol,
85.         Min(minLKolesterol, minUKolesterol),
86.         Max(maxLKolesterol, maxUKolesterol));
87.     dataUNormal.HDL = Normalisasi(training.HDL, Min(minLHdl,
88.         minUHdl), Max(maxLHdl, maxUHdl));
89.     dataUNormal.Trigliserida =
90.         Normalisasi(training.Trigliserida,
91.         Min(minLTrigliserida, minUTrigliserida),
92.         Max(maxLTrigliserida, maxUTrigliserida));
93.     dataUNormal.Sistolik = Normalisasi(training.Sistolik,
94.         Min(minLSistolik, minUSistolik),
95.         Max(maxLSistolik, maxUSistolik));
96.     dataUNormal.Nilai = training.Nilai;
97.     dataUNormal.Kategori = training.Kategori;
98.     normalUji.Add(dataUNormal);
99. }
100. inputDataBindingSource2.DataSource = normalUji;
101.
102. ///////////NORMALISASI DATA LATIH//////////
103. normalLatih = new List<InputData>();
104. foreach (var latih in dataLatih){
105.     InputData dataLNormal = new InputData();
106.     dataLNormal>NamaData = latih>NamaData;
107.     dataLNormal.NilaiK = latih.NilaiK;
108.     dataLNormal.Umur = Normalisasi(latih.Umur, Min(minLUmur,
109.         minUUmur), Max(maxLUmur, maxUUmur));
110.     dataLNormal.LDL = Normalisasi(latih.LDL, Min(minLLDL,
111.         minULDL), Max(maxLLDL, maxULDL));
112.     dataLNormal.Kolesterol = Normalisasi(latih.Kolesterol,
113.         Min(minLKolesterol, minUKolesterol),
114.         Max(maxLKolesterol, maxUKolesterol));
115.     dataLNormal.HDL = Normalisasi(latih.HDL, Min(minLHdl,
116.         minUHdl), Max(maxLHdl, maxUHdl));
117.     dataLNormal.Trigliserida =
118.         Normalisasi(latih.Trigliserida,
119.         Min(minLTrigliserida, minUTrigliserida),
120.         Max(maxLTrigliserida, maxUTrigliserida));
121. }
122.
123.
124.
```

```

125.                                     Max(maxLTrigliserida,
126.                                     maxUTrigliserida));
127.     dataLNormal.Sistolik = Normalisasi(latih.Sistolik,
128.                                     Min(minLSistolik, minUSistolik),
129.                                     Max(maxLSistolik, maxUSistolik));
130.     dataLNormal.Nilai = latih.Nilai;
131.     dataLNormal.Kategori = latih.Kategori;
132.     normalLatih.Add(dataLNormal);
133. }
134. inputDataBindingSource3.DataSource = normalLatih;

```

Sourcecode 4. 10 Normalisasi setiap atribut

Perhitungan nilai bobot setiap atribut sesuai dengan kelas pada data latih, dengan menggunakan bobot “mean”. Implementasi perhitungan bobot di tampilkan pada *sourcecode* 4.11.

```

1.  List<InputData> bobot = new List<InputData>();
2.      List<String> kategoriLatih = normalLatih.Select(n =>
3.          n.Kategori).Distinct().ToList();
4.      foreach (var item in kategoriLatih)
5.      {
6.          List<InputData> dataKategori =
7.              normalLatih.Where(n =>
8.                  n.Kategori == item).ToList();
9.          InputData bobotKategori = new InputData();
10.         bobotKategori.Umur = dataKategori.Average(d =>
11.             d.Umur);
12.         bobotKategori.LDL = dataKategori.Average(d =>
13.             d.LDL);
14.         bobotKategori.Kolesterol =
15.             dataKategori.Average(d =>
16.                 d.Kolesterol);
17.         bobotKategori.HDL = dataKategori.Average(d =>
18.             d.HDL);
19.         bobotKategori.Trigliserida =
20.             dataKategori.Average(d =>
21.                 d.Trigliserida);
22.         bobotKategori.Sistolik =
23.             dataKategori.Average(d
24.                 => d.Sistolik);
25.         bobotKategori.Kategori = item;
26.         bobot.Add(bobotKategori);
27.     }
28.

```

Sourcecode 4. 11 bobot setiap atribut (nilai tengah atribut kelas)

5.2.2.3. Implementasi K-Nearest Neighbor

Dalam proses klasifikasi, terdapat tiga proses diantaranya yaitu proses pencarian jarak dengan menggunakan metode *Euclidean distance*, kemudian pembobotan dengan menggunakan metode *Weight voting* dan setelah itu pencarian nilai *Weight* dengan sejumlah k.

5.2.2.3.1. Implementasi Euclidean Distance

Data inputan pada implementasi *Euclidean distance* berupa nilai normalisasi dari data uji terhadap normalisasi data latih. Proses perhitungan jarak dengan metode *Euclidean distance* dilakukan oleh method `HitungEuclidean()` dengan parameter `Input` berupa data `InputData` untuk data uji dan data latih. Implementasi dari proses ini akan di tampilkan pada sourcecode 4.12.

```

1. private Euclidean HitungEuclidean(InputData latih, InputData
2.     uji, List<InputData> bobot) {
3.
4.     Euclidean dataEuclidean = new Euclidean();
5.
6.     dataEuclidean>NamaData = latih>NamaData;
7.     dataEuclidean.Nilai = latih.Nilai;
8.     dataEuclidean.Kategori = latih.Kategori;
9.     dataEuclidean.SangatRendah = latih.SangatRendah;
10.    dataEuclidean.Rendah = latih.Rendah;
11.    dataEuclidean.Sedang = latih.Sedang;
12.    dataEuclidean.Tinggi = latih.Tinggi;
13.    dataEuclidean.SangatTinggi = latih.SangatTinggi;
14.
15.    InputData bobotkategori = bobot.Where(b => b.Kategori ==
16.        latih.Kategori).Single<InputData>();
17.
18.    dataEuclidean>DataEucli = Math.Sqrt(((bobotkategori.Umur
19.        * (Kuadrat(uji.Umur - latih.Umur)) +
20.        (bobotkategori.LDL * (Kuadrat(uji.LDL -
21.        latih.LDL))) + (bobotkategori.Kolesterol *
22.        (Kuadrat(uji.Kolesterol -
23.        latih.Kolesterol))) + (bobotkategori.HDL *
24.        (Kuadrat(uji.HDL - latih.HDL))) +
25.        (bobotkategori.Trigliserida *
26.        (Kuadrat(uji.Trigliserida -
27.        latih.Trigliserida))) +
28.        (bobotkategori.Sistolik *
29.        (Kuadrat(uji.Sistolik - latih.Sistolik)))) /
30.        (bobotkategori.Umur + bobotkategori.LDL +
31.        bobotkategori.Kolesterol + bobotkategori.HDL
32.        + bobotkategori.Trigliserida +
33.        bobotkategori.Sistolik));
34.
35.    return dataEuclidean;
36. }

```

Sourcecode 4. 12 Implementasi HitungEuclidean()

Pemberian nilai *Euclidean distance* pada setiap data uji dengan pusat cluster data latih. Implementasi proses ini ditampilkan pada *sourcecode* 4.13. Pada program nilai jarak *Euclidean* untuk tiap data latih akan disimpan pada variable `dataEucli` yang merupakan *list* dengan tipe data objek *Euclidean*. Selanjutnya data jarak *Euclidean* tiap data uji akan disimpan pada variable `listEucli`.

```

1. List<List<Euclidean>> listEucli = new List<List<Euclidean>>();

```

```

2.      foreach (var uji in normalUji)
3.      {
4.          List<Euclidean> dataEucli = new List<Euclidean>();
5.          foreach (var latih in normalLatih)
6.          {
7.              dataEucli.Add(HitungEuclidean(latih, uji,bobot));
8.          }
9.          listEucli.Add(dataEucli);
10.     }
11.

```

Sourcecode 4. 13 Pemberian nilai Euclidean pada setiap data uji

5.2.2.3.2. Implementasi *Weight*

Dalam proses *Weight*, data inputan merupakan hasil dari proses *HitungEuclidean()*. Dimana nilai *weight* didapat dari satu per hasil dari *HitungEuclidean()* kuadrat. Implementasi dari proses ini akan di tampilkan pada *sourcecode 4.14*.

```

1.      private Weight HitungWeight(Euclidean inputEucli)
2.      {
3.          Weight dataWeight = new Weight();
4.
5.          dataWeight>NamaData = inputEucli>NamaData;
6.          dataWeight.Nilai = inputEucli.Nilai;
7.          dataWeight.Kategori = inputEucli.Kategori;
8.          dataWeight.SangatRendah = inputEucli.SangatRendah;
9.          dataWeight.Rendah = inputEucli.Rendah;
10.         dataWeight.Sedang = inputEucli.Sedang;
11.         dataWeight.Tinggi = inputEucli.Tinggi;
12.         dataWeight.SangatTinggi = inputEucli.SangatTinggi;
13.         dataWeight.DataWeight = 1 /
14.             Math.Pow(inputEucli.DataEucli, 2);
15.
16.         return dataWeight;}

```

Sourcecode 4. 14 Implementasi *HitungWeight()*

Perhitungan nilai *Weight* pada setiap data latih terhadap data uji dengan *listEucli* disimpan pada variable *dataWeight*. Adapun tiap data uji nilai *weight* jaraknya akan disimpan pada variable *ListWeight()*. Implementasi proses ini ditampilkan pada *sourcecode 4.15*.

```

1.      List<List<Weight>> listWeight = new List<List<Weight>>();
2.
3.      foreach (var w in listEucli)
4.      {
5.          List<Weight> dataWeight = new List<Weight>();
6.          foreach (var eachData in w)
7.          {
8.              Weight data = HitungWeight(eachData);
9.              data.Nilai = eachData.Nilai;
10.             data>NamaData = eachData>NamaData;
11.             data.Kategori = eachData.Kategori;

```

```

12.         data.Rendah = eachData.Rendah;
13.         data.SangatRendah = eachData.SangatRendah;
14.         data.Sedang = eachData.Sedang;
15.         data.Tinggi = eachData.Tinggi;
16.         data.SangatTinggi = eachData.SangatTinggi;
17.         dataWeight.Add(data);
18.     }
19.     listWeight.Add(dataWeight);

```

Sourcecode 4. 15 Pemberian nilai Weight pada setiap data

5.2.2.3.3. Implementasi *Weight* sejumlah k

Setelah didapat nilai *weight*, nilai tersebut akan disortir dari nilai *weight* tertinggi ke rendah dengan method `OrderByDescending()` dengan parameter data `DataWeight`. Selanjutnya akan diambil sejumlah data sejumlah k yang memiliki nilai *weight* tertinggi dengan fungsi `Take()`. Jika nilai k yang di ambil adalah empat, maka nilai *weight* tersebut akan di ambil empat nilai terbesar. Implementasi ini akan di tampilkan pada *sorcecode* 4.16.

```

1.     int inpK = int.Parse(kComboBox.SelectedItem.ToString());
2.
3.     List<List<Weight>> listWeightK = new List<List<Weight>>();
4.
5.     foreach (var k in listWeight){
6.         List<Weight> dataWeightK = k.OrderByDescending(l =>
7.             l.DataWeight).Take(inpK).ToList
8.             <Weight>();
9.         listWeightK.Add(dataWeightK);
10.    }

```

Sourcecode 4. 16 Implementasi weight k

5.2.3. Implementasi Transformasi ke data *fuzzy*

Dalam implementasi transformasi data ke *fuzzy*, dimana setiap data akan disesuaikan dengan *membership*nya. Proses perhitungan nilai *membership* akan dilakukan oleh *method* `HitungNilaiMembership()`. Pada *method* ini data uji akan dihitung nilai *membership*nya berdasarkan data *weight* pada KNN yang dikalikan dengan *membership*nya masing-masing. Implementasi dari proses ini akan di tampilkan pada *sourcecode* 4.17.

```

1.     private double HitungNilaiMembership(List<Weight> inputWK,
2.         String namaMembership,
3.         double totalWK){
4.
5.         double hasil = 0;
6.         double mkaliWk = 0;
7.
8.         foreach (var m in inputWK){
9.             double membershipK = 0;
10.            if (namaMembership.Equals("Sangat Rendah")){

```

```

10.         membershipK = m.SangatRendah;
11.     else if (namaMembership.Equals("Rendah")){
12.         membershipK = m.Rendah;
13.     else if (namaMembership.Equals("Sedang")){
14.         membershipK = m.Sedang;
15.     else if (namaMembership.Equals("Tinggi")){
16.         membershipK = m.Tinggi;
17.     else if (namaMembership.Equals("Sangat_Tinggi")){
18.         membershipK = m.SangatTinggi;
19.         mkaliWk += m.DataWeight * membershipK;
20.     }
21.     hasil = mkaliWk / totalWK;
22.     return hasil;
23. }

```

Sourcecode 4. 17 Impelementasi HitungNilaiMembership()

Perhitungan nilai *membership* pada setiap data disimpan pada variabel *dataMembership*. Adapun tiap data *membership* jaraknya akan disimpan pada variable *ListMembership()*. Dimana terdapat 4 seleksi penyimpanan *membership*, dimana pada *sourcecode* baris 6 untuk seleksi pertama untuk semua *membership* mulai dari sangat rendah, rendah, sedang, tinggi dan sangat tinggi. Selanjutnya seleksi berikutnya mulai dari *sourcecode* baris 54, untuk *membership* sangat rendah, rendah dan sedang. Seleksi berikutnya mulai baris 83 untuk *membership* rendah, sedang dan tinggi. Dan seleksi terakhir mulai baris 112 untuk *membership* sedang, tinggi dan sangat tinggi. Implementasi proses ini ditampilkan pada *sourcecode* 4.18.

```

1.     List<List<Membership>> dataMembership = new
2.         List<List<Membership>>();
3.
4.     for (int i = 0; i < listWeightK.Count; i++){
5.
6.         // seleksi pertama (Sangat Rendah-Rendah-Sedang-
7.         Tinggi-Sangat tinggi)
8.
9.         if (kryptonRadioButton1.Checked){
10.             List<Membership> listMembership = new
11.                 List<Membership>();
12.             Membership sangatRendah = new Membership();
13.             sangatRendah>NamaMembership = "Sangat_Rendah";
14.             sangatRendah.NilaiMembership =
15.                 HitungNilaiMembership(listWeightK[i],
16.                 sangatRendah>NamaMembership,
17.                 totalWeightK[i]);
18.             listMembership.Add(sangatRendah);
19.
20.             Membership rendah = new Membership();
21.             rendah>NamaMembership = "Rendah";
22.             rendah.NilaiMembership =
23.                 HitungNilaiMembership(listWeightK[i],
24.                 rendah>NamaMembership,
25.                 totalWeightK[i]);

```

```

26.         listMembership.Add(rendah);
27.
28.         Membership sedang = new Membership();
29.         sedang>NamaMembership = "Sedang";
30.         sedang.NilaiMembership =
31.             HitungNilaiMembership(listWeightK[i],
32.                 sedang>NamaMembership,
33.                 totalWeightK[i]);
34.         listMembership.Add(sedang);
35.
36.         Membership tinggi = new Membership();
37.         tinggi>NamaMembership = "Tinggi";
38.         tinggi.NilaiMembership =
39.             HitungNilaiMembership(listWeightK[i],
40.                 tinggi>NamaMembership,
41.                 totalWeightK[i]);
42.         listMembership.Add(tinggi);
43.
44.         Membership sangatTinggi = new Membership();
45.         sangatTinggi>NamaMembership = "Sangat_Tinggi";
46.         sangatTinggi.NilaiMembership =
47.             HitungNilaiMembership(listWeightK[i],
48.                 sangatTinggi>NamaMembership,
49.                 totalWeightK[i]);
50.         listMembership.Add(sangatTinggi);
51.
52.         dataMembership.Add(listMembership);}
53.
54.         // seleksi kedua (Sangat Rendah-Rendah-Sedang)
55.         else if (kryptonRadioButton2.Checked){
56.             List<Membership> listMembership = new
57.                 List<Membership>();
58.             Membership sangatRendah = new Membership();
59.             sangatRendah>NamaMembership = "Sangat_Rendah";
60.             sangatRendah.NilaiMembership =
61.                 HitungNilaiMembership(listWeightK[i],
62.                     sangatRendah>NamaMembership,
63.                     totalWeightK[i]);
64.             listMembership.Add(sangatRendah);
65.
66.             Membership rendah = new Membership();
67.             rendah>NamaMembership = "Rendah";
68.             rendah.NilaiMembership =
69.                 HitungNilaiMembership(listWeightK[i],
70.                     rendah>NamaMembership,
71.                     totalWeightK[i]);
72.             listMembership.Add(rendah);
73.
74.             Membership sedang = new Membership();
75.             sedang>NamaMembership = "Sedang";
76.             sedang.NilaiMembership =
77.                 HitungNilaiMembership(listWeightK[i],
78.                     sedang>NamaMembership,
79.                     totalWeightK[i]);
80.             listMembership.Add(sedang);
81.             dataMembership.Add(listMembership);}
82.
83.         // seleksi ketiga (Rendah-Sedang-Tinggi)
84.         else if (kryptonRadioButton3.Checked){
85.             List<Membership> listMembership = new
86.                 List<Membership>();
87.             Membership rendah = new Membership();
88.             rendah>NamaMembership = "Rendah";
89.             rendah.NilaiMembership =

```

