

**PENERAPAN METODE AHP DAN SVM UNTUK KLASIFIKASI
PENERIMA BEASISWA
(STUDI KASUS : LEMBAGA GNOTA KEDIRI)**

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:
Whenty Ariyanti
NIM: 145150209111021



**PROGRAM STUDI TEKNIK INFORMATIKA/ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2016**

PENGESAHAN

PENERAPAN METODE AHP DAN SVM UNTUK KLASIFIKASI PENERIMA BEASISWA
(STUDI KASUS : LEMBAGA GNOTA KEDIRI)

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :

Whenty Ariyanti

NIM: 145150209111021

Skripsi ini telah diuji dan dinyatakan lulus pada
19 Mei 2016

Telah diperiksa dan disetujui oleh :

Dosen Pembimbing I

Dosen Pembimbing II

Imam Cholissodin, S.Si., M.Kom

NIK: 201201 850719 1001

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

NIP: 198410152014041002

Mengetahui,

Ketua Program Studi Informatika/Illmu Komputer

Issa Arwani, S.Kom, M.Sc

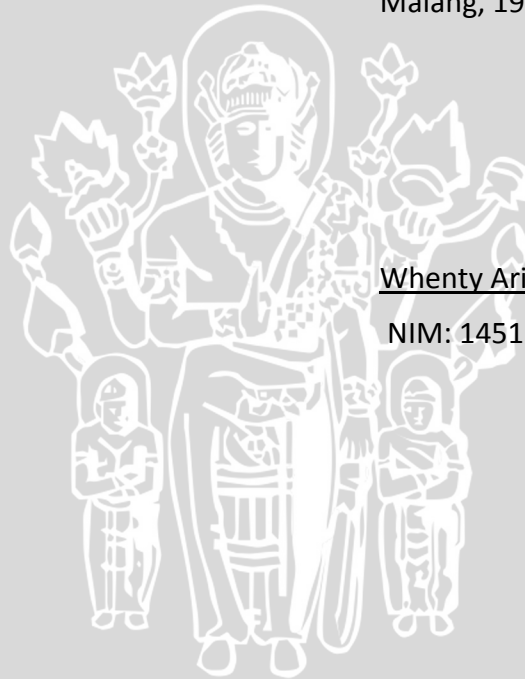
NIP : 19830922 201212 1 003

PERNYATAAN ORISINALITAS

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 disitasi dalam naskah ini dan disebutkan dalam 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, 19 Mei 2016



Whenty Ariyanti

NIM: 145150209111021

KATA PENGANTAR

Puji syukur kehadiran Allah SWT yang telah melimpahkan rahmat, taufiq dan hidayah-Nya sehingga laporan skripsi yang berjudul “Penerapan Metode AHP dan SVM untuk klasifikasi Penerima Beasiswa (Studi Kasus : Lembaga GNOTA Kediri)” ini dapat terselesaikan.

Penulis menyadari bahwa skripsi ini tidak akan berhasil tanpa bantuan dari beberapa pihak. Oleh karena itu, penulis ingin menyampaikan rasa hormat dan terimakasih kepada :

1. Bapak Imam Cholissodin, S.Si., M.Kom dan Bapak Budi Darma Setiawan, S.Kom., M.Cs selaku dosen pembimbing skripsi yang telah dengan sabar membimbing dan mengarahkan penulis sehingga dapat menyelesaikan skripsi ini.
2. Bapak Wayan Firdaus Mahmudy, S.Si, M.T, Ph.D selaku Dekan Fakultas Ilmu Komputer Universitas Brawijaya Malang.
3. Ir. Heru Nurwasito M.Kom, Drs. Mardji, M.T dan Edy Santoso, S.Si., M.Kom selaku Wakil Dekan I, Wakil Dekan II dan Wakil Dekan III Fakultas Ilmu Komputer Universitas Brawijaya Malang.
4. Bapak Issa Arwani, S.Kom., M.Sc selaku Kepala Program Studi Ilmu Komputer/Informatika Universitas Brawijaya Malang.
5. Ibu Lailil Muflikah S.Kom., M.Sc selaku dosen penasehat akademik yang selalu memberikan nasehat kepada penulis selama menempuh masa studi.
6. Pemerintah Daerah Kabupaten Kediri yang memberikan ijin dalam melakukan penelitian ini, serta seluruh instansi terkait.
7. Lembaga GNOTA (Gerakan Nasional Orang Tua Asuh) Kabupaten Kediri yang telah bersedia dalam memberikan fasilitas dan waktu dalam penelitian ini.
8. Ibu dan Bapak serta seluruh keluarga besar yang telah membesarkan saya dari dahulu, maaf belum bisa berkontribusi apa-apa dan terimakasih atas nasehat, kasih sayang, perhatian dan kesabarannya di dalam membesarkan dan mendidik penulis, serta yang senantiasa tiada henti-hentinya memberikan do’a dan semangat demi terselesaikannya skripsi ini.
9. Sahabat seperlabilan *all-nighter* pejuang skripsi Garsinia, Badi’, Rima, Gilvy yang telah memberikan waktunya untuk selalu menyemangati dan menemani saya dalam mengerjakan skripsi ini. Terimakasih atas malam-malam skripsi yang bahagia, akhirnya kita berhasil menyelesaikan skripsi setelah berbagai masalah. Saya sangat bersyukur dipertemukan dengan sahabat-sababat seperti kalian.