```

90.         HitungNilaiMembership(listWeightK[i],
91.         rendah>NamaMembership,
92.         totalWeightK[i]);
93.         listMembership.Add(rendah);
94.
95.         Membership sedang = new Membership();
96.         sedang>NamaMembership = "Sedang";
97.         sedang.NilaiMembership =
98.         HitungNilaiMembership(listWeightK[i],
99.         sedang>NamaMembership,
100.         totalWeightK[i]);
101.         listMembership.Add(sedang);
102.
103.         Membership tinggi = new Membership();
104.         tinggi>NamaMembership = "Tinggi";
105.         tinggi.NilaiMembership =
106.         HitungNilaiMembership(listWeightK[i],
107.         tinggi>NamaMembership,
108.         totalWeightK[i]);
109.         listMembership.Add(tinggi);
110.         dataMembership.Add(listMembership);}
111.
112.         // seleksi keempat (Sedang-Tinggi-SangatTinggi)
113.         else if (kryptonRadioButton4.Checked){
114.             List<Membership> listMembership = new
115.             List<Membership>();
116.             Membership sedang = new Membership();
117.             sedang>NamaMembership = "Sedang";
118.             sedang.NilaiMembership =
119.             HitungNilaiMembership(listWeightK[i],
120.             sedang>NamaMembership,
121.             totalWeightK[i]);
122.             listMembership.Add(sedang);
123.
124.             Membership tinggi = new Membership();
125.             tinggi>NamaMembership = "Tinggi";
126.             tinggi.NilaiMembership =
127.             HitungNilaiMembership(listWeightK[i],
128.             tinggi>NamaMembership,
129.             totalWeightK[i]);
130.             listMembership.Add(tinggi);
131.
132.             Membership sangatTinggi = new Membership();
133.             sangatTinggi>NamaMembership = "Sangat_Tinggi";
134.             sangatTinggi.NilaiMembership =
135.             HitungNilaiMembership(listWeightK[i],
136.             sangatTinggi>NamaMembership,
137.             totalWeightK[i]);
138.             listMembership.Add(sangatTinggi);
139.             dataMembership.Add(listMembership);}}

```

Sourcecode 4. 18 Pemberian nilai Membership pada setiap data

Untuk perhitungan *membership* akan dihitung dari setiap *linguistic* yang ada, yaitu *linguistic* sangat rendah, rendah, sedang, tinggi dan sangat tinggi. Untuk implementasi dari proses ini dapat di tampilkan pada *sourcecode* 4.19 – 4.23.

Untuk *Membership* Sangat Rendah, terdapat tiga seleksi. Untuk seleksi pertama jika nilai lebih besar sama dengan 0 dan nilai kurang dari 5, maka nilai

tersebut = 1. Jika tidak, maka akan masuk ke dalam seleksi kedua, dimana jika nilai lebih besar sama dengan 5 dan nilai kurang dari 15 maka nilai tersebut akan di hitung sesuai persamaan untuk mendapatkan nilainya. Untuk seleksi ke tiga, yaitu jika nilai lebih besar sama dengan 15, maka nilai = 0.

```

1. public double SangatRendah{
2.     get{
3.         double hasilSR = 0;
4.         if ((nilai >= 0) && (nilai < 5)){
5.             hasilSR = 1;}
6.         else if ((nilai >= 5) && (nilai < 15)){
7.             hasilSR = (nilai - 5.0) / 10.0;}
8.         else if (nilai >= 15){
9.             hasilSR = 0;}
10.        return hasilSR;}
11.    }

```

Sourcecode 4. 19 Fungsi membership SangatRendah()

Untuk *Membership Rendah*, juga terdapat tiga seleksi. Untuk seleksi pertama jika nilai lebih besar sama dengan 15 dan nilai kurang dari 25, maka nilai tersebut akan dihitung sesuai persamaan. Jika tidak, maka akan masuk ke dalam seleksi kedua, dimana jika nilai lebih besar sama dengan 5 dan nilai kurang dari 15 maka nilai tersebut akan di hitung untuk mendapatkan nilainya. Untuk seleksi ke tiga, yaitu jika nilai lebih besar sama dengan 5 atau nilai lebih besar sama dengan 25, maka nilai = 0.

```

1. public double Rendah{
2.     get{
3.         double hasilR = 0;
4.         if ((nilai >= 15) && (nilai < 25)){
5.             hasilR = (25 - nilai) / 10.0;}
6.         else if ((nilai >= 5) && (nilai < 15)){
7.             hasilR = (nilai - 5.0) / 10.0;}
8.         else if ((nilai <= 5) || (nilai >= 25)){
9.             hasilR = 0;}
10.        return hasilR;}
11.    }

```

Sourcecode 4. 20 Fungsi membership Rendah()

Untuk *Membership Sedang*, juga terdapat tiga seleksi. Untuk seleksi pertama jika nilai lebih besar sama dengan 25 dan nilai kurang dari 35, maka nilai tersebut akan dihitung sesuai persamaan. Jika tidak, maka akan masuk ke dalam seleksi kedua, dimana jika nilai lebih besar sama dengan 15 dan nilai kurang dari 25 maka nilai tersebut akan di hitung untuk mendapatkan nilainya. Untuk seleksi

ke tiga, yaitu jika nilai lebih besar sama dengan 15 atau nilai lebih besar sama dengan 35, maka nilai = 0.

```

1. public double Sedang{
2.     get{
3.         double hasilS = 0;
4.         if ((nilai >= 25) && (nilai <= 35)){
5.             hasilS = (35 - nilai) / 10.0;}
6.         else if ((nilai >= 15) && (nilai < 25)){
7.             hasilS = (nilai - 15.0) / 10.0;}
8.         else if ((nilai <= 15) || (nilai >= 35)){
9.             hasilS = 0;}
10.        return hasilS;}
11.    }

```

Sourcecode 4. 21 Fungsi membership Sedang()

Untuk *Membership Tinggi*, juga terdapat tiga seleksi. Untuk seleksi pertama jika nilai lebih besar sama dengan 35 dan nilai kurang dari 45, maka nilai tersebut akan dihitung sesuai persamaan. Jika tidak, maka akan masuk ke dalam seleksi kedua, dimana jika nilai lebih besar sama dengan 25 dan nilai kurang dari 35 maka nilai tersebut akan di hitung untuk mendapatkan nilainya. Untuk seleksi ke tiga, yaitu jika nilai lebih besar sama dengan 25 atau nilai lebih besar sama dengan 45, maka nilai = 0.

```

1. public double Tinggi{
2.     get{
3.         double hasilT = 0;
4.         if ((nilai >= 35) && (nilai < 45)){
5.             hasilT = (45 - nilai) / 10.0;}
6.         else if ((nilai >= 25) && (nilai < 35)){
7.             hasilT = (nilai - 25.0) / 10.0;}
8.         else if ((nilai <= 25) || (nilai >= 45)){
9.             hasilT = 0;}
10.        return hasilT;}
11.    }

```

Sourcecode 4. 22 Fungsi membership Tinggi()

Untuk *Membership Sangat Tinggi*, juga terdapat tiga seleksi. Untuk seleksi pertama jika nilai lebih besar sama dengan 45 maka nilai tersebut sama dengan 1. Jika tidak, maka akan masuk ke dalam seleksi kedua, dimana jika nilai lebih besar sama dengan 35 dan nilai kurang dari 45 maka nilai tersebut akan di hitung untuk mendapatkan nilainya. Untuk seleksi ke tiga, yaitu jika nilai kurang dari sama dengan 35 maka nilai = 0.

```

1. public double SangatTinggi{

```

```

2.      get{
3.          double hasilST = 0;
4.          if (nilai >= 45){
5.              hasilST = 1;}
6.          else if ((nilai >= 35) && (nilai < 45)){
7.              hasilST = (nilai - 35) / 10.0;}
8.          else if (nilai <= 35){
9.              hasilST = 0;}
10.         return hasilST;}
11.     }

```

Sourcecode 4. 23 Fungsi membership SangatTinggi()

5.3. Implementasi Antarmuka

Impelementasi antarmuka dari sistem ini terdiri dari dua form bagian utama, yaitu:

1. Form Prediksi

Pada form prediksi digunakan sebagai antarmuka untuk melakukan pengujian dengan cara menginputkan data tunggal secara manual dan kemudian akan diketahui hasil prediksi dan nilai *membership*-nya.

2. Form Pengujian

Pada form pengujian digunakan sebagai antarmuka untuk melakukan pengujian prediksi tingkat risiko penyakit jantung koroner berdasarkan klasifikasi dan *membership*nya. Dimana hasil dari pengujian dapat diketahui tingkat akurasi sistem.

5.3.1. Form Prediksi FK-NN

Pada gambar 4.1 merupakan antarmuka prediksi data dengan menginputkan data secara manual. Awalnya dimasukkan data latih yang digunakan sebagai acuan dari data yang baru. Setelah selesai diinputkan data uji dengan mengisi seluruh atribut yang dibutuhkan. Kemudian tekan tombol proses untuk mengetahui hasil prediksi. Pada form prediksi ini akan langsung didapatkan hasil prediksi penyakit jantung koroner (sangat rendah, rendah, sedang, tinggi dan tinggi).

Gambar 4. 1 Antarmuka Prediksi FK-NN

Keterangan Gambar 4.1:

1. *Textbox* yang digunakan untuk menginputkan data latih, dari file berupa .txt.
2. Tombol *browse* digunakan untuk mencari data yang akan diinputkan ke dalam sistem.
3. Tombol input data digunakan untuk memasukkan data dan menampilkan data ke dalam *datagridview*.
4. *Textbox* yang digunakan untuk menginputkan data uji yang meliputi 6 atribut.
5. *Combobox* untuk memilih inputan nilai k.
6. Tombol proses digunakan untuk melakukan proses prediksi.
7. Tab FK-NN untuk menampilkan hasil akhir prediksi FK-NN dalam bentuk *datagridview*, dengan menampilkan nama *membership* dan nilai *memberhip*.
8. *Labeltext* yang digunakan untuk menampilkan hasil prediksi yang berupa kategori.

9. *LabelText* yang digunakan untuk menampilkan hasil prediksi yang berupa nilai *membership*.

Pada proses prediksi diatas, data uji yang dimasukkan merupakan data tunggal. Dimana data yang akan diuji dimasukkan pada *textbox* yang telah tersedia untuk 6 atribut yang ada. Selain 6 atribut, juga dimasukkan nilai *k* pada *checkbox*. Kemudian dilakukan proses prediksi dengan menekan tombol proses. Hasil yang akan diperoleh merupakan hasil prediksi dari FK-NN. Prediksi dengan input data tunggal ini dapat dilihat pada gambar 4.2.

PJK dengan Fuzzy K-NN

Prediksi Tingkat Risiko Penyakit Jantung Koroner (PJK) dengan Menggunakan Fuzzy k-Nearest Neighbor

Data Uji

Input Data Uji :

C:\Users\chaca\Documents\...

Umur : 51

LDL : 196

Kolesterol : 200

HDL : 72

Trigliserida : 159

Sistolik : 140

Input k : 4

Prediksi

NamaMembership	NilaiMembership
Sangat Rendah	0.056829050248174369
Rendah	0.29377167345047378
Sedang	0.3089945818110908
Tinggi	0.26463262749269517
Sangat Tinggi	0

Hasil Prediksi FKNN

Membership : **Sedang**

Nilai : **0.308994581811091**

Gambar 4. 2 Antarmuka Proses prediksi dengan input data tunggal

Prediksi dengan input data tunggal digunakan untuk memprediksi data baru yang masih belum diketahui tingkat risiko PJK-nya. Pada *tab* prediksi akan ditampilkan hasil akhir sistem prediksi, dengan menampilkan prediksi kelas dan nilai dari *membership* data baru.

5.3.2. Antarmuka Pengujian FK-NN

Form pengujian merupakan antarmuka pengujian yang digunakan untuk melakukan prediksi tingkat risiko penyakit jantung koroner. Pada form ini terdapat 5 *tab*, yaitu untuk menampilkan *datagridview* dari data uji, data latih, *weight k*, *membership* dan akurasi. Gambar dari *form* pengujian ditunjukkan pada gambar 4.3.

Gambar 4. 3 Antarmuka Pengujian FK-NN

Keterangan Gambar 4.3:

1. *Textbox* yang digunakan untuk menginputkan data uji, dari file berupa .txt.
2. Tombol *browse* digunakan untuk mencari data uji yang akan diinputkan ke dalam sistem.
3. Tombol input data digunakan untuk memasukkan data uji dan menampilkan data ke dalam *datagridview*.
4. *Textbox* yang digunakan untuk menginputkan data latih, dari file berupa .txt.
5. Tombol *browse* digunakan untuk mencari data latih yang akan diinputkan ke dalam sistem.

6. Tombol input data digunakan untuk memasukkan data latih dan menampilkan data ke dalam *datagridview*.
7. *Combobox* untuk memilih inputan nilai k.
8. *Radiobutton* digunakan untuk memasukkan akurasi data.
9. *Textbox* untuk member keterangan dari *radiobutton* pada *groupbox* akurasi.
10. Tombol proses digunakan untuk melakukan proses prediksi.
11. *Tab* data uji untuk menampilkan data uji yang telah dimasukkan.
12. *Tab* data latih untuk menampilkan data latih yang telah dimasukkan.
13. *Tab weight K* untuk menampilkan *datagridview* dari data yang telah diproses sampai dengan proses *weight* berdasarkan nilai k yang dimasukkan.
14. *Tab membership* untuk menampilkan nilai dari data uji berdasarkan *membership* kelasnya.
15. *Tab* akurasi untuk menampilkan hasil akhir pengujian sistem FK-NN dalam bentuk *datagridview*.

Seperti yang telah dijelaskan diatas bahwa *form* ini memiliki 5 *tab* yaitu *tab* data uji, data latih, *weight k*, *membership* dan akurasi. Pada *tab* data uji akan ditampilkan data uji yang telah dimasukkan. Tampilan pada *tab* data uji dapat dilihat pada gambar 4.4.

Data Uji

Input Data Uji :
C:\Users\chaca\Documents\Visu...
Input Data

Data Latih

Input Data Latih :
C:\Users\chaca\Documents\Visu...
Input Data

Input Nilai k :
Input k : 7

Akurasi

Keterangan

☒ SR-R-S-T-ST SR : Sangat Rendah
☐ SR-R-S R : Rendah
☐ R-S-T S : Sedang
☐ S-T-ST T : Tinggi
ST : Sangat Tinggi

Proses

NamaData	Umur	LDL	Kolesterol	HDL	Triglicerida	Sistolik
Data Uji-1	45	123	266	54	147	200
Data Uji-2	67	143	221	25	267	130
Data Uji-3	49	156	240	55	138	160
Data Uji-4	66	190	165	62	116	170
Data Uji-5	70	161	201	38	110	170
Data Uji-6	58	154	230	37	224	140
Data Uji-7	67	280	232	47	294	140
Data Uji-8	68	153	201	39	298	180
Data Uji-9	75	190	148	42	174	150
Data Uji-10	46	292	152	40	185	120
Data Uji-11	41	176	236	47	205	150
Data Uji-12	32	192	167	40	155	170
Data Uji-13	68	138	212	37	229	180
Data Uji-14	44	174	203	47	145	160
Data Uji-15	60	136	193	37	140	180
Data Uji-16	63	143	214	49	212	130
Data Uji-17	53	135	185	38	137	170
Data Uji-18	62	113	183	43	144	170

Gambar 4. 4 Antarmuka Pengujian pada tab data uji

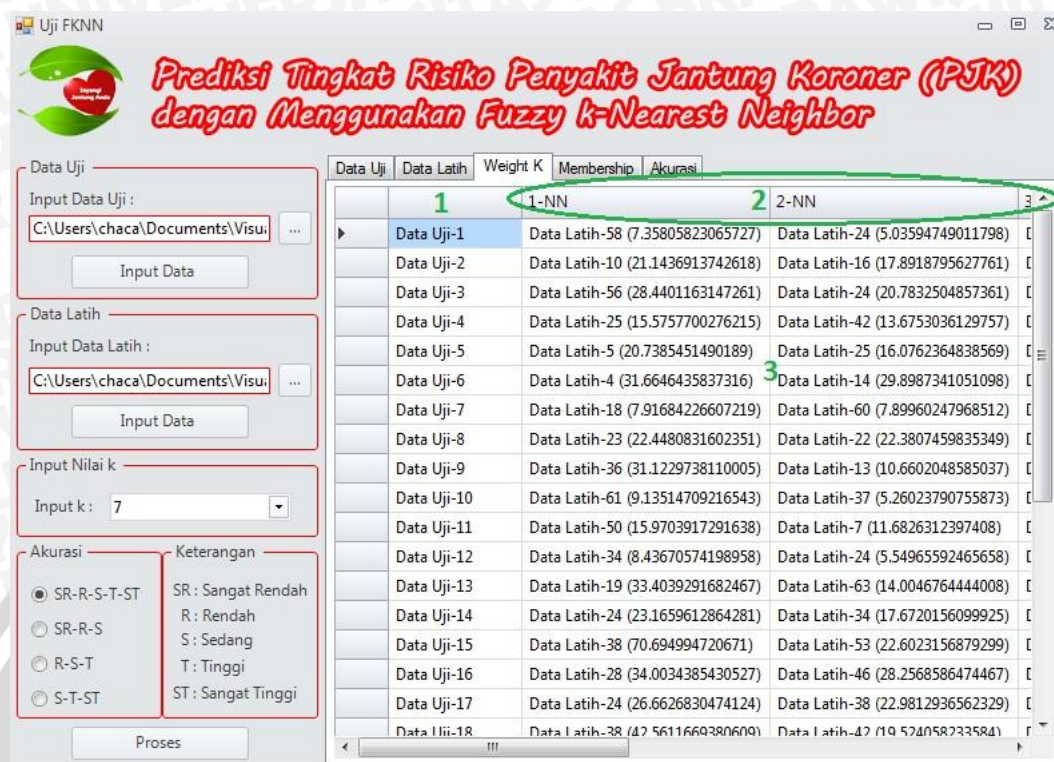
Sama dengan *tab* data uji, pada *tab* data latih juga akan ditampilkan data latih yang telah dimasukkan. Dimana pada *datagridview* akan ditampilkan seluruh data uji dengan nilai setiap atributnya. Tampilan pada *tab* data latih dapat dilihat pada gambar 4.5.

NamaData	Umur	LDL	Kolesterol	HDL	Triglicerida	Sistolik
Data Latih-1	57	115	218	66	274	150
Data Latih-2	60	171	182	57	286	140
Data Latih-3	58	134	223	25	130	150
Data Latih-4	61	167	231	39	254	130
Data Latih-5	77	148	208	33	134	180
Data Latih-6	77	186	168	66	177	130
Data Latih-7	41	185	208	50	236	130
Data Latih-8	38	119	289	41	232	140
Data Latih-9	63	186	197	40	319	120
Data Latih-10	60	150	218	32	284	140
Data Latih-11	51	196	200	72	159	140
Data Latih-12	47	192	171	41	350	120
Data Latih-13	67	191	196	53	188	140
Data Latih-14	59	176	245	44	233	150
Data Latih-15	50	166	236	35	284	130
Data Latih-16	65	137	214	13	302	140
Data Latih-17	62	178	215	35	258	120
Data Latih-18	68	256	221	48	190	140

Gambar 4. 5 Antarmuka Pengujian pada *tab* data latih

Untuk *tab* berikutnya adalah *tab* *Weight* FK-NN, dimana pada *tab* ini ditampilkan hasil dari nilai *weight* setiap data uji terhadap data latih terdekatnya. Dimana nilai terdekat pada *weight* merupakan nilai terbesar yang dimiliki data latih. *Weight* K merupakan pengambilan *weight* berdasarkan nilai K yang dimasukkan.

Pada gambar diatas ditunjukkan bahwa nilai untuk k yang dimasukkan adalah 7, sehingga nilai *weight* K akan diambil 7 nilai *weight* setiap data latih. Setelah itu nilai *weight* tersebut diurutkan untuk mendapatkan 7 nilai terbesar *weight* yang dimiliki data uji dengan data latih. Gambar dari *tab* *weight* k ditunjukkan pada gambar 4.6.

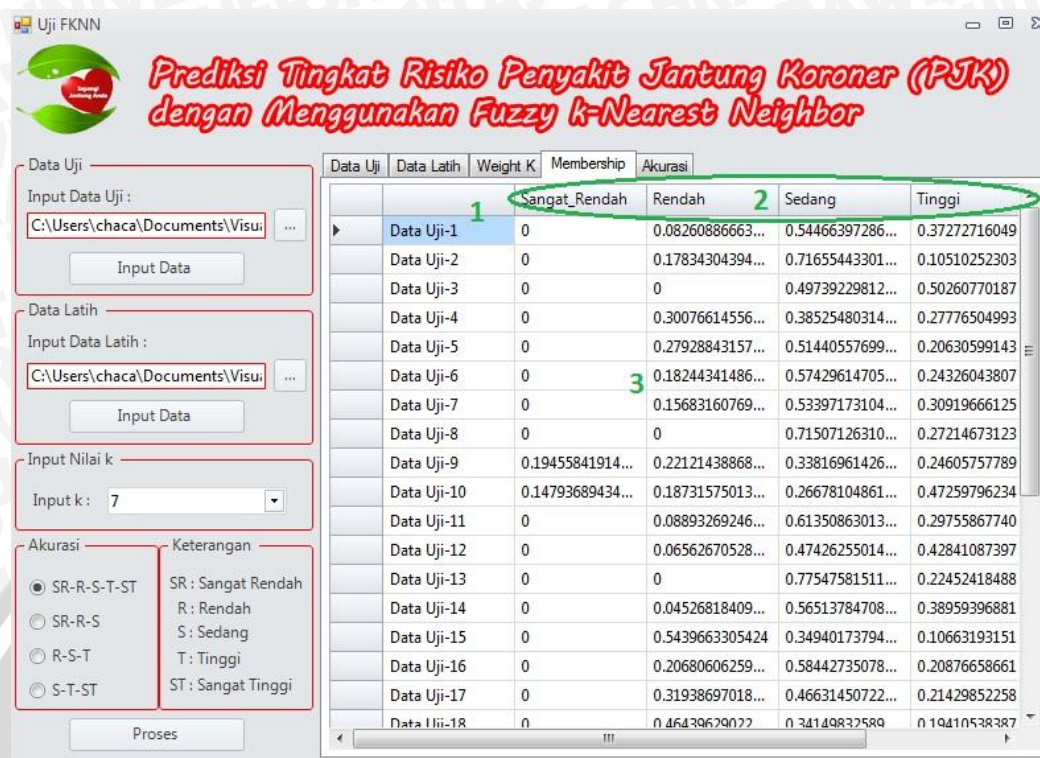


Gambar 4. 6 Antarmuka Pengujian pada tab weight k

Keterangan Gambar 4.6:

1. *Datagridview* pada kolom 1 merupakan nama untuk setiap data uji.
2. *Datagridview header* untuk menunjukkan nilai *sorting* data mulai dari yang tertinggi. Dimana 1-NN menunjukkan untuk nilai tertinggi pertama, 2-NN nilai tertinggi kedua sampai dengan 7-NN, sesuai dengan nilai *k* yang dimasukkan.
3. *Datagridview body* menampilkan nilai tertinggi *weight* dari data uji dengan data latihnya.

Pada tab *membership* akan ditunjukkan nilai *membership* setiap data uji dalam setiap kelasnya. Dimana terdapat lima kelas atau kategori pada sistem, yaitu sangat rendah, rendah, sedang, tinggi dan sangat tinggi. Setiap kategori memiliki nilai, dikarenakan setiap data memiliki nilai disetiap *linguistic*-nya. Gambar dari tab *membership* ditunjukkan pada gambar 4.7.



Gambar 4. 7 Antarmuka Pengujian pada tab Membership

Keterangan Gambar 4.7:

1. *Datagridview* pada kolom 1 merupakan nama untuk setiap data uji.
2. *Datagridview header* untuk menunjukkan kategori atau kelas.
3. *Datagridview body* menampilkan nilai *membership* data uji di setiap kelas atau *linguistic*-nya.

Pada tab Akurasi akan ditampilkan hasil prediksi sistem dan prediksi sebenarnya. Dimana dari kedua hasil tersebut dapat dicari nilai akurasinya. Gambar dari *tab* akurasi ditunjukkan pada gambar 4.8.

Prediksi Tingkat Risiko Penyakit Jantung Koroner (PJK) dengan Menggunakan Fuzzy k-Nearest Neighbor

Data Uji: Input Data Uji: C:\Users\chaca\Documents\Visu... Input Data

Data Latih: Input Data Latih: C:\Users\chaca\Documents\Visu... Input Data

Input Nilai k: Input k: 7

Akurasi: SR-R-S-T-ST (selected), SR-R-S, R-S-T, S-T-ST. Keterangan: SR: Sangat Rendah, R: Rendah, S: Sedang, T: Tinggi, ST: Sangat Tinggi.

data uji -	Prediksi Sistem	Prediksi Seben...	Hasil Prediksi
Data Uji-1	Sedang	Sedang	Benar
Data Uji-2	Sedang	Sedang	Benar
Data Uji-3	Tinggi	Sedang	Salah
Data Uji-4	Sedang	Sedang	Benar
Data Uji-5	Sedang	Sedang	Benar
Data Uji-6	Sedang	Sedang	Benar
Data Uji-7	Sedang	Sedang	Benar
Data Uji-8	Sedang	Sedang	Benar
Data Uji-9	Sedang	Sedang	Benar
Data Uji-10	Tinggi	Sedang	Salah
Data Uji-11	Sedang	Sedang	Benar
Data Uji-12	Sedang	Sedang	Benar
Data Uji-13	Sedang	Sedang	Benar
Data Uji-14	Sedang	Sedang	Benar
Data Uji-15	Rendah	Rendah	Benar

Akurasi: 60 % Benar: 18 Salah: 12 Save to Excel Clear

Gambar 4. 8 Antarmuka Pengujian pada tab Akurasi

Keterangan Gambar 4.8:

1. *Datagridview* pada kolom 1 menunjukkan nama untuk setiap data uji.
2. *Datagridview* pada kolom 2 menunjukkan hasil dari prediksi sistem untuk setiap data uji.
3. *Datagridview* pada kolom 3 menampilkan hasil sebenarnya yang dimiliki data uji.