10. Teman-teman Alih Program Teknik Informatika 2014 yang selalu memberikan bantuan dan motivasi dalam menyelesaikan skripsi ini.
11. Seluruh civitas akademik Informatika Universitas Brawijaya yang telah banyak memberi bantuan dan dukungan selama penulis menempuh studi di Informatika Universitas Brawijaya dan selama penyelesaian skripsi ini.

Penulis menyadari bahwa dalam penyusunan skripsi ini masih banyak kekurangan sehingga saran dan kritik yang membangun sangat penulis harapkan. Akhir kata penulis berharap skripsi ini dapat membawa manfaat bagi semua pihak yang menggunakannya.

Malang, 19 Mei 2016

Whenty Ariyanti

whenty.ariyanti@gmail.com



ABSTRAK

Whenty Ariyanti, 2016. Penerapan Metode AHP dan SVM untuk Klasifikasi Penerima Beasiswa (Studi Kasus : Lembaga GNOTA Kediri). Skripsi Program Studi Informatika/Ilmu Komputer, Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya. Pembimbing : Imam Cholissodin, S.Si., M.Kom dan Budi Darma Setiawan, S.Kom., M.Cs.

Pendidikan merupakan kebutuhan pokok bagi seluruh masyarakat Indonesia. Manfaat pendidikan bagi masyarakat Indonesia sangat beragam seperti untuk karir atau pekerjaan. Dalam penelitian ini diambil kasus pemilihan penerima beasiswa pada Lembaga GNOTA (Gerakan Nasional Orang Tua Asuh) Kediri sehingga nantinya kita dapat melakukan penyeleksian secara *realtime*, cepat dan akurat terhadap penerima beasiswa dengan suatu sistem penyeleksian yang mampu melakukan filterisasi secara otomatis proses pengambilan keputusan penentuan penerima beasiswa selama ini dilakukan secara manual. Manual disini maksudnya adalah data pemohon beasiswa direkapitulasi oleh panitia penyeleksi, lalu panitia melakukan survei satu persatu terhadap data yang diajukan oleh pemohon. Setelah proses survei dilakukan maka panitia akan meninjau dan menganalisa kembali data pemohon beasiswa tersebut. Proses ini membutuhkan waktu yang lama, kurang lebih tiga bulan hanya untuk meninjau dan menganalisa data pemohon beasiswa yang cukup banyak. Cara ini menyebabkan penilaian terhadap penerima beasiswa dilakukan secara subjektif, yang berarti penilaian berdasarkan sudut pandang panitia survei. Selain itu permasalahan ketepatan penyaluran beasiswa GNOTA yang selama ini sering terjadi dalam penentuan penerima beasiswa belum memiliki bobot secara pasti pada setiap kriteria dan sub kriteria yang dimiliki oleh pemohon beasiswa yang layak menerima beasiswa. "Penerapan Metode AHP dan SVM untuk Klasifikasi Penerima Beasiswa (Studi Kasus : Lembaga GNOTA Kediri)". Diharapkan dalam pemilihan penerima beasiswa digunakan metode AHP dan SVM untuk dapat menghasilkan *output* yang sesuai dengan keadaan atau kriteria yang ada sehingga dapat memudahkan Lembaga GNOTA dalam melakukan pengambilan keputusan dan meminimalkan kesalahan dalam pengambilan kebijakan. Tingkat akurasi rata-rata yang dihasilkan oleh sistem adalah sebesar 89.74% dengan pemilihan rasio 90%:10%, nilai *threshold* = 0.01, nilai lamda = 0.5, nilai konstanta gamma=0.003, nilai epsilon=0.001, iterasi maksimum=100 dan nilai C=1.