4. *Datagridview* pada kolom 4 menunjukkan hasil dari prediksi sistem dengan hasil sebenarnya. Ketika hasil sesuai, ditampilkan Benar, namun ketika tidak sesuai ditampilkan Salah.
5. *Textbox* untuk menampilkan nilai akurasi sistem.
6. *Textbox* untuk menampilkan berapa data uji yang memiliki hasil prediksi Benar.
7. *Textbox* untuk menampilkan berapa data uji yang memiliki hasil prediksi Salah.
8. Tombol *Save to Excel* untuk menyimpan hasil prediksi ke dalam data excel.
9. Tombol *Clear* untuk mereset semua data pada *form*.

5.4. Sistematika Pengujian

Seperti yang telah dibahas pada bab 3, terdapat dua macam pengujian yang dilakukan. Pengujian yang pertama dilakukan untuk mengetahui pengaruh nilai k dan jumlah data latih terhadap tingkat akurasi. Sedangkan pengujian kedua untuk mengetahui pengaruh data latih dengan kategori mayoritas dan minoritas terhadap tingkat akurasi.

5.4.1. Sistematika Pengujian Pengaruh Nilai k dan Data Latih

Pengujian yang pertama adalah menguji nilai k yang baik untuk prediksi tingkat risiko PJK. Pengujian ini dilakukan pada empat jumlah data latih yang berbeda yaitu data latih 40, 50, 60 dan 70. Sedangkan data yang diujikan bernilai sama yaitu berjumlah 30 data. Kemudian pada setiap data latih dilakukan pengujian sebanyak 9 kali, disesuaikan dengan jumlah nilai k yang ada. Setiap nilai k yang dimasukan, digunakan sebagai parameter pengujian. Setiap data latih, dilakukan pengujian yang sama dengan nilai k , sehingga dari pengujian tersebut dapat diketahui pengaruh nilai k terhadap tingkat akurasi sistem.

5.4.2. Sistematika Pengujian Pengaruh Kelas Mayoritas dan Minoritas

Pengujian yang kedua adalah menguji data latih dengan jumlah berbeda dan dengan kategori yang berbeda juga. Pada pengujian ini dilakukan empat kali pengujian dengan jumlah dan kelas data yang berbeda, namun data uji berjumlah dan memiliki kelas tetap. Untuk pengujian pertama dilakukan pada data latih berjumlah 70 data dengan 5 kelas, yaitu Sangat Rendah, Rendah, Sedang, Tinggi dan Sangat Tinggi. Untuk pengujian kedua, dilakukan pada data latih berjumlah 47 data dengan 3 kelas yaitu Sangat Rendah, Rendah dan Sedang. Untuk pengujian ketiga, dilakukan pada data latih berjumlah 67 data dengan 3 kelas yaitu Rendah, Sedang dan Tinggi. Kemudian untuk pengujian terakhir, dilakukan pada data latih yang berjumlah 58 data dengan 3 kelas yaitu Sedang, Tinggi dan Sangat Tinggi. Pengujian ini dimaksudkan untuk mengetahui nilai akurasi dari pengaruh jumlah kelas mayoritas dan minoritas yang berbeda-beda pada data latih.

5.5. Implementasi Uji Coba

Pada sub bab ini dibahas mengenai implementasi dari uji coba untuk sistem yang sudah dikembangkan sesuai dengan sistematika pengujian yang ada.

5.5.1. Implementasi Pengujian Pengaruh Nilai k dan Pengaruh Jumlah Data Latih

Pada pengujian pengaruh nilai k akan menggunakan nilai k dalam rentang 1–9. Data latih yang digunakan, dimulai dari data 40, 50, 60 dan data 70, sedangkan data uji yang digunakan bernilai sama yaitu berjumlah 30 data. Tabel 4.1 menunjukkan nilai pengujian pengaruh nilai k dengan data latih 40.

Tabel 4. 1 Pengujian Pengaruh nilai k pada 40 data latih

k	Prediksi Benar	Prediksi Salah	Akurasi (%)
k = 1	14 data	16 data	46.67 %
k = 2	16 data	14 data	53.33 %
k = 3	17 data	13 data	56.67 %
k = 4	18 data	12 data	60 %
k = 5	17 data	13 data	56.67 %
k = 6	15 data	15 data	50 %
k = 7	15 data	15 data	50 %
k = 8	15 data	15 data	50 %
k = 9	15 data	15 data	50 %
K=10	15 data	15 data	50 %
K=11	14 data	16 data	46.67 %
K=12	14 data	16 data	46.67 %
K=13	14 data	16 data	46.67 %
K=14	14 data	16 data	46.67 %
K=15	13 data	17 data	43.33 %
K=20	13 data	17 data	43.33 %
K=25	13 data	17 data	43.33 %
K=30	13 data	17 data	43.33 %
K=35	13 data	17 data	43.33 %
K=40	13 data	17 data	43.33 %

Keterangan :

Warna hijau : nilai akurasi (%) tertinggi

Warna merah : nilai akurasi (%) terendah

Pada tabel 4.1 dapat terlihat bahwa nilai akurasi terkecil didapatkan pada nilai $k \geq 15$, dengan nilai akurasi 43.33%, sedangkan untuk nilai akurasi terbesar berada pada nilai $k=4$, dengan nilai akurasi 60%. Hasil dari pengujian data latih 50 ditunjukkan pada tabel 4.2.

Tabel 4. 2 Pengujian Pengaruh nilai k pada 50 data latih

k	Prediksi Benar	Prediksi Salah	Akurasi (%)
k = 1	17 data	13 data	56.67 %
k = 2	18 data	12 data	60 %
k = 3	19 data	11 data	63.33 %
k = 4	17 data	13 data	56.67 %
k = 5	17 data	13 data	56.67 %
k = 6	15 data	15 data	50 %
k = 7	14 data	16 data	46.67 %
k = 8	14 data	16 data	46.67 %
k = 9	14 data	16 data	46.67 %
K=10	14 data	16 data	46.67 %
K=11	14 data	16 data	46.67 %
K=12	14 data	16 data	46.67 %
K=13	14 data	16 data	46.67 %
K=14	14 data	16 data	46.67 %
K=15	14 data	16 data	46.67 %
K=20	14 data	16 data	46.67 %
K=25	14 data	16 data	46.67 %
K=30	14 data	16 data	46.67 %
K=35	14 data	16 data	46.67 %
K=40	14 data	16 data	46.67 %
K=45	14 data	16 data	46.67 %
K=50	14 data	16 data	46.67 %

Keterangan :

Warna hijau : nilai akurasi (%) tertinggi

Warna merah : nilai akurasi (%) terendah

Dari tabel 4.2 dapat diketahui nilai akurasi terkecil berada pada nilai $k \geq 7$ dengan nilai akurasi 46.67%, sedangkan untuk nilai akurasi terbesar berada pada nilai $k=3$ dengan nilai akurasi 63.33%. Hasil dari pengujian data latih 60 ditunjukkan pada tabel 4.3.

Tabel 4. 3 Pengujian Pengaruh nilai k pada 60 data latih

k	Prediksi Benar	Prediksi Salah	Akurasi (%)
k = 1	16 data	14 data	53.33 %
k = 2	16 data	14 data	53.33 %
k = 3	17 data	13 data	56.67 %
k = 4	17 data	13 data	56.67 %
k = 5	18 data	12 data	60 %
k = 6	19 data	11 data	63.33 %
k = 7	19 data	11 data	63.33 %
k = 8	18 data	12 data	60 %
k = 9	20 data	10 data	66.67 %
K=10	18 data	12 data	60 %
K=11	18 data	12 data	60 %
K=12	18 data	12 data	60 %
K=13	18 data	12 data	60 %
K=14	18 data	12 data	60 %
K=15	18 data	12 data	60 %
K=20	18 data	12 data	60 %
K=25	17 data	13 data	56.67 %
K=30	17 data	13 data	56.67 %
K=35	17 data	13 data	56.67 %
K=40	16 data	14 data	53.33 %
K=45	15 data	15 data	50 %
K=50	15 data	15 data	50 %
K=55	15 data	15 data	50 %
K=60	15 data	15 data	50 %

Keterangan :

Warna hijau : nilai akurasi (%) tertinggi

Warna merah : nilai akurasi (%) terendah

Pada tabel 4.3 dapat terlihat bahwa nilai akurasi terkecil berada pada nilai $k \geq 45$, dengan nilai akurasi 50%, sedangkan untuk nilai akurasi terbesar berada pada nilai $k=9$ dengan nilai akurasi 66.67%. Namun, dapat dilihat bahwa setelah k mencapai nilai maksimum pada $k=9$, grafik akurasi terus mengalami penurunan sampai dengan $k=60$. Hasil dari pengujian data latih 70 ditunjukkan pada tabel 4.4.

Tabel 4. 4 Pengujian Pengaruh nilai k pada 70 data latih

k	Prediksi Benar	Prediksi Salah	Akurasi (%)
k = 1	16 data	14 data	53.33 %
k = 2	16 data	14 data	53.33 %
k = 3	17 data	13 data	56.67 %
k = 4	17 data	13 data	56.67 %
k = 5	19 data	11 data	63.33 %
k = 6	20 data	10 data	66.67 %
k = 7	20 data	10 data	66.67 %
k = 8	20 data	10 data	66.67 %
k = 9	19 data	11 data	63.33 %
K=10	19 data	11 data	63.33 %
K=11	18 data	12 data	60 %
K=12	17 data	13 data	56.67 %
K=13	17 data	13 data	56.67 %
K=14	17 data	13 data	56.67 %
K=15	17 data	13 data	56.67 %
K=20	17 data	13 data	56.67 %
K=25	17 data	13 data	56.67 %
K=30	17 data	13 data	56.67 %
K=35	16 data	14 data	53.33 %
K=40	15 data	15 data	50 %
K=45	15 data	15 data	50 %
K=50	15 data	15 data	50 %
K=55	15 data	15 data	50 %
K=60	15 data	15 data	50 %
K=65	15 data	15 data	50 %
K=70	15 data	15 data	50 %

Keterangan :

Warna hijau : nilai akurasi (%) tertinggi

Warna merah : nilai akurasi (%) terendah

Pada tabel 4.4 dapat diketahui nilai akurasi terkecil berada pada nilai $k \geq 40$, dengan nilai akurasi 50%, sedangkan untuk nilai akurasi terbesar berada pada nilai $k=6$ sampai dengan $k=8$, dengan nilai akurasi 66.67%. Dan dapat diketahui grafik akurasi mengalami penurunan setelah mencapai nilai akurasi k tertinggi. Untuk total perbandingan pengaruh jumlah data latih dapat ditunjukkan pada tabel 4.5.

Tabel 4. 5 Pengujian Pengaruh nilai k dan jumlah data latih

Data Latih	K	Akurasi (%)
40 Data Latih	k = 1	46.67 %
	k = 4	60 %
	k = 8	50 %
	k = 12	46.67 %
	k = 15	43.33 %
	k = 20	43.33 %
	k = 25	43.33 %
	k = 30	43.33 %
	k = 40	43.33 %
50 Data Latih	k = 1	56.67 %
	k = 4	56.67 %
	k = 8	46.67 %
	k = 12	46.67 %
	k = 15	46.67 %
	k = 20	46.67 %
	k = 25	46.67 %
	k = 30	46.67 %
	k = 40	46.67 %
60 Data Latih	k = 1	53.33 %
	k = 4	56.67 %
	k = 8	60 %
	k = 12	60 %
	k = 15	60 %
	k = 20	60 %
	k = 25	56.67 %
	k = 30	56.67 %
	k = 40	53.33 %
70 Data Latih	k = 1	53.33 %
	k = 4	56.67 %
	k = 8	66.67 %
	k = 12	56.67 %
	k = 15	56.67 %
	k = 20	56.67 %
	k = 25	56.67 %
	k = 30	56.67 %
	k = 40	50 %

Keterangan :

Warna hijau : nilai akurasi (%) tertinggi

Warna merah : nilai akurasi (%) terendah

Pada tabel 4.5 dapat terlihat bahwa nilai akurasi terkecil berada pada data latih dengan jumlah 40 data, dengan nilai akurasi 43.33%. Sedangkan untuk nilai akurasi terbesar berada data latih yang berjumlah 70 data, dengan nilai akurasi 66.67% pada $k=8$.

5.5.2. Implementasi Pengujian Pengaruh Kelas Mayoritas dan Minoritas

Pada pengujian pengaruh kelas mayoritas dan minoritas dilakukan dengan 4 pengujian berbeda. Untuk pengujian pertama pada data latih yang memiliki 5 kelas, sedangkan untuk pengujian kedua sampai dengan pengujian keempat pada data latih yang memiliki 3 kelas namun dengan kelas yang berbeda-beda. Data latih yang digunakan, dimulai dari 70, 67, 58 dan 47 data latih dengan kelas yang berbeda-beda disetiap 4 pengujian ini. Sedangkan data uji yang digunakan bernilai sama yaitu berjumlah 30 data dengan 5 kelas yaitu sangat rendah, rendah, sedang, tinggi dan sangat tinggi. Tabel 4.6 menunjukkan nilai pengujian pengaruh kelas mayoritas dan minoritas pada 70 data latih dengan 5 kelas (Sangat rendah, rendah, sedang, tinggi dan sangat tinggi).

Tabel 4. 6 Pengujian Pengaruh Kelas Mayoritas dan Minoritas pada 70 Data Latih dengan 5 Kelas.

Kelas Data Latih	k	Benar	Salah	Akurasi (%)
Sangat Rendah, Rendah, Sedang, Tinggi, Sangat Tinggi	k = 1	16	14	53.33 %
	k = 2	16	14	53.33 %
	k = 3	17	13	56.67 %
	k = 4	17	13	56.67 %
	k = 5	19	11	63.33 %
	k = 6	20	10	66.67 %
	k = 7	20	10	66.67 %
	k = 8	20	10	66.67 %
	k = 9	19	11	63.33 %

Keterangan :

Warna hijau : nilai akurasi (%) tertinggi

Warna merah : nilai akurasi (%) terendah

Pada tabel 4.6 dapat terlihat bahwa nilai akurasi terkecil berada pada data latih dengan nilai $k=1$ dan $k=2$, dengan nilai akurasi 53.33%, sedangkan untuk nilai akurasi terbesar berada pada data latih dengan nilai $k=6$, $k=7$ dan $k=8$, dengan nilai akurasi 66.67%. Hasil dari pengujian pengaruh kelas mayoritas dan minoritas pada 47 data latih dengan 3 kelas (Sangat rendah, rendah dan sedang) ditunjukkan pada tabel 4.7.

Tabel 4. 7 Pengujian Pengaruh Kelas Mayoritas dan Minoritas pada 47 Data Latih dengan 3 Kelas.

Kelas Data Latih	k	Benar	Salah	Akurasi (%)
Sangat Rendah, Rendah, Sedang	k = 1	15	15	50 %
	k = 2	14	16	46.67 %
	k = 3	15	15	50 %
	k = 4	16	14	53.33 %
	k = 5	16	14	53.33 %
	k = 6	16	14	53.33 %
	k = 7	16	14	53.33 %
	k = 8	15	15	50 %
	k = 9	15	15	50 %

Keterangan :

Warna hijau : nilai akurasi (%) tertinggi

Warna merah : nilai akurasi (%) terendah

Pada tabel 4.7 dapat terlihat bahwa nilai akurasi menurun dibanding dengan pengujian sebelumnya nilai. Dimana nilai akurasi terkecil berada pada data latih dengan nilai $k=2$ dimana memiliki akurasi 46.67%, sedangkan untuk nilai akurasi terbesar berada data latih dengan nilai $k=4$ sampai dengan $k=7$, dengan nilai akurasi 53.33%.

Hasil dari pengujian pengaruh kelas mayoritas dan minoritas pada 67 data latih dengan 3 kelas (Rendah, sedang dan tinggi) ditunjukkan pada tabel 4.8.

Tabel 4. 8 Pengujian Pengaruh Kelas Mayoritas dan Minoritas pada 67 Data Latih dengan 3 Kelas

Kelas Data Latih	k	Benar	Salah	Akurasi (%)
Rendah, Sedang, Tinggi	k = 1	15	14	53.33 %
	k = 2	18	12	60 %
	k = 3	15	15	50 %
	k = 4	15	14	53.33 %
	k = 5	17	13	56.67 %
	k = 6	17	13	56.67 %
	k = 7	17	13	56.67 %
	k = 8	17	13	56.67 %
	k = 9	16	14	53.33 %

Keterangan :

Warna hijau : nilai akurasi (%) tertinggi

Warna merah : nilai akurasi (%) terendah

Pada tabel 4.8 dapat terlihat bahwa nilai akurasi terkecil berada pada data latih dengan nilai k=3 dengan nilai akurasi 50%, sedangkan untuk nilai akurasi terbesar berada data latih dengan nilai k=2, dengan nilai akurasi 60%. Hasil dari pengujian pengaruh kelas mayoritas dan minoritas pada 58 data latih dengan 3 kelas (Sedang, tinggi dan sangat tinggi) ditunjukkan pada tabel 4.9.

Tabel 4. 9 Pengujian Pengaruh Kelas Mayoritas dan Minoritas pada 58 Data Latih dengan 3 Kelas

Kelas Data Latih	k	Benar	Salah	Akurasi (%)
Sedang, Tinggi, Sangat Tinggi	k = 1	14	16	46.67 %
	k = 2	15	15	50 %
	k = 3	15	15	50 %
	k = 4	17	13	56.67 %
	k = 5	16	14	53.33 %
	k = 6	15	15	50 %
	k = 7	15	15	50 %
	k = 8	16	14	53.33 %
	k = 9	16	14	53.33 %

Keterangan :

Warna hijau : nilai akurasi (%) tertinggi

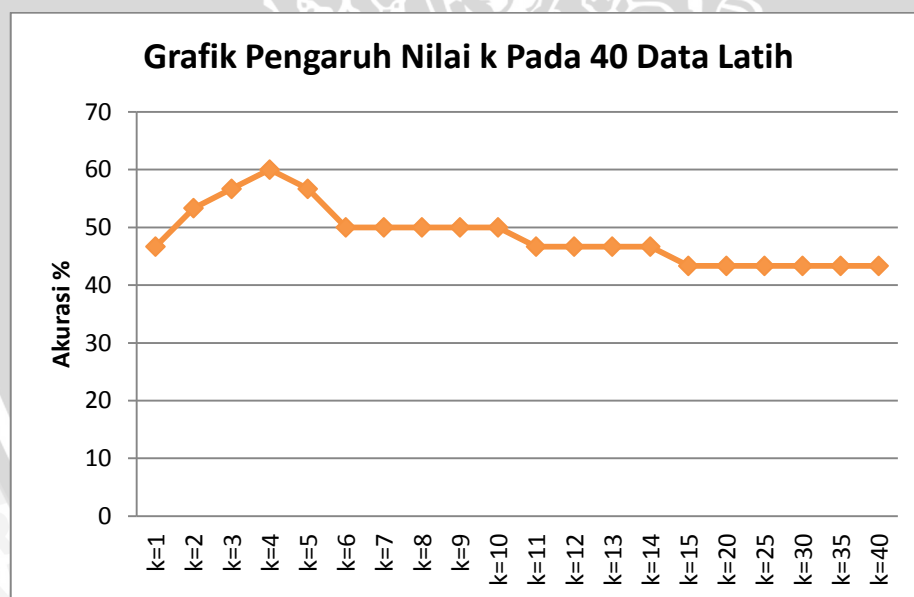
Warna merah : nilai akurasi (%) terendah

Pada tabel 4.9 dapat terlihat bahwa nilai akurasi terkecil berada pada data latih dengan nilai $k=1$, dengan nilai akurasi 46.67%, sedangkan untuk nilai akurasi terbesar berada data latih dengan nilai $k=4$, dengan nilai akurasi 56.67%.

5.6. Analisa Hasil

5.6.1. Analisa Hasil Pengaruh Nilai k dan Pengaruh Jumlah Data Latih

Pada hasil uji coba pertama menggunakan 40 data latih yang terlihat pada tabel 4.1 bahwa nilai akurasi yang dihasilkan tiap-tiap nilai k berbeda. Untuk nilai akurasi terkecil berada pada nilai $k=4$ dan $k=9$, sedangkan untuk nilai akurasi terbesar berada pada nilai $k=2$. Berdasarkan tabel 4.1 dapat dibuat grafik hubungan antara nilai k dengan nilai akurasi. Grafik nilai ini ditunjukkan pada gambar 4.9.

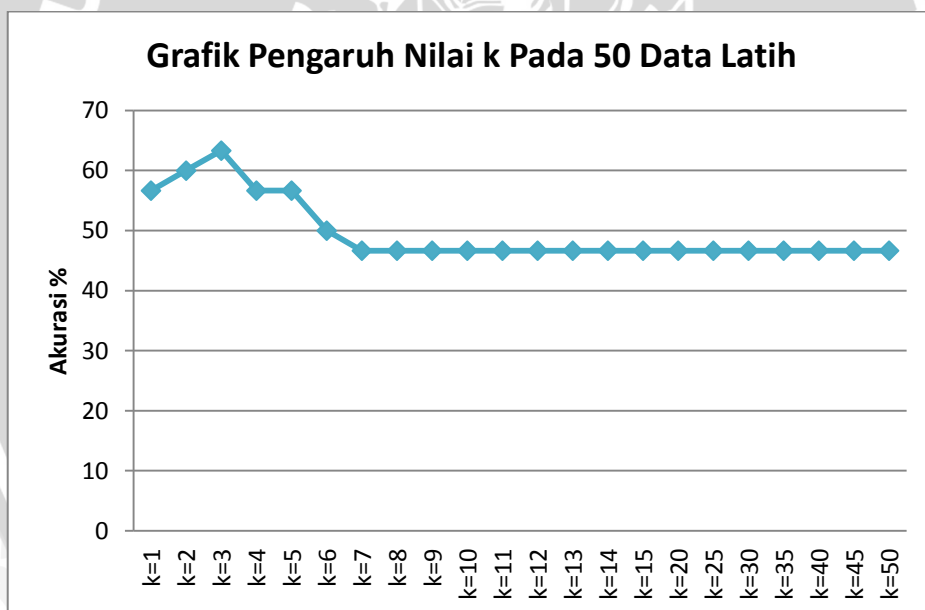


Gambar 4. 9 Grafik Pengaruh nilai k terhadap 40 data latih

Pada gambar grafik 4.9 ditunjukkan bahwa mulai dari nilai $k=1$ sampai dengan $k=2$ nilai akurasi naik, namun pada nilai $k=3$ nilai mengalami penurunan akurasi sampai dengan $k=4$, namun pada $k=5$ nilai akurasi kembali naik lagi dan konstan sampai $k=8$, dan kemudian turun lagi pada $k=9$. Ketika data latih yang digunakan 40 data nilai k memiliki akurasi yang bagus pada $k=1$ dan $k=2$, namun

setelah itu nilai akurasi rendah. Nilai k pada $k=1$ dan $k=2$ naik di karenakan kelas tetangga terdekat masih 1 atau 2 kelas dengan nilai keanggotaan kelas yang tertinggi. Sehingga nilai kelas terdekat tersebut hanya dari kelas tersebut saja namun pada nilai k ini, factor *noise* masih sangat besar. Pada $k=3$ dan berikutnya grafik mulai menurun, dapat dikarenakan pengaruh kelas-kelas terdekat berikutnya yang masuk dalam pengelompokan nilai k . Dimana perhitungan keanggotaan vektor setiap kelas prediksi dengan semakin banyaknya kelas dapat mempengaruhi naik turunnya akurasi sistem.

Pada hasil uji coba kedua menggunakan 50 data latih yang terlihat pada tabel 4.2 bahwa nilai akurasi terkecil berada pada nilai $k=4$, sedangkan nilai akurasi terbesar berada pada nilai $k=1$ dan $k=2$. Berdasarkan tabel 4.2 dapat dibuat grafik hubungan antara nilai k dengan nilai akurasi. Grafik nilai ini ditunjukkan pada gambar grafik 4.10.

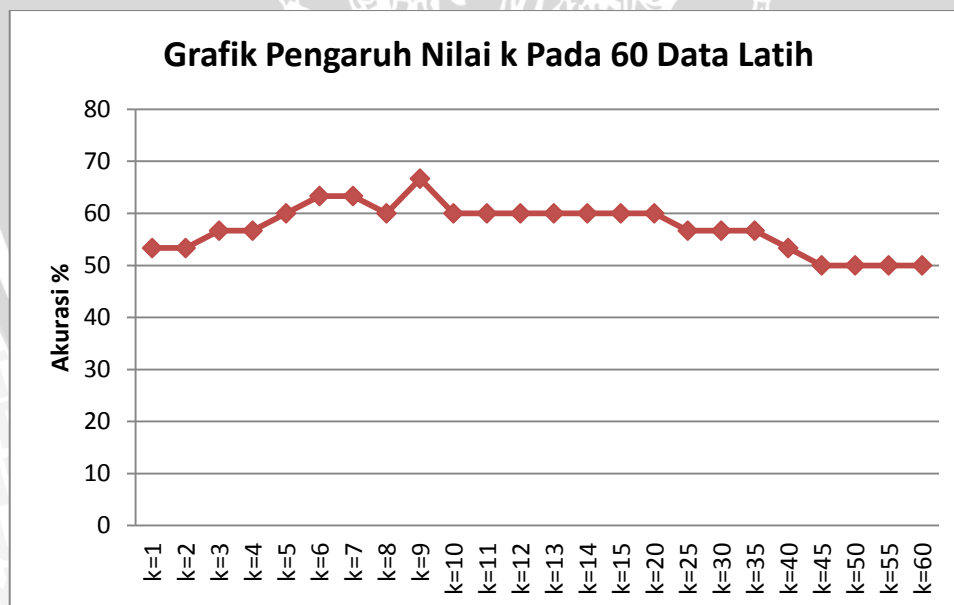


Gambar 4. 10 Grafik Pengaruh 50 Data Latih terhadap nilai k

Pada gambar grafik 4.10 ditunjukkan bahwa grafik turun mulai nilai $k=3$ sampai dengan $k=4$. Bertambahnya nilai k sangat mempengaruhi keanggotaan kelas terdekat, dimana semakin bertambah nilai k , tingkat *noise* semakin kecil namun kemungkinan kelas prediksi akan semakin banyak, sehingga sensitifitas akan semakin berkurang. Namun pada nilai $k=5$ grafik mulai naik kembali dan

selanjutnya konstan sampai dengan $k=9$. Pada grafik dengan data latih 50 lebih memiliki akurasi yang lebih bagus dibanding dengan data latih 40. Pada data latih 50, k awal terletak pada rentang nilai akurasi 60, namun mulai $k=4$ grafik tertinggi hanya mencapai rentang 50 – 60, dimana pengelompokan kelas tetangga terdekat mulai bertambah dan keanggotaan vektor kelas prediksi sesuai dengan kelas tetangga tersebut dengan semakin banyaknya kelas juga mempengaruhi. Tingkat *noise* pada sistem merupakan pengaruh data yang terlalu besar, misalnya data latih n memiliki kategori kelas Rendah, namun dilakukan pengujian dengan nilai $k = 1$ dan ditemukan kelas terdekat j dengan kelas Tinggi. Hal ini yang merupakan pengaruh *noise* sangat besar, tanpa mengetahui pengaruh atribut yang lainnya juga.

Pada hasil uji coba kedua menggunakan 60 data latih yang terlihat pada tabel 4.3 bahwa nilai akurasi terkecil berada pada nilai $k=3$ dan $k=4$, sedangkan nilai akurasi tertinggi dicapai saat nilai $k=5$ sampai dengan nilai $k=7$ dan $k=9$. Berdasarkan tabel 4.3 dapat dibuat grafik hubungan antara nilai k dengan nilai akurasi. Grafik nilai ini ditunjukkan pada gambar grafik 4.11.

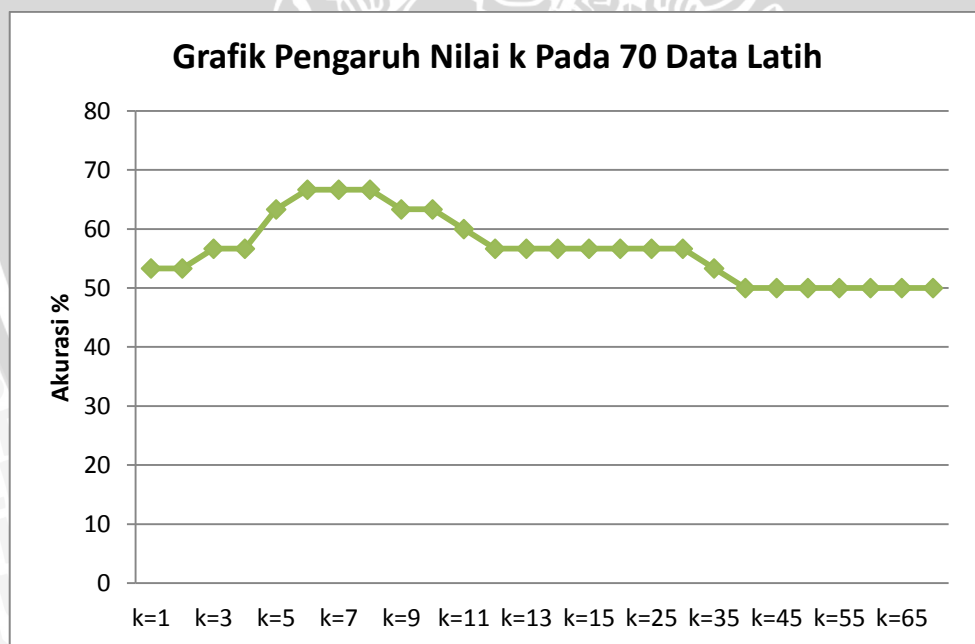


Gambar 4. 11 Grafik Pengaruh 60 Data Latih terhadap nilai k

Pada gambar grafik 4.11 ditunjukkan bahwa nilai $k=1$ sampai dengan $k=2$ nilai konstan, namun setelah itu terjadi penurunan nilai akurasi, setelah itu nilai

akurasi naik lagi pada nilai $k=4$ kemudian konstan dan ketika $k=8$ mengalami penurunan nilai dan kemudian naik lagi pada nilai $k=9$. Pada grafik dengan data latih 60, nilai $k=1$ ditunjukkan dengan grafik naik dimana kelas keanggotaan kelas masih hanya kelas tetangga terdekat itu saja. Pada $k=2$ dan $k=3$ terlihat sama dimana kelas keanggotaan vektor kelas terdekat memberikan akurasi yang sama. Selanjutnya untuk $k=4$ sampai dengan $k=6$ grafik terus mengalami kenaikan, ini karena semakin banyak k semakin banyak juga keanggotaan kelas yang mendekati kelas prediksi. Namun pada $k=7$ sampai dengan $k=9$ grafik mengalami penurunan, dikarenakan pada $k=6$ sudah mencapai titik optimal, sehingga untuk k berikutnya tingkat akurasi menurun.

Pada hasil uji coba kedua menggunakan 70 data latih yang terlihat pada tabel 4.4 bahwa nilai akurasi terkecil berada pada nilai $k=1$, sedangkan nilai akurasi tertinggi dicapai saat nilai $k=6$ yaitu 66.67%. Berdasarkan tabel 4.4 dapat dibuat grafik hubungan antara nilai k dengan nilai akurasi. Grafik nilai ini ditunjukkan pada gambar grafik 4.12.



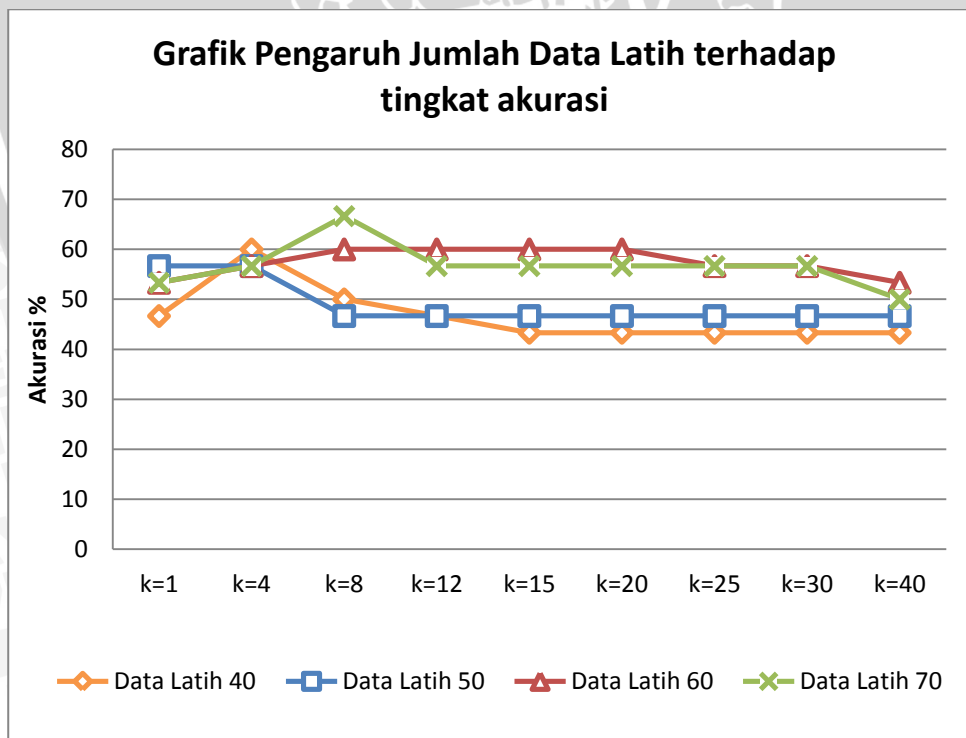
Gambar 4. 12 Grafik Pengaruh 70 Data Latih terhadap nilai k

Pada gambar grafik 4.12 ditunjukkan bahwa mulai nilai $k=1$ terjadi kenaikan laju akurasi sampai dengan $k=6$ kemudian setelah itu terjadi penurunan nilai akurasi. Pada data latih 70, semakin terlihat akurasi data PJK yang dihasilkan

menunjukkan hasil grafik yang sama dengan akurasi yang dihasilkan pada data latih 60, dimana nilai akurasi pada data latih 70 akurasi tertinggi pada $k=6$. Namun pada grafik ini akurasi yang dihasilkan lebih bagus dibandingkan dengan akurasi pada data latih 60.

Dapat diartikan bahwa, jumlah data latih yang semakin bertambah sangat mempengaruhi meningkatnya akurasi. Pengaruh k juga mempengaruhi hasil akurasi, dimana pada $k=1$ sampai dengan $k=6$ ditunjukkan grafik naik sampai dengan k optimalnya yaitu $k=6$, dimana semakin banyaknya nilai k maka semakin memperkecil *noise* yang ada sehingga nilai akurasi semakin bagus. Namun setelah $k=6$ grafik menurun dikarenakan keanggotaan kelas yang semakin banyak memberikan nilai sensitifitas yang semakin besar pada keanggotaan *vektor*.

Untuk hasil uji coba berikutnya, seperti yang terlihat pada tabel 4.5 pada empat jumlah data latih yang berbeda yaitu 40, 50, 60 dan 70 data latih. Dimana nilai akurasi terendah ditunjukkan oleh grafik dengan 40 data latih dan nilai akurasi tertinggi ditunjukkan oleh grafik dengan 70 data latih. Berdasarkan tabel 4.5 dapat dibuat grafik hubungan antara nilai k , data latih dengan nilai akurasi. Grafik nilai ini ditunjukkan pada gambar grafik 4.13.

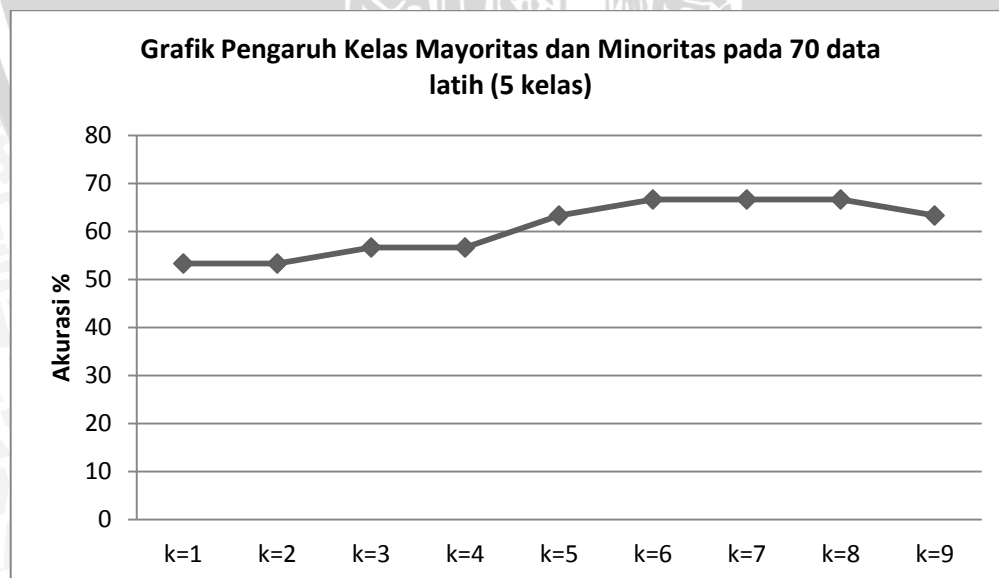


Gambar 4. 13 Grafik Pengaruh Jumlah Data Latih

Pada gambar grafik 4.13 ditunjukkan bahwa data latih 70 dengan warna hijau lebih memiliki nilai akurasi yang tinggi dibanding dengan data latih 40, 50, 60 dan 70. Dari grafik ini dapat diketahui bahwa, semakin banyak data latih akan didapatkan nilai akurasi yang semakin tinggi juga. Karena dengan semakin banyaknya data latih, semakin banyaknya data yang memungkinkan mendukung keanggotaan kelas prediksi. Jika pada data 40, kemungkinan kelas prediksi yang sesuai dengan data uji hanya berdasarkan keanggotaan kelas terdekat dari 40 data tersebut. Namun dengan bertambahnya data latih, hingga data latih 70 tersebut kemungkinan kelas prediksi dari keanggotaan kelas terdekat dari 70 data latih tersebut.

5.6.2. Analisa Hasil Pengaruh Kelas Mayoritas dan Minoritas

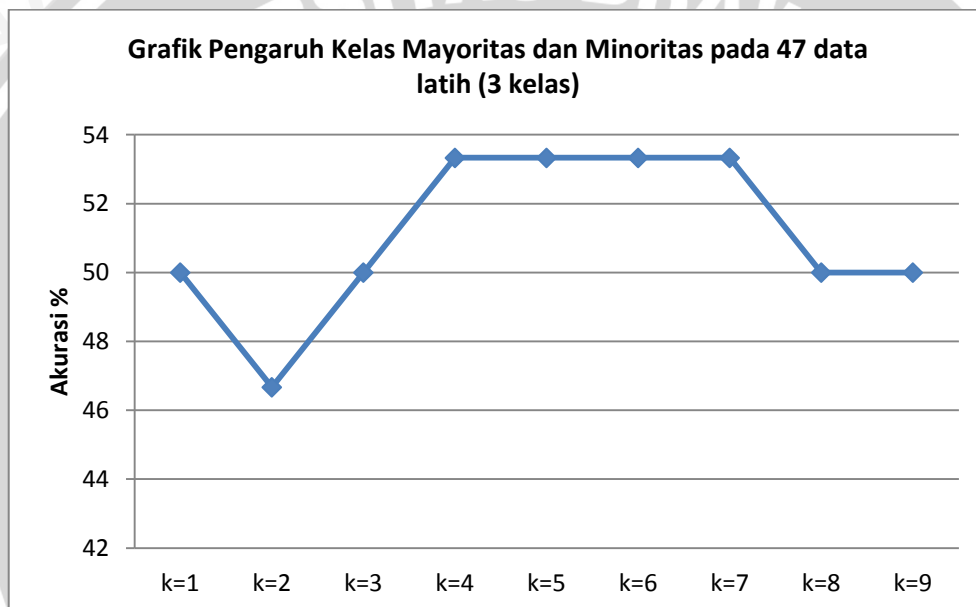
Pada hasil uji coba pertama menggunakan 70 data latih dengan 5 kelas (Sangat rendah, rendah, sedang, tinggi, sangat tinggi), seperti yang terlihat pada tabel 4.6. Dapat diketahui bahwa nilai akurasi yang dihasilkan tiap nilai k berbeda. Untuk nilai akurasi terkecil berada pada nilai $k=1$, sedangkan untuk nilai akurasi terbesar berada pada nilai $k=6$, $k=7$ dan $k=8$. Berdasarkan tabel 4.6 dapat dibuat grafik hubungan antara nilai k dengan nilai akurasi. Grafik nilai ini ditunjukkan pada gambar 4.14.



Gambar 4. 14 Grafik Pengaruh kelas mayoritas dan minoritas pada 70 data latih (5 kelas)

Pada gambar grafik 4.14 ditunjukkan bahwa akurasi mengalami kenaikan mulai dari $k=1$ sampai dengan $k=8$, dan mencapai puncak nilai tertinggi pada $k=6$ sampai dengan $k=8$. Namun setelah itu laju akurasi mengalami penurunan. Grafik dengan 70 data latih yang memiliki 5 kelas ini sama dengan gambar 4.11.

Untuk hasil uji coba berikutnya menggunakan 47 data latih dengan 3 kelas (Sangat rendah, rendah, sedang) dan dengan data uji yang sama dengan pengujian sebelumnya, seperti yang terlihat pada tabel 4.7. Untuk nilai akurasi terkecil berada pada nilai $k=2$, sedangkan untuk nilai akurasi terbesar berada pada nilai $k=4$ sampai dengan $k=7$. Grafik nilai ini ditunjukkan pada gambar 4.15.

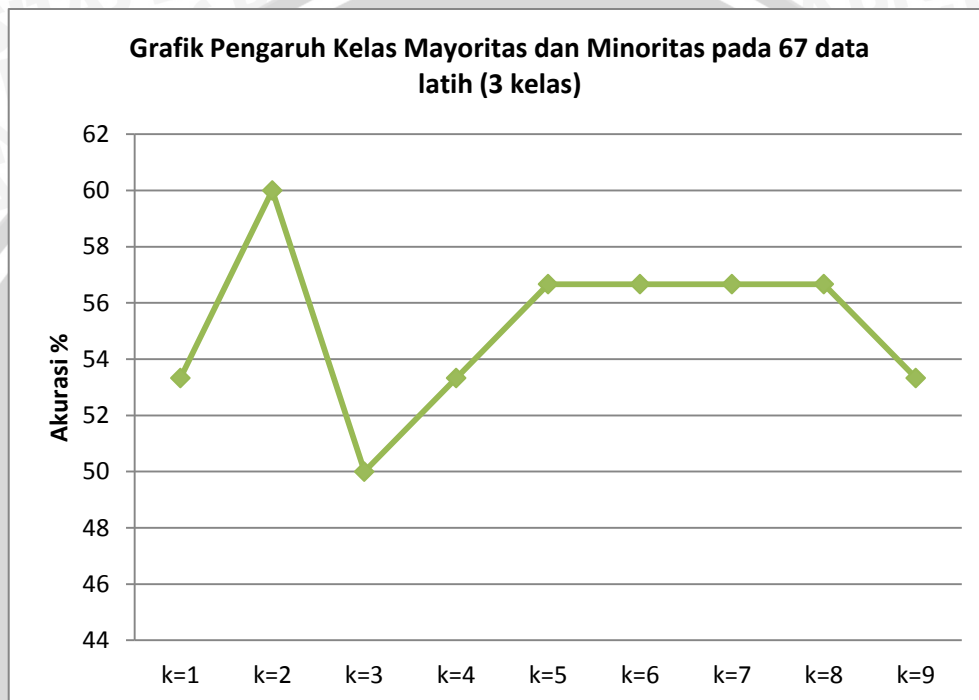


Gambar 4. 15 Grafik Pengaruh kelas mayoritas dan minoritas pada 47 data latih (3 kelas)