Kata kunci: AHP, SVM, GNOTA, Beasiswa

ABSTRACT

Whenty Ariyanti, 2016. Implementation AHP and SVM Method for Classification Receiver Scholarship (Case Study : GNOTA Institute Kediri). Undergraduate Thesis, Informatics/Computer Science Department, Information Technology and Computer Science Program Brawijaya University. Advisor : Imam Cholissodin S.Si., M.Kom and Budi Darma Setiawan S.Kom., M.Cs.

Education is a basic requirement for all Indonesian citizens. Educational benefits for the people of Indonesia are as varied as for career or job. In this research taken the case of the selection of scholarship recipients at the Institute GNOTA (National Movement for Foster Parents) Kediri so that later we can do the screening in real time, quickly and accurately to the grantee with a system of selection that is able to perform filtering automatic decision-making process of determining awardees as long as this is done manually. Manual here means the scholarship applicant data is summarized by the committee of selectors, the committee conducted a survey one by one to the data submitted by the applicant. After the survey was done, the committee will review and analyze data of the scholarship applicants. This process takes a long time, about three months just to review and analyze data that is quite a lot of scholarship applicants. In this way led to an assessment of the awardees is done subjectively, which means an assessment based on the viewpoint of the survey committee. Besides the problems precision GNOTA scholarship distribution which has been often the case in determining the recipients dont have weight exactly in each criteria and sub-criteria that are owned by the applicant that the eligible scholarship. "Application of AHP and SVM Method for Classification of Scholarship Recipients (Case Study: Non GNOTA Kediri)". Expected in the selection of awardees is used AHP and SVM to produce output appropriate to the circumstances or the existing criteria so as to facilitate the Institute GNOTA in making decisions and minimize errors in decision kebijakanTingkat average accuracy generated by the system is 89.74% with the selection ratio of 90: 10%, threshold value = 0.01, the value of lamda = 0.5, the value constanta gamma = 0.003, the value of epsilon = 0.001, maximum iteration=100 respectively, the value of C parameter = 1.

Keywords: AHP, SVM, GNOTA, Scholarships

DAFTAR ISI

PENGESAHAN	ii
PERNYATAAN ORISINALITAS	iii
KATA PENGANTAR.....	iv
ABSTRAK	vi
ABSTRACT	vii
DAFTAR ISI	viii
DAFTAR TABEL.....	xii
DAFTAR GAMBAR.....	xvi
DAFTAR KODE PROGRAM	xviii
DAFTAR LAMPIRAN	xix
BAB 1 PENDAHULUAN.....	1
1.1 Latar belakang.....	1
1.2 Rumusan masalah.....	3
1.3 Tujuan	3
1.4 Manfaat.....	3
1.5 Batasan masalah	3
1.6 Sistematika pembahasan.....	3
BAB 2 LANDASAN KEPUSTAKAAN	5
2.1 Kajian Pustaka	5
2.2 Beasiswa.....	8
2.3 Gerakan Nasional Orang Tua Asuh (GNOTA).....	9
2.4 Sistem Pendukung Keputusan	9
2.4.1 Konsep Dasar Sistem Pendukung Keputusan	10
2.4.2 Karakteristik dan Kemampuan Sistem Pendukung Keputusan ..	11
2.4.3 Fase-fase Proses Pengambilan Keputusan.....	12
2.5 <i>Analytical Hierarchy Process (AHP)</i>	14
2.5.1 Langkah-langkah Penggunaan Metode AHP	14
2.6 Normalisasi	17
2.7 Klasifikasi.....	18
2.8 Support Vector Machine (SVM).....	18
2.8.1 Support Vector Machine Linier	19