Pada gambar grafik 4.15 ditunjukkan bahwa grafik akurasi mengalami naik-turun, dimana nilai awalnya turun ketika $k=2$ dan kemudian naik sampai dengan $k=4$ dan kemudian konstan sampai $k=7$, dan kemudian turun sampai dengan $k=9$. Menurunnya kelas pada data mempengaruhi akurasi data. Dimana pada grafik ini ditunjukkan bahwa akurasi tidak mencapai 54%. Pada pengujian ini, hanya kelas sangat rendah, rendah dan sedang. Jumlah setiap data juga sangat berbeda, dimana data latih dengan kelas sedang sangat mendominasi data. Sehingga prediksi kelas

terdekat sangat terpengaruh pada *noise*, dimana terdapat data yang terlalu mendominasi data latih.

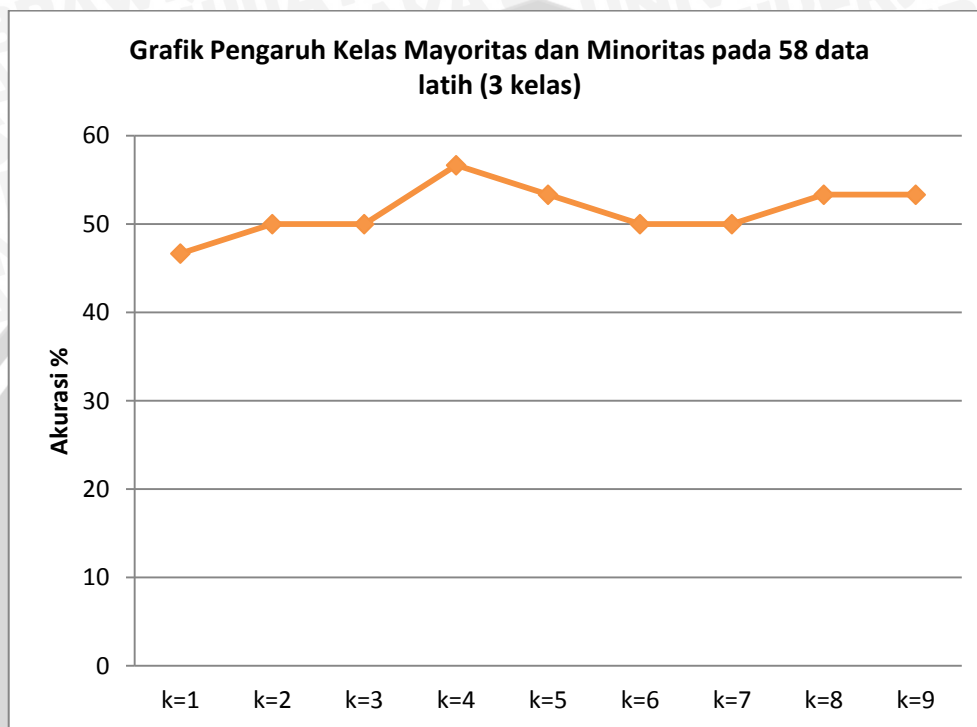
Untuk hasil uji coba berikutnya menggunakan 67 data latih dengan 3 kelas (Rendah, sedang, tinggi), seperti yang terlihat pada tabel 4.8. Diketahui bahwa nilai akurasi terkecil berada pada nilai $k=3$, sedangkan untuk nilai akurasi tertinggi berada pada nilai $k=2$. Grafik nilai ini ditunjukkan pada gambar 4.16.



Gambar 4. 16 Grafik Pengaruh kelas mayoritas dan minoritas pada 67 data latih (3 kelas)

Pada gambar grafik 4.16 ditunjukkan bahwa grafik akurasi mengalami kenaikan mulai dari $k=1$ dan mencapai nilai tertinggi pada $k=2$. Namun setelah itu grafik menunjukkan arah turun-naik di setiap nilai k tertentu. Pada pengujian ini penyebaran data lebih agak merata, walaupun masih terdapat data dengan kelas yang terlalu mendominasi. Dapat diketahui bahwa penyebaran data juga mempengaruhi meningkatnya akurasi. Jika pada data latih, terdapat data yang terlalu mendo-minasi maka akurasi akan semakin menurun, dikarenakan kedekatan *vektor* kelas akan terpengaruh dengan data yang terlalu dominan dengan jumlah yang lebih banyak.

Untuk hasil uji coba berikutnya menggunakan 58 data latih dengan 3 kelas (Sedang, tinggi, sangat tinggi), seperti yang terlihat pada tabel 4.9. Diketahui bahwa nilai akurasi terkecil berada pada nilai $k=1$, sedangkan untuk nilai akurasi tertinggi berada pada nilai $k=4$. Grafik nilai ini ditunjukkan pada gambar 4.17.



Gambar 4. 17 Grafik Pengaruh kelas mayoritas dan minoritas pada 58 data latih (3 kelas)

Pada gambar grafik 4.17 ditunjukkan bahwa grafik akurasi mengalami kenaikan mulai dari $k=1$ dan mencapai nilai tertinggi pada $k=4$. Setelah itu grafik menunjukkan penurunan nilai sampai dengan $k=8$, kemudian naik sedikit pada nilai $k=9$. Sama halnya dengan kelas sangat rendah, rendah dan sedang, pada pengujian dengan data latih sedang, tinggi dan sangat tinggi, data dengan kelas sedang lebih mendominasi dibandingkan dengan data kelas tinggi dan sangat tinggi. Dimana jumlah data sedang terlalu banyak, sehingga kedekatan kelas yang di ujikan lebih mendekati keanggotaan vektor data latih yang memiliki jumlah lebih banyak.

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Berdasarkan hasil penelitian tentang prediksi tingkat risiko penyakit jantung koroner (PJK) dengan menggunakan *Fuzzy K-Nearest Neighbor* (FK-NN), dapat disimpulkan bahwa:

1. Metode *Fuzzy K-Nearest Neighbor* (FK-NN) telah berhasil digunakan untuk prediksi tingkat risiko penyakit jantung koroner (pjk). Dengan menggunakan atribut umur, kadar LDL, kolesterol, kadar HDL, trigliserida dan sistolik.
2. Nilai k yang digunakan mulai dari k=1, k=2, sampai dengan k=9. Dimana untuk implementasi sistem PJK ini tingkat akurasi tertinggi didapatkan pada nilai k=6 sampai dengan k=8 pada data latih 70 dengan prosentase nilai 66.67%. Nilai k kurang dari 6 maupun nilai k lebih dari 6 menunjukkan bahwa hasil akurasi kurang baik, karena sudah mencapai nilai optimalnya. Dapat diketahui bahwa nilai k yang terlalu kecil menyebabkan klasifikasi terlalu terpengaruh kepada kelas keanggotaan terdekat saja. Sedangkan pemilihan nilai k yang terlalu tinggi juga akan menyebabkan akurasi rendah karena semakin banyak data yang tidak relevan (*noise*), sehingga untuk mendapatkan hasil akhir sistem, hanya akan diambil keanggotaan vektor dengan nilai yang terbesar saja.
3. Atribut dengan sedikit nilai beberapa atribut atau yang disebut dengan atribut minoritas, akan mempengaruhi akurasi dikarenakan nilai terlalu dipengaruhi oleh atribut mayoritas yang menyebabkan prediksi sistem tidak sesuai dengan prediki sebenarnya.
4. Tingkat akurasi tertinggi yang dapat dihasilkan oleh sistem adalah sebesar 66,67% pada data latih 70 dan data latih 60. Penambahan data latih mempengaruhi peningkatan akurasi sistem, karena dengan semakin banyaknya data latih, maka kemungkinan semakin banyaknya jarak *record* yang mendekati kelas data prediksi adalah semakin tinggi. Dimana

jumlah data latih yang digunakan untuk proses pengujian adalah 40, 50, 60 dan 70 data latih.

5.2. Saran

Beberapa saran untuk pengembangan lebih lanjut yang dapat diberikan oleh penulis adalah:

1. Melakukan prediksi tingkat risiko PJK dengan data latih yang lebih banyak dan penyebaran *linguistic* kelas *membership* yang merata untuk mengetahui apakah sistem memiliki akurasi yang lebih baik atau tidak.
2. Pada penelitian selanjutnya dapat dicari nilai k yang optimal untuk mengetahui nilai k pada sistem yang memiliki akurasi yang paling baik. Salah satu metode yang dapat digunakan untuk mencari nilai k optimal yaitu dengan menggunakan metode *k-fold cross validation*.
3. Digunakannya metode pencarian jarak selain *Euclidean Distance* seperti metode *cityblock*, *cosine similarity* dan *correlation* untuk mengetahui tingkat akurasi yang dihasilkan sistem.

DAFTAR PUSTAKA

- [ABD-07] Abdul, Majid. 2007. *Penyakit Jantung Koroner: Patofisiologi, Pencegahan dan Pengobatan Terkini*. Universitas Sumatera Utara. Sumatera Utara
- [ANW-04] Anwar, T. Bahri. 2004. *Faktor Risiko Penyakit Jantung Koroner*. Universitas Sumatera Utara. Sumatera Utara
- [ALB-95] Albert T. Zebua dan Wahidin Wahab. 1995. *Teknologi Sistem Fuzzy*. <http://www.elektroindonesia.com/elektro/no6b.html>. Tanggal akses: 21 Maret 2012
- [COX-05] Cox E. 2005. *Fuzzy Modeling and Genetic Algorithms for Data Mining dan Exploration*. USA: Morgan Kaufman Publishers
- [DIR-06] Direktorat Bina Farmasi. 2006. *Pharmaceutical Care untuk Pasien Penyakit Jantung Koroner: Fokus sindrom akut*. Jakarta
- [EFF-08] Effendy, Nazrul, Subagja, dan Amir Faisal. 2008. *Prediksi Penyakit Jantung Koroner (PJK) Berdasarkan Faktor Risiko Menggunakan Jaringan Syaraf Tiruan Backpropagation*. Seminar Nasional Aplikasi Teknologi Informasi. Yogyakarta.
- [HAN-00] Han, Jiawei dan Kamber, Micheline. 2000. *Data Mining: Concepts and Techniques*. San Fransisco: Morgan Kaufmann Publishers..
- [HAN-01] Han, Jiawei dan Micheline Khamber. 2001. *Data Mining: Concepts and Technique*. Morgan Kaufmann Publisher: San Francisco, USA.

- [HAY-99] Hayes, Daniel M.D. 1999. *Distress sudden exercise raise heart attack risk*, American Heart Association.
- [JIN-04] Jin, Weiqing. 2004. *Fuzzy Classification Based on Fuzzy Association Rule Mining*. North Carolina State University.
- [JKI-86] J.Kittler, "Feature Selection dan Extraction", in *Handbook of Pattern Recognition and Image Processing*, Tzay Y. Young, King Sun Fu Ed. Academic Press, 1986
- [JNI-96] J.Nilsson, Nill. "Introduction To Machine Learning". 1996. Stanford University: Stanford, CA 94305
- [KEL-85] Keller, M. James, Michael R Gray, James A. Givens. 1985. *A Fuzzy K-Nearest Neighbor*. IEEE Transactions On Sistem, Man And Cybernetics, Vol. SMC-15 NO 4
- [KRI-02] Krismi, Arum. 2002. *Perbedaan Faktor Risiko Primer Pada Penderita Penyakit Jantung Koroner dan Kontrol di RSUP Dr. Sardjito Yogyakarta*. Skripsi. Yogyakarta.
- [KUS-07] Kusnawi. 2007. *Pengantar Solusi Data Mining* (Online), (<http://p3m.amikom.ac.id/p3m/56%20%20PENGANTAR%20SOLUSI%20DATA%20MINING.pdf>, diakses tanggal 8 Maret 2012).
- [KUS-09] Kusrini dan Luthfi, Emha Tufiq. 2009. *Algoritma Data Mining*. Yogyakarta: Penerbit ANDI.
- [LAR-05] Larose, Daniel T. 2005. *Discovering Knowledge in Data. An Introduction to Data mining*. John willey dan Sons. New Jersey

- [MOR-09] Moradian, M. dan Baarani, A. 2009. KNNBA: k-Nearest-Neighbor-Based-Association Algorithm.
- [NUG-06] Nugraha, Dany dan Ramdhany. 2006. *Diagnosis Gangguan Sistem Urinari pada Anjing dan Kucing menggunakan VFI 5*. IPB. Bandung
- [PRA-12] Prasetyo, Eko. 2012. *Fuzzy K-Nearest Neighbor In Every Class Untuk Klasifikasi Data*. Seminar Nasional Teknik Informatika (SANTIKA 2012). Universitas Pembangunan Nasional Veteran Jawa Timur
- [PRI-94] Price, Sylvia A. dan Lorraine M. 1994. *PATOFISIOLOGI. Konsep Klinis proses-proses penyakit*. Kedokteran EGC. Jakarta
- [PRI-05] Priyono, Agus Priyono, Muhammad Ridwan, Ahmad Jais Alias, Riza Atiq, O. K. Rahmat, Azmi Hassan & Mohd. Alauddin Mohd. Ali. 2005. *Generation Of Fuzzy Rules With Subtractive Clustering*, Jurnal Teknologi, 43(D) Dis. Universitas Teknologi Malaysia.
- [SOE-04] Soeharto, Imam. 2004. *Serangan Jantung dan Stroke Hubungannya dengan Lemak dan Kolesterol*. PT Gramedia Pustaka Utama. Jakarta
- [SOE-10] Soeharto, Imam. 2010. *Penyakit Jantung Koroner (PJK): Sebab, Mekanisme dan Gejala*. <http://fkunhas.com/penyakit-jantung-koroner-pjk-sebab-mekanisme-dan-gejala-20100716347.html>. Tanggal akses: 1 Maret 2012.
- [SRI-10] Sri, Kusumadewi, Purnomo, Hari. 2010. *Aplikasi Logika Fuzzy untuk Pendukung Keputusan*. Jakarta. Penerbit Graha Ilmu
- [SUD-08] Sudrajat. 2008. *Jurnal : Dasar-dasar Fuzzy Logic*. Bandung.

- [SUP-08] Supriyono, Mamat. 2008. *Faktor-faktor Risiko Yang Berpengaruh Terhadap Kejadian Penyakit Jantung Koroner Pada Kelompok Usia <45 Tahun*. <http://eprints.undip.ac.id/18090/>. Tanggal akses: 24 Februari 2012
- [TUR-05] Turban, 2005, *Decision Support Systems and Intelligent Systems (Sistem Pendukung Keputusan dan Sistem Cerdas)* jilid 1. Andi Offset: Yogyakarta.
- [TJO-01] Tjokprawiro, Askandar. 2001. *Diabetes Melitus: capita Selecta 2001-B (Clinical Experiences and Recent Advances)*. Yogyakarta Diabetes Update. Yogyakarta
- [WAH-11] Wahyuni, Kristin. 2011. *Diagnosis Penyakit Jantung Koroner (PJK) Berdasarkan Faktor Risiko Menggunakan Metode FES*. Skripsi. Malang.
- [ZHA-09] Zhang, Juan, Yi Niu, Huabei Nie. 2009. "Web Document Classification Based on Fuzzy KNN Algorithm". International Conference on Computational Intelligence and Security.

LAMPIRAN

Lampiran 1 Data Pelatihan 70

No	umur	ldl	kol_total	hdl	trigliserida	sistolik	nilai_risiko	Kategori
1	57	115	218	66	274	150	20	sedang
2	60	171	182	57	286	140	20	sedang
3	58	134	223	25	130	150	20	sedang
4	61	167	231	39	254	130	21	sedang
5	77	148	208	33	134	180	22	sedang
6	77	186	168	66	177	130	22	sedang
7	41	185	208	50	236	130	22	sedang
8	38	119	289	41	232	140	22	sedang
9	63	186	197	40	319	120	22	Sedang
10	60	150	218	32	284	140	23	Sedang
11	51	196	200	72	159	140	23	Sedang
12	47	192	171	41	350	120	23	Sedang
13	67	191	196	53	188	140	23	Sedang
14	59	176	245	44	233	150	24	Sedang
15	50	166	236	35	284	130	24	Sedang
16	65	137	214	13	302	140	25	Sedang
17	62	178	215	35	258	120	25	Sedang
18	68	256	221	48	190	140	25	sedang
19	64	128	214	44	193	180	25	sedang
20	71	105	218	51	203	160	25	sedang
21	82	90	164	40	169	170	25	sedang
22	72	108	213	48	304	180	25	sedang
23	68	123	232	45	292	170	25	sedang
24	49	146	221	44	162	170	26	sedang
25	67	209	194	48	112	160	26	sedang
26	66	151	229	60	272	180	27	sedang
27	38	195	270	35	166	130	27	sedang
28	63	151	224	50	162	130	28	sedang
29	56	198	173	38	131	200	28	sedang
30	61	98	256	33	269	150	29	sedang
31	65	158	229	30	272	150	29	sedang
32	55	154	253	41	167	150	29	sedang
33	51	104	390	53	176	150	29	sedang
34	43	195	181	60	154	170	29	sedang
35	63	299	159	47	152	160	30	sedang
36	73	170	149	44	144	140	10	rendah

37	55	190	160	39	123	120	14	rendah
38	60	116	170	40	129	180	15	rendah
39	70	264	226	44	111	140	15	rendah
40	56	145	187	41	278	130	17	rendah
41	66	148	130	48	270	140	17	rendah
42	59	149	172	52	129	180	18	rendah
43	67	94	182	75	149	160	19	rendah
44	58	105	170	34	223	140	19	rendah
45	51	79	189	34	399	140	19	rendah
46	68	140	220	37	205	130	19	rendah
47	61	90	155	90	101	130	5	sangat rendah
48	35	380	350	30	280	180	42	sangat tinggi
49	70	254	280	30	165	160	42	sangat tinggi
50	50	164	243	39	193	160	30	tinggi
51	74	283	145	32	134	160	30	tinggi
52	67	167	255	39	220	140	31	tinggi
53	58	160	239	38	168	180	31	tinggi
54	60	148	266	39	292	160	32	tinggi
55	63	152	239	45	180	160	32	tinggi
56	54	178	262	54	173	160	33	tinggi
57	65	183	248	36	196	140	33	tinggi
58	62	158	269	59	162	200	33	tinggi
59	59	175	153	52	170	160	33	tinggi
60	63	242	196	34	231	150	34	tinggi
61	54	284	113	21	170	120	35	tinggi
62	69	238	335	47	292	130	35	tinggi
63	70	186	176	33	200	170	35	tinggi
64	68	291	145	38	281	180	36	tinggi
65	70	189	180	58	273	180	36	tinggi
66	50	299	215	26	156	170	38	tinggi
67	57	285	132	30	274	140	39	tinggi
68	63	292	166	58	175	160	39	tinggi
69	58	269	133	30	198	170	39	tinggi
70	44	184	325	40	330	170	39	tinggi

Lampiran 2 Data Pelatihan 60

No	umur	Ldl	kol_total	hdl	trigliserida	sistolik	nilai_risiko	Kategori
1	60	171	182	57	286	140	20	sedang
2	58	134	223	25	130	150	20	sedang
3	61	167	231	39	254	130	21	sedang
4	77	148	208	33	134	180	22	sedang
5	38	119	289	41	232	140	22	sedang

6	63	186	197	40	319	120	22	sedang
7	60	150	218	32	284	140	23	sedang
8	51	196	200	72	159	140	23	sedang
9	47	192	171	41	350	120	23	sedang
10	50	166	236	35	284	130	24	sedang
11	65	137	214	13	302	140	25	sedang
12	62	178	215	35	258	120	25	sedang
13	68	256	221	48	190	140	25	sedang
14	64	128	214	44	193	180	25	sedang
15	71	105	218	51	203	160	25	sedang
16	82	90	164	40	169	170	25	sedang
17	72	108	213	48	304	180	25	sedang
18	68	123	232	45	292	170	25	sedang
19	49	146	221	44	162	170	26	sedang
20	67	209	194	48	112	160	26	sedang
21	66	151	229	60	272	180	27	sedang
22	38	195	270	35	166	130	27	sedang
23	63	151	224	50	162	130	28	sedang
24	56	198	173	38	131	200	28	sedang
25	61	98	256	33	269	150	29	sedang
26	65	158	229	30	272	150	29	sedang
27	55	154	253	41	167	150	29	sedang
28	51	104	390	53	176	150	29	sedang
29	43	195	181	60	154	170	29	sedang
30	63	299	159	47	152	160	30	sedang
31	73	170	149	44	144	140	10	rendah
32	70	264	226	44	111	140	15	rendah
33	56	145	187	41	278	130	17	rendah
34	66	148	130	48	270	140	17	rendah
35	59	149	172	52	129	180	18	rendah
36	67	94	182	75	149	160	19	rendah
37	51	79	189	34	399	140	19	rendah
38	58	152	257	56	202	160	33	tinggi
39	68	140	220	37	205	130	19	rendah
40	61	90	155	90	101	130	5	sangat rendah
41	35	380	350	30	280	180	42	sangat tinggi
42	70	254	280	30	165	160	42	sangat tinggi
43	74	283	145	32	134	160	30	tinggi
44	67	167	255	39	220	140	31	tinggi
45	58	160	239	38	168	180	31	tinggi
46	60	148	266	39	292	160	32	tinggi
47	63	152	239	45	180	160	32	tinggi

48	62	158	269	59	162	200	33	tinggi
49	59	175	153	52	170	160	33	tinggi
50	63	242	196	34	231	150	34	tinggi
51	54	284	113	21	170	120	35	tinggi
52	69	238	335	47	292	130	35	tinggi
53	70	186	176	33	200	170	35	tinggi
54	68	291	145	38	281	180	36	tinggi
55	70	189	180	58	273	180	36	tinggi
56	50	299	215	26	156	170	38	tinggi
57	57	285	132	30	274	140	39	tinggi
58	63	292	166	58	175	160	39	tinggi
59	58	269	133	30	198	170	39	tinggi
60	44	184	325	40	330	170	39	tinggi

Lampiran 3 Data Pelatihan 50

No	umur	ldl	kol_total	hdl	trigliserida	sistolik	nilai_risiko	Kategori
1	58	134	223	25	130	150	20	sedang
2	61	167	231	39	254	130	21	sedang
3	77	148	208	33	134	180	22	sedang
4	38	119	289	41	232	140	22	sedang
5	51	196	200	72	159	140	23	sedang
6	47	192	171	41	350	120	23	sedang
7	50	166	236	35	284	130	24	sedang
8	65	137	214	13	302	140	25	sedang
9	62	178	215	35	258	120	25	sedang
10	68	256	221	48	190	140	25	sedang
11	64	128	214	44	193	180	25	sedang
12	71	105	218	51	203	160	25	sedang
13	82	90	164	40	169	170	25	sedang
14	72	108	213	48	304	180	25	sedang
15	68	123	232	45	292	170	25	sedang
16	49	146	221	44	162	170	26	sedang
17	67	209	194	48	112	160	26	sedang
18	66	151	229	60	272	180	27	sedang
19	38	195	270	35	166	130	27	sedang
20	63	151	224	50	162	130	28	sedang
21	56	198	173	38	131	200	28	sedang
22	61	98	256	33	269	150	29	sedang
23	65	158	229	30	272	150	29	sedang
24	55	154	253	41	167	150	29	sedang
25	51	104	390	53	176	150	29	sedang
26	63	299	159	47	152	160	30	sedang

27	73	170	149	44	144	140	10	rendah
28	70	264	226	44	111	140	15	rendah
29	56	145	187	41	278	130	17	rendah
30	59	149	172	52	129	180	18	rendah
31	67	94	182	75	149	160	19	rendah
32	51	79	189	34	399	140	19	rendah
33	61	90	155	90	101	130	5	sangat rendah
34	35	380	350	30	280	180	42	sangat tinggi
35	70	254	280	30	165	160	42	sangat tinggi
36	67	167	255	39	220	140	31	tinggi
37	63	152	239	45	180	160	32	tinggi
38	62	158	269	59	162	200	33	tinggi
39	59	175	153	52	170	160	33	tinggi
40	63	242	196	34	231	150	34	tinggi
41	54	284	113	21	170	120	35	tinggi
42	69	238	335	47	292	130	35	tinggi
43	70	186	176	33	200	170	35	tinggi
44	68	291	145	38	281	180	36	tinggi
45	70	189	180	58	273	180	36	tinggi
46	50	299	215	26	156	170	38	tinggi
47	57	285	132	30	274	140	39	tinggi
48	63	292	166	58	175	160	39	tinggi
49	58	269	133	30	198	170	39	tinggi
50	44	184	325	40	330	170	39	tinggi

Lampiran 4 Data Pelatihan 40

No	umur	ldl	kol_total	hdl	triglicerida	sistolik	nilai_risiko	Kategori
1	58	134	223	25	130	150	20	sedang
2	61	167	231	39	254	130	21	sedang
3	77	148	208	33	134	180	22	sedang
4	68	256	221	48	190	140	25	sedang
5	64	128	214	44	193	180	25	sedang
6	71	105	218	51	203	160	25	sedang
7	82	90	164	40	169	170	25	sedang
8	72	108	213	48	304	180	25	sedang
9	68	123	232	45	292	170	25	sedang
10	49	146	221	44	162	170	26	sedang
11	67	209	194	48	112	160	26	sedang
12	66	151	229	60	272	180	27	sedang
13	38	195	270	35	166	130	27	sedang
14	63	151	224	50	162	130	28	sedang
15	56	198	173	38	131	200	28	sedang

16	61	98	256	33	269	150	29	sedang
17	65	158	229	30	272	150	29	sedang
18	55	154	253	41	167	150	29	sedang
19	51	104	390	53	176	150	29	sedang
20	63	299	159	47	152	160	30	sedang
21	73	170	149	44	144	140	10	rendah
22	70	264	226	44	111	140	15	rendah
23	56	145	187	41	278	130	17	rendah
24	67	94	182	75	149	160	19	rendah
25	51	79	189	34	399	140	19	rendah
26	61	90	155	90	101	130	5	sangat rendah
27	35	380	350	30	280	180	42	sangat tinggi
28	70	254	280	30	165	160	42	sangat tinggi
29	67	167	255	39	220	140	31	tinggi
30	63	152	239	45	180	160	32	tinggi
31	62	158	269	59	162	200	33	tinggi
32	59	175	153	52	170	160	33	tinggi
33	63	242	196	34	231	150	34	tinggi
34	54	284	113	21	170	120	35	tinggi
35	70	189	180	58	273	180	36	tinggi
36	50	299	215	26	156	170	38	tinggi
37	57	285	132	30	274	140	39	tinggi
38	63	292	166	58	175	160	39	tinggi
39	58	269	133	30	198	170	39	tinggi
40	44	184	325	40	330	170	39	tinggi

Lampiran 5 Data Pelatihan Sangat Rendah -Rendah-Sedang-Tinggi-Sangat Tinggi

No	umur	ldl	kol_total	hdl	trigliserida	sistolik	nilai_risiko	Kategori
1	57	115	218	66	274	150	20	sedang
2	60	171	182	57	286	140	20	sedang
3	58	134	223	25	130	150	20	sedang
4	61	167	231	39	254	130	21	sedang
5	77	148	208	33	134	180	22	sedang
6	77	186	168	66	177	130	22	sedang
7	41	185	208	50	236	130	22	sedang
8	38	119	289	41	232	140	22	sedang
9	63	186	197	40	319	120	22	Sedang
10	60	150	218	32	284	140	23	Sedang
11	51	196	200	72	159	140	23	Sedang
12	47	192	171	41	350	120	23	Sedang
13	67	191	196	53	188	140	23	Sedang
14	59	176	245	44	233	150	24	Sedang

15	50	166	236	35	284	130	24	Sedang
16	65	137	214	13	302	140	25	Sedang
17	62	178	215	35	258	120	25	Sedang
18	68	256	221	48	190	140	25	sedang
19	64	128	214	44	193	180	25	sedang
20	71	105	218	51	203	160	25	sedang
21	82	90	164	40	169	170	25	sedang
22	72	108	213	48	304	180	25	sedang
23	68	123	232	45	292	170	25	sedang
24	49	146	221	44	162	170	26	sedang
25	67	209	194	48	112	160	26	sedang
26	66	151	229	60	272	180	27	sedang
27	38	195	270	35	166	130	27	sedang
28	63	151	224	50	162	130	28	sedang
29	56	198	173	38	131	200	28	sedang
30	61	98	256	33	269	150	29	sedang
31	65	158	229	30	272	150	29	sedang
32	55	154	253	41	167	150	29	sedang
33	51	104	390	53	176	150	29	sedang
34	43	195	181	60	154	170	29	sedang
35	63	299	159	47	152	160	30	sedang
36	73	170	149	44	144	140	10	rendah
37	55	190	160	39	123	120	14	rendah
38	60	116	170	40	129	180	15	rendah
39	70	264	226	44	111	140	15	rendah
40	56	145	187	41	278	130	17	rendah
41	66	148	130	48	270	140	17	rendah
42	59	149	172	52	129	180	18	rendah
43	67	94	182	75	149	160	19	rendah
44	58	105	170	34	223	140	19	rendah
45	51	79	189	34	399	140	19	rendah
46	68	140	220	37	205	130	19	rendah
47	61	90	155	90	101	130	5	sangat rendah
48	35	380	350	30	280	180	42	sangat tinggi
49	70	254	280	30	165	160	42	sangat tinggi
50	50	164	243	39	193	160	30	tinggi
51	74	283	145	32	134	160	30	tinggi
52	67	167	255	39	220	140	31	tinggi
53	58	160	239	38	168	180	31	tinggi
54	60	148	266	39	292	160	32	tinggi
55	63	152	239	45	180	160	32	tinggi
56	54	178	262	54	173	160	33	tinggi

57	65	183	248	36	196	140	33	tinggi
58	62	158	269	59	162	200	33	tinggi
59	59	175	153	52	170	160	33	tinggi
60	63	242	196	34	231	150	34	tinggi
61	54	284	113	21	170	120	35	tinggi
62	69	238	335	47	292	130	35	tinggi
63	70	186	176	33	200	170	35	tinggi
64	68	291	145	38	281	180	36	tinggi
65	70	189	180	58	273	180	36	tinggi
66	50	299	215	26	156	170	38	tinggi
67	57	285	132	30	274	140	39	tinggi
68	63	292	166	58	175	160	39	tinggi
69	58	269	133	30	198	170	39	tinggi
70	44	184	325	40	330	170	39	tinggi

Lampiran 6 Data Pelatihan Sangat Rendah -Rendah-Sedang

No	umur	ldl	kol_total	hdl	trigliserida	sistolik	nilai_risiko	Kategori
1	57	115	218	66	274	150	20	sedang
2	60	171	182	57	286	140	20	sedang
3	58	134	223	25	130	150	20	sedang
4	61	167	231	39	254	130	21	sedang
5	77	148	208	33	134	180	22	sedang
6	77	186	168	66	177	130	22	sedang
7	41	185	208	50	236	130	22	sedang
8	38	119	289	41	232	140	22	sedang
9	63	186	197	40	319	120	22	Sedang
10	60	150	218	32	284	140	23	Sedang
11	51	196	200	72	159	140	23	Sedang
12	47	192	171	41	350	120	23	Sedang
13	67	191	196	53	188	140	23	Sedang
14	59	176	245	44	233	150	24	Sedang
15	50	166	236	35	284	130	24	Sedang
16	65	137	214	13	302	140	25	Sedang
17	62	178	215	35	258	120	25	Sedang
18	68	256	221	48	190	140	25	sedang
19	64	128	214	44	193	180	25	sedang
20	71	105	218	51	203	160	25	sedang
21	82	90	164	40	169	170	25	sedang
22	72	108	213	48	304	180	25	sedang
23	68	123	232	45	292	170	25	sedang
24	49	146	221	44	162	170	26	sedang
25	67	209	194	48	112	160	26	sedang

26	66	151	229	60	272	180	27	sedang
27	38	195	270	35	166	130	27	sedang
28	63	151	224	50	162	130	28	sedang
29	56	198	173	38	131	200	28	sedang
30	61	98	256	33	269	150	29	sedang
31	65	158	229	30	272	150	29	sedang
32	55	154	253	41	167	150	29	sedang
33	51	104	390	53	176	150	29	sedang
34	43	195	181	60	154	170	29	sedang
35	63	299	159	47	152	160	30	sedang
36	73	170	149	44	144	140	10	rendah