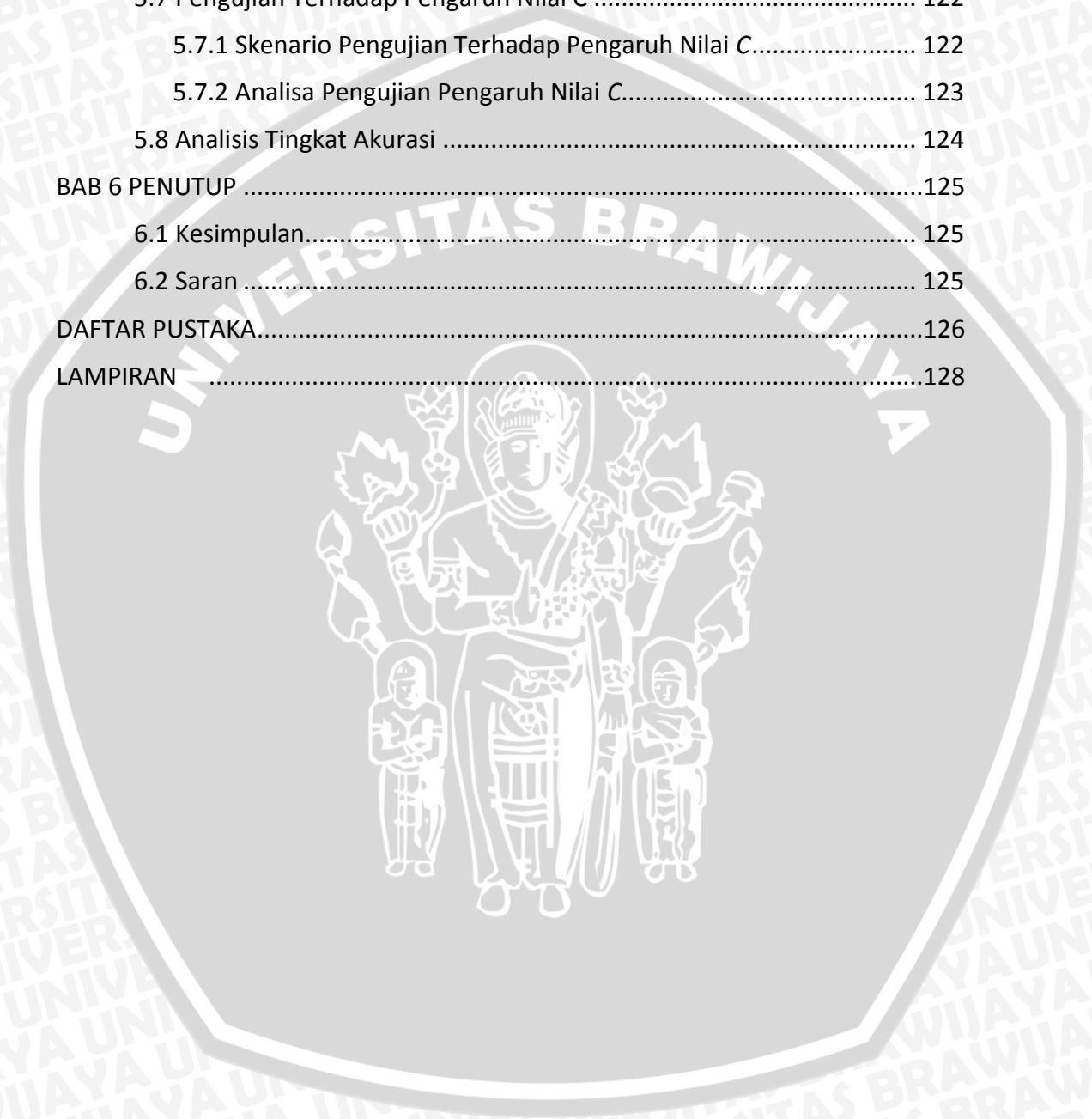
2.8.2 Support Vector Machine Non-Linier	21
2.9 Multi-Class SVM	22
2.9.1 Metode <i>One-Against-All</i>	23
2.9.2 Metode <i>One-Against-One</i>	23
2.10 Metode Sequential Training pada SVM	24
2.11 Evaluasi	25
BAB 3 METODOLOGI DAN PERANCANGAN	28
3.1 Studi Literatur	28
3.2 Pengumpulan Data	28
3.3 Perhitungan Klasifikasi	30
3.3.1 Deskripsi Umum.....	30
3.3.1 Model Perancangan Sistem	30
3.3.2 Perhitungan Manual.....	47
3.4 Implementasi Sistem	87
3.4.1 Sub Sistem Antar Muka.....	87
3.5 Perancangan Pengujian	88
3.5.1 Pengujian tingkat akurasi terhadap perbandingan jumlah data <i>training</i> dan data <i>testing</i>	88
3.5.2 Pengujian tingkat akurasi berdasarkan pengaruh nilai <i>Threshold</i> AHP.....	89
3.5.3 Pengujian Berdasarkan Nilai Parameter <i>Lamda</i>	89
3.5.4 Pengujian tingkat akurasi berdsasarkan nilai konstanta <i>gamma</i>	90
3.5.5 Pengujian tingkat akurasi terhadap pengaruh nilai <i>epsilon</i>	90
3.5.6 Pengujian tingkat akurasi terhadap pengaruh iterasi maksimum	91
3.5.7 Pengujian tingkat akurasi berdasarkan nilai <i>C (Complexity)</i>	91
3.6 Pengambilan Kesimpulan dan Saran	92
BAB 4 IMPLEMENTASI	93
4.1 Lingkup Implementasi.....	93
4.1.1 Lingkup Implementasi Perangkat Keras.....	94
4.1.2 Lingkup Implementasi Perangkat Lunak	94
4.2 Batasan Implementasi	94
4.3 Implementasi Algoritma	95



4.3.1 Implementasi Algoritma Perhitungan AHP	95
4.3.2 Implementasi Algoritma Normalisasi.....	97
4.3.3 Implementasi Algoritma Proses <i>Sequential Training</i>	99
4.3.4 Implementasi Algoritma Perhitungan Nilai $f(x)$	104
4.4 Implementasi Antarmuka	105
4.4.1 Antarmuka Konfigurasi Data Alternatif.....	106
4.4.2 Antarmuka Konfigurasi Matriks Perbandingan.....	106
4.4.3 Antarmuka Load Data Matriks Kriteria	107
4.4.4 Antarmuka Perhitungan Sub Bobot Kriteria AHP	107
4.4.5 Antarmuka Hasil Akhir Perhitungan AHP.....	108
4.4.6 Antarmuka Load Data SVM	108
4.4.7 Antarmuka Normalisasi Dataset SVM.....	109
4.4.8 Antarmuka Perkalian Bobot Setelah Normalisasi	109
4.4.9 Antarmuka Proses <i>Sequential Training</i>	110
4.4.10 Antarmuka Perhitungan Iterasi.....	110
4.4.11 Antarmuka Perhitungan Nilai $f(x)$	111
BAB 5 PENGUJIAN DAN ANALISIS.....	113
5.1 Pengujian Terhadap Rasio Perbandingan Data <i>Training</i> dan Data <i>Testing</i> Terhadap Akurasi	113
5.1.1 Skenario Pengujian Rasio Perbandingan.....	113
5.1.2 Analisis Pengujian Rasio Perbandingan	114
5.2 Pengujian Nilai <i>Threshold</i> AHP.....	115
5.2.1 Skenario Pengujian Nilai <i>Threshold</i> AHP.....	115
5.2.2 Analisis Pengujian Nilai <i>Threshold</i> AHP	116
5.3 Pengujian Terhadap Tingkat Akurasi dari Pengaruh Parameter Lamda	116
5.3.1 Skenario Pengujian Lamda	116
5.3.2 Analisis Pengujian Nilai Lamda.....	117
5.4 Pengujian Terhadap Konstanta Gamma	118
5.4.1 Skenario Pengujian Konstanta Gamma.....	118
5.4.2 Analisis Pengujian Konstanta Gamma.....	119
5.5 Pengujian Terhadap Nilai Epsilon	119
5.5.1 Skenario Pengujian Nilai Epsilon	119



5.5.2 Analisis Pengujian Nilai Epsilon.....	120
5.6 Pengujian Tingkat Akurasi Terhadap Iterasi Maksimum	121
5.6.1 Skenario Pengujian Iterasi Maksimum.....	121
5.6.2 Analisa Pengujian Iterasi Maksimum	122
5.7 Pengujian Terhadap Pengaruh Nilai C	122
5.7.1 Skenario Pengujian Terhadap Pengaruh Nilai C.....	122
5.7.2 Analisa Pengujian Pengaruh Nilai C.....	123
5.8 Analisis Tingkat Akurasi	124
BAB 6 PENUTUP	125
6.1 Kesimpulan.....	125
6.2 Saran	125
DAFTAR PUSTAKA.....	126
LAMPIRAN	128



DAFTAR TABEL

Tabel 2.1 Kajian Pustaka	6
Tabel 2.2 Kriteria Penerimaan Beasiswa.....	8
Tabel 2.3 Susunan Matriks Perbandingan Berpasangan.....	15
Tabel 2.4 Intensitas Kepentingan Perangkingan Akhir	15
Tabel 2.5 Ratio Index.....	17
Tabel 2.6 Metode One-against-all dengan 3 Kelas	23
Tabel 2.7 Metode One-against-one dengan 6 SVM Biner	23
Tabel 2.8 Confusion Matrix 2x2	26
Tabel 2.9 Confusion Matrix 3x3	26
Tabel 3.1 Bentuk Data Penelitian.....	29
Tabel 3.2 <i>Sample Dataset</i>	48
Tabel 3.3 Matriks Perbandingan Berpasangan	48
Tabel 3.4 Normalisasi Matrik	49
Tabel 3.5 Bobot Kriteria	50
Tabel 3.6 Hasil Perhitungan Nilai Ax	51
Tabel 3.7 Matrik Perbandingan Kriteria Penghasilan	51
Tabel 3.8 Hasil Normalisasi Sub Kriteria Penghasilan	52
Tabel 3.9 Matriks Perbandingan Kriteria Penghasilan.....	52
Tabel 3.10 Bobot Sub Kriteria Jumlah Tanggungan.....	53
Tabel 3.11 Hasil Normalisasi Sub Kriteria Jumlah Tanggungan	53
Tabel 3.12 Bobot Sub Kriteria Jumlah Tanggungan	53
Tabel 3.13 Matrik Perbandingan Kriteria Pekerjaan.....	54
Tabel 3.14 Hasil Normalisasi Sub Kriteria Pekerjaan	54
Tabel 3.15 Bobot Sub Kriteria Pekerjaan	54
Tabel 3.16 Matriks Perbandingan Kriteria Daya Listrik.....	55
Tabel 3.17 Hasil Normalisasi Sub Kriteria Daya Listrik.....	55
Tabel 3.18 Bobot Sub Kriteria Daya Listrik.....	56
Tabel 3.19 Matrik Perbandingan Kriteria Biaya Listrik.....	56
Tabel 3.20 Hasil Normalisasi Sub Kriteria Biaya Listrik	56
Tabel 3.21 Bobot Sub Kriteria Biaya Listrik	57
Tabel 3.22 Matriks Perbandingan Kriteria Jumlah Rumah	57

Tabel 3.23 Hasil Normalisasi Sub Kriteria Jumlah Rumah.....	58
Tabel 3.24 Bobot Sub Kriteria Jumlah Rumah.....	58
Tabel 3.25 Matriks Perbandingan Kriteria Status Rumah.....	58
Tabel 3.26 Hasil Normalisasi Sub Kriteria Status Rumah.....	59
Tabel 3.27 Bobot Sub Kriteria Status Rumah.....	59
Tabel 3.28 Matriks Perbandingan Kriteria Kondisi Dinding.....	59
Tabel 3.29 Hasil Normalisasi Sub Kriteria Kondisi Dinding.....	60
Tabel 3.30 Bobot Sub Kriteria Kondisi Dinding.....	60
Tabel 3.31 Matriks Perbandingan Kriteria Kondisi Lantai.....	60
Tabel 3.32 Hasil Normalisasi Sub Kriteria Kondisi Lantai.....	61
Tabel 3.33 Bobot Sub Kriteria Kondisi Lantai.....	61
Tabel 3.34 Matriks Perbandingan Kriteria PBB.....	61
Tabel 3.35 Hasil Normalisasi Sub Kriteria PBB.....	62
Tabel 3.36 Bobot Sub Kriteria PBB.....	62
Tabel 3.37 Matriks Perbandingan Kriteria Jumlah Motor.....	63
Tabel 3.38 Hasil Normalisasi Sub Kriteria Jumlah Motor.....	63
Tabel 3.39 Bobot Sub Kriteria Jumlah Motor.....	63
Tabel 3.40 Matrik Perbandingan Sub Kriteria Jumlah Mobil.....	64
Tabel 3.41 Normalisasi Sub Kriteria Jumlah Mobil.....	64
Tabel 3.42 Bobot Sub Kriteria Jumlah Mobil.....	65
Tabel 3.43 Matriks Perbandingan Sub Kriteria Program Beasiswa.....	65
Tabel 3.44 Hasil Normalisasi Sub Kriteria Program Beasiswa.....	65
Tabel 3.45 Bobot Sub Kriteria Program Beasiswa.....	66
Tabel 3.46 Nilai Bobot Seluruh Kriteria dan Sub Kriteria.....	66
Tabel 3.47 Kriteria Setelah Reduksi.....	66
Tabel 3.48 Hasil Normalisasi Data <i>Training</i>	67
Tabel 3.49 Hasil Normalisasi Data <i>Testing</i>	67
Tabel 3.50 Perkalian Bobot Sub Kriteria dengan Hasil Normalisasi Data <i>Training</i>	68
Tabel 3.51 Perkalian Bobot Sub Kriteria dengan Normalisasi Data <i>Testing</i>	68
Tabel 3.52 Hasil Perhitungan <i>Kernel Polynomial Degree d</i>	69
Tabel 3.53 Hasil Perhitungan Matriks Hessian.....	70
Tabel 3.54 Hasil Perhitungan Iterasi 1 Level 1.....	70



Tabel 3.55 Hasil Perhitungan Nilai Delta Alpha	71
Tabel 3.56 Hasil Perhitungan Nilai Alpha	71
Tabel 3.57 Hasil Perhitungan Iterasi 2 Level 1	71
Tabel 3.58 Hasil Perhitungan Nilai Delta Alpha Iterasi 2	72
Tabel 3.59 Hasil Perhitungan Nilai Alpha Iterasi 2	72
Tabel 3.60 Mencari Nilai kernel $x +$ dan $x -$	73
Tabel 3.61 Nilai Bobot Kelas Positif dan Negatif Data Training	74
Tabel 3.62 Hasil Perhitungan Data Uji 1 Level 1	76
Tabel 3.63 Hasil Perhitungan Data Uji 2 Level 1	76
Tabel 3.64 Hasil Perhitungan Data Uji 3 Level 1	76
Tabel 3.65 Hasil Perhitungan Nilai $f(x)$	77
Tabel 3.66 Normalisasi Data <i>Training</i> Level 2	77
Tabel 3.67 Normalisasi Data <i>Testing</i> Level 2	78
Tabel 3.68 Hasil Perkalian Normalisasi Data <i>Training</i> dengan Sub Bobot Kriteria Terpilih Level 2	78
Tabel 3.69 Hasil Perkalian Normalisasi Data <i>Testing</i> dengan Sub Bobot Kriteria Terpilih Level 2	79
Tabel 3.70 Hasil Perhitungan Kernel <i>Polynomial Degree</i> Level 2	79
Tabel 3.71 Hasil Perhitungan Matriks <i>Hessian</i> Level 2	80
Tabel 3.72 Hasil Perhitungan Iterasi 1 Level 2	80
Tabel 3.73 Hasil Perhitungan Nilai Delta Alpha Level 2	81
Tabel 3.74 Hasil Perhitungan Nilai Alpha Level 2	81
Tabel 3.75 Hasil Perhitungan Iterasi 2 Level 2	81
Tabel 3.76 Hasil Perhitungan Nilai Delta Alpha Iterasi 2 Level 2	82
Tabel 3.77 Hasil Perhitungan Nilai Alpha Iterasi 2 Level 2	82
Tabel 3.78 Mencari Nilai Kernel x^+ dan x^-	83
Tabel 3.79 Nilai Bobot Kelas Positif dan Negatif Level 2	84
Tabel 3.80 Hasil Pengujian Data Uji 1 Level 2	85
Tabel 3.81 Hasil Perhitungan Data Uji 2 Level 2	85
Tabel 3.82 Hasil