37	55	190	160	39	123	120	14	rendah
38	60	116	170	40	129	180	15	rendah
39	70	264	226	44	111	140	15	rendah
40	56	145	187	41	278	130	17	rendah
41	66	148	130	48	270	140	17	rendah
42	59	149	172	52	129	180	18	rendah
43	67	94	182	75	149	160	19	rendah
44	58	105	170	34	223	140	19	rendah
45	51	79	189	34	399	140	19	rendah
46	68	140	220	37	205	130	19	rendah
47	61	90	155	90	101	130	5	sangat rendah

Lampiran 7 Data Pelatihan Rendah-Sedang-Tinggi

No	umur	ldl	kol_total	hdl	trigliserida	sistolik	nilai_risiko	Kategori
1	57	115	218	66	274	150	20	sedang
2	60	171	182	57	286	140	20	sedang
3	58	134	223	25	130	150	20	sedang
4	61	167	231	39	254	130	21	sedang
5	77	148	208	33	134	180	22	sedang
6	77	186	168	66	177	130	22	sedang
7	41	185	208	50	236	130	22	sedang
8	38	119	289	41	232	140	22	sedang
9	63	186	197	40	319	120	22	Sedang
10	60	150	218	32	284	140	23	Sedang
11	51	196	200	72	159	140	23	Sedang
12	47	192	171	41	350	120	23	Sedang
13	67	191	196	53	188	140	23	Sedang
14	59	176	245	44	233	150	24	Sedang
15	50	166	236	35	284	130	24	Sedang
16	65	137	214	13	302	140	25	Sedang
17	62	178	215	35	258	120	25	Sedang

18	68	256	221	48	190	140	25	sedang
19	64	128	214	44	193	180	25	sedang
20	71	105	218	51	203	160	25	sedang
21	82	90	164	40	169	170	25	sedang
22	72	108	213	48	304	180	25	sedang
23	68	123	232	45	292	170	25	sedang
24	49	146	221	44	162	170	26	sedang
25	67	209	194	48	112	160	26	sedang
26	66	151	229	60	272	180	27	sedang
27	38	195	270	35	166	130	27	sedang
28	63	151	224	50	162	130	28	sedang
29	56	198	173	38	131	200	28	sedang
30	61	98	256	33	269	150	29	sedang
31	65	158	229	30	272	150	29	sedang
32	55	154	253	41	167	150	29	sedang
33	51	104	390	53	176	150	29	sedang
34	43	195	181	60	154	170	29	sedang
35	63	299	159	47	152	160	30	sedang
36	73	170	149	44	144	140	10	rendah
37	55	190	160	39	123	120	14	rendah
38	60	116	170	40	129	180	15	rendah
39	70	264	226	44	111	140	15	rendah
40	56	145	187	41	278	130	17	rendah
41	66	148	130	48	270	140	17	rendah
42	59	149	172	52	129	180	18	rendah
43	67	94	182	75	149	160	19	rendah
44	58	105	170	34	223	140	19	rendah
45	51	79	189	34	399	140	19	rendah
46	68	140	220	37	205	130	19	rendah
47	50	164	243	39	193	160	30	tinggi
48	74	283	145	32	134	160	30	tinggi
49	67	167	255	39	220	140	31	tinggi
50	58	160	239	38	168	180	31	tinggi
51	60	148	266	39	292	160	32	tinggi
52	63	152	239	45	180	160	32	tinggi
53	54	178	262	54	173	160	33	tinggi
54	65	183	248	36	196	140	33	tinggi
55	62	158	269	59	162	200	33	tinggi
56	59	175	153	52	170	160	33	tinggi
57	63	242	196	34	231	150	34	tinggi
58	54	284	113	21	170	120	35	tinggi
59	69	238	335	47	292	130	35	tinggi

60	70	186	176	33	200	170	35	tinggi
61	68	291	145	38	281	180	36	tinggi
62	70	189	180	58	273	180	36	tinggi
63	50	299	215	26	156	170	38	tinggi
64	57	285	132	30	274	140	39	tinggi
65	63	292	166	58	175	160	39	tinggi
66	58	269	133	30	198	170	39	tinggi
67	44	184	325	40	330	170	39	tinggi

Lampiran 8 Data Pelatihan Sedang-Tinggi-Sangat Tinggi

No	umur	ldl	kol_total	hdl	trigliserida	sistolik	nilai_risiko	Kategori
1	57	115	218	66	274	150	20	sedang
2	60	171	182	57	286	140	20	sedang
3	58	134	223	25	130	150	20	sedang
4	61	167	231	39	254	130	21	sedang
5	77	148	208	33	134	180	22	sedang
6	77	186	168	66	177	130	22	sedang
7	41	185	208	50	236	130	22	sedang
8	38	119	289	41	232	140	22	sedang
9	63	186	197	40	319	120	22	Sedang
10	60	150	218	32	284	140	23	Sedang
11	51	196	200	72	159	140	23	Sedang
12	47	192	171	41	350	120	23	Sedang
13	67	191	196	53	188	140	23	Sedang
14	59	176	245	44	233	150	24	Sedang
15	50	166	236	35	284	130	24	Sedang
16	65	137	214	13	302	140	25	Sedang
17	62	178	215	35	258	120	25	Sedang
18	68	256	221	48	190	140	25	sedang
19	64	128	214	44	193	180	25	sedang
20	71	105	218	51	203	160	25	sedang
21	82	90	164	40	169	170	25	sedang
22	72	108	213	48	304	180	25	sedang
23	68	123	232	45	292	170	25	sedang
24	49	146	221	44	162	170	26	sedang
25	67	209	194	48	112	160	26	sedang
26	66	151	229	60	272	180	27	sedang
27	38	195	270	35	166	130	27	sedang
28	63	151	224	50	162	130	28	sedang
29	56	198	173	38	131	200	28	sedang
30	61	98	256	33	269	150	29	sedang
31	65	158	229	30	272	150	29	sedang

32	55	154	253	41	167	150	29	sedang
33	51	104	390	53	176	150	29	sedang
34	43	195	181	60	154	170	29	sedang
35	63	299	159	47	152	160	30	sedang
36	35	380	350	30	280	180	42	sangat tinggi
37	70	254	280	30	165	160	42	sangat tinggi
38	50	164	243	39	193	160	30	tinggi
39	74	283	145	32	134	160	30	tinggi
40	67	167	255	39	220	140	31	tinggi
41	58	160	239	38	168	180	31	tinggi
42	60	148	266	39	292	160	32	tinggi
43	63	152	239	45	180	160	32	tinggi
44	54	178	262	54	173	160	33	tinggi
45	65	183	248	36	196	140	33	tinggi
46	62	158	269	59	162	200	33	tinggi
47	59	175	153	52	170	160	33	tinggi
48	63	242	196	34	231	150	34	tinggi
49	54	284	113	21	170	120	35	tinggi
50	69	238	335	47	292	130	35	tinggi
51	70	186	176	33	200	170	35	tinggi
52	68	291	145	38	281	180	36	tinggi
53	70	189	180	58	273	180	36	tinggi
54	50	299	215	26	156	170	38	tinggi
55	57	285	132	30	274	140	39	tinggi
56	63	292	166	58	175	160	39	tinggi
57	58	269	133	30	198	170	39	tinggi
58	44	184	325	40	330	170	39	tinggi

Lampiran 9 Data Pengujian

No	umur	ldl	kol_total	hdl	trigliserida	sistolik	nilai_risiko	Kategori
1	45	123	266	54	147	200	23	sedang
2	67	143	221	25	267	130	26	sedang
3	49	156	240	55	138	160	15	sedang
4	66	190	165	62	116	170	24	sedang
5	70	161	201	38	110	170	22	sedang
6	58	154	230	37	224	140	22	sedang
7	67	280	232	47	294	140	25	sedang
8	68	153	201	39	298	180	29	sedang
9	75	190	148	42	174	150	27	sedang
10	46	292	152	40	185	120	25	sedang
11	41	176	236	47	205	150	25	sedang
12	32	192	167	40	155	170	22	sedang

13	68	138	212	37	229	180	28	sedang
14	44	174	203	47	145	160	23	sedang
15	60	136	193	37	140	180	19	rendah
16	63	143	214	49	212	130	16	rendah
17	53	135	185	38	137	170	17	rendah
18	62	113	183	43	144	170	15	rendah
19	55	167	119	35	96	140	8	sangat rendah
20	50	324	290	32	190	170	42	sangat tinggi
21	53	160	200	39	160	150	32	tinggi
22	56	214	301	61	209	150	39	tinggi
23	67	147	229	42	200	190	33	tinggi
24	54	299	158	48	195	200	35	tinggi
25	67	196	302	70	159	170	38	tinggi
26	39	198	189	34	235	130	36	tinggi
27	47	235	291	30	178	180	39	tinggi
28	63	152	294	60	243	150	32	tinggi
29	72	163	255	45	163	180	33	tinggi
30	60	154	259	58	204	160	33	tinggi

Lampiran 10 Hasil Pengujian dengan 40 data latih

- K=1

data uji -	Prediksi Sistem	Prediksi Sebenarnya	Hasil Prediksi
Data Uji-1	Tinggi	Sedang	Salah
Data Uji-2	Sedang	Sedang	Benar
Data Uji-3	Sedang	Sedang	Benar
Data Uji-4	Sedang	Sedang	Benar
Data Uji-5	Sedang	Sedang	Benar
Data Uji-6	Sedang	Sedang	Benar
Data Uji-7	Sedang	Sedang	Benar
Data Uji-8	Sedang	Sedang	Benar
Data Uji-9	Sangat_Rendah	Sedang	Salah
Data Uji-10	Tinggi	Sedang	Salah
Data Uji-11	Sedang	Sedang	Benar
Data Uji-12	Sedang	Sedang	Benar
Data Uji-13	Sedang	Sedang	Benar
Data Uji-14	Sedang	Sedang	Benar
Data Uji-15	Sedang	Rendah	Salah
Data Uji-16	Sedang	Rendah	Salah
Data Uji-17	Sedang	Rendah	Salah
Data Uji-18	Sedang	Rendah	Salah
Data Uji-19	Tinggi	Sangat_Rendah	Salah
Data Uji-20	Tinggi	Sangat_Tinggi	Salah

Data Uji-21	Sedang	Tinggi	Salah
Data Uji-22	Sedang	Tinggi	Salah
Data Uji-23	Sedang	Tinggi	Salah
Data Uji-24	Sedang	Tinggi	Salah
Data Uji-25	Sangat_Tinggi	Tinggi	Salah
Data Uji-26	Sedang	Tinggi	Salah
Data Uji-27	Tinggi	Tinggi	Benar
Data Uji-28	Tinggi	Tinggi	Benar
Data Uji-29	Sedang	Tinggi	Salah
Data Uji-30	Tinggi	Tinggi	Benar

• K=2

data uji -	Prediksi Sistem	Prediksi Sebenarnya	Hasil Prediksi
Data Uji-1	Tinggi	Sedang	Salah
Data Uji-2	Sedang	Sedang	Benar
Data Uji-3	Sedang	Sedang	Benar
Data Uji-4	Sedang	Sedang	Benar
Data Uji-5	Sedang	Sedang	Benar
Data Uji-6	Sedang	Sedang	Benar
Data Uji-7	Sedang	Sedang	Benar
Data Uji-8	Sedang	Sedang	Benar
Data Uji-9	Sangat_Rendah	Sedang	Salah
Data Uji-10	Tinggi	Sedang	Salah
Data Uji-11	Sedang	Sedang	Benar
Data Uji-12	Sedang	Sedang	Benar
Data Uji-13	Sedang	Sedang	Benar
Data Uji-14	Sedang	Sedang	Benar
Data Uji-15	Sedang	Rendah	Salah
Data Uji-16	Sedang	Rendah	Salah
Data Uji-17	Sedang	Rendah	Salah
Data Uji-18	Sedang	Rendah	Salah
Data Uji-19	Tinggi	Sangat_Rendah	Salah
Data Uji-20	Tinggi	Sangat_Tinggi	Salah
Data Uji-21	Sedang	Tinggi	Salah
Data Uji-22	Tinggi	Tinggi	Benar
Data Uji-23	Sedang	Tinggi	Salah
Data Uji-24	Sedang	Tinggi	Salah
Data Uji-25	Tinggi	Tinggi	Benar
Data Uji-26	Sedang	Tinggi	Salah
Data Uji-27	Tinggi	Tinggi	Benar

Data Uji-28	Tinggi	Tinggi	Benar
Data Uji-29	Sedang	Tinggi	Salah
Data Uji-30	Tinggi	Tinggi	Benar

• K=3

data uji -	Prediksi Sistem	Prediksi Sebenarnya	Hasil Prediksi
Data Uji-1	Sedang	Sedang	Benar
Data Uji-2	Sedang	Sedang	Benar
Data Uji-3	Sedang	Sedang	Benar
Data Uji-4	Sedang	Sedang	Benar
Data Uji-5	Sedang	Sedang	Benar
Data Uji-6	Sedang	Sedang	Benar
Data Uji-7	Sedang	Sedang	Benar
Data Uji-8	Sedang	Sedang	Benar
Data Uji-9	Sangat_Rendah	Sedang	Salah
Data Uji-10	Tinggi	Sedang	Salah
Data Uji-11	Sedang	Sedang	Benar
Data Uji-12	Sedang	Sedang	Benar
Data Uji-13	Sedang	Sedang	Benar
Data Uji-14	Sedang	Sedang	Benar
Data Uji-15	Sedang	Rendah	Salah
Data Uji-16	Sedang	Rendah	Salah
Data Uji-17	Sedang	Rendah	Salah
Data Uji-18	Sedang	Rendah	Salah
Data Uji-19	Rendah	Sangat_Rendah	Salah
Data Uji-20	Tinggi	Sangat_Tinggi	Salah
Data Uji-21	Sedang	Tinggi	Salah
Data Uji-22	Tinggi	Tinggi	Benar
Data Uji-23	Sedang	Tinggi	Salah
Data Uji-24	Tinggi	Tinggi	Benar
Data Uji-25	Tinggi	Tinggi	Benar
Data Uji-26	Sedang	Tinggi	Salah
Data Uji-27	Tinggi	Tinggi	Benar
Data Uji-28	Tinggi	Tinggi	Benar
Data Uji-29	Sedang	Tinggi	Salah
Data Uji-30	Sedang	Tinggi	Salah

• K=4

data uji -	Prediksi Sistem	Prediksi Sebenarnya	Hasil Prediksi
Data Uji-1	Sedang	Sedang	Benar
Data Uji-2	Sedang	Sedang	Benar
Data Uji-3	Sedang	Sedang	Benar
Data Uji-4	Sedang	Sedang	Benar
Data Uji-5	Sedang	Sedang	Benar
Data Uji-6	Sedang	Sedang	Benar
Data Uji-7	Sedang	Sedang	Benar
Data Uji-8	Sedang	Sedang	Benar
Data Uji-9	Sedang	Sedang	Benar
Data Uji-10	Tinggi	Sedang	Salah
Data Uji-11	Sedang	Sedang	Benar
Data Uji-12	Tinggi	Sedang	Salah
Data Uji-13	Sedang	Sedang	Benar
Data Uji-14	Sedang	Sedang	Benar
Data Uji-15	Sedang	Rendah	Salah
Data Uji-16	Sedang	Rendah	Salah
Data Uji-17	Sedang	Rendah	Salah
Data Uji-18	Sedang	Rendah	Salah
Data Uji-19	Sedang	Sangat_Rendah	Salah
Data Uji-20	Tinggi	Sangat_Tinggi	Salah
Data Uji-21	Sedang	Tinggi	Salah
Data Uji-22	Tinggi	Tinggi	Benar
Data Uji-23	Sedang	Tinggi	Salah
Data Uji-24	Tinggi	Tinggi	Benar
Data Uji-25	Tinggi	Tinggi	Benar
Data Uji-26	Sedang	Tinggi	Salah
Data Uji-27	Tinggi	Tinggi	Benar
Data Uji-28	Tinggi	Tinggi	Benar
Data Uji-29	Sedang	Tinggi	Salah
Data Uji-30	Tinggi	Tinggi	Benar

Lampiran 11 Pengujian Akurasi Berulang pada Data Latih 50 (kategori sama)

k	Data Latih					Rata-rata (%)
	I	II	III	IV	V	
1	56.67 %	56.67 %	56.67 %	56.67 %	56.67 %	56.67 %
2	60 %	60 %	60 %	60 %	60 %	60 %
3	63.33 %	63.33 %	63.33 %	63.33 %	63.33 %	63.33 %
4	56.67 %	56.67 %	56.67 %	56.67 %	56.67 %	56.67 %
5	56.67 %	56.67 %	56.67 %	56.67 %	56.67 %	56.67 %
6	50 %	50 %	50 %	50 %	50 %	50 %
7	46.67 %	46.67 %	46.67 %	46.67 %	46.67 %	46.67 %
8	46.67 %	46.67 %	46.67 %	46.67 %	46.67 %	46.67 %
9	46.67 %	46.67 %	46.67 %	46.67 %	46.67 %	46.67 %

Dapat terlihat bahwa setiap percobaan yang dilakukan berulang pada data uji dan data latih yang sama, didapatkan nilai akurasi yang sama dari ke lima percobaan yang telah dilakukan.

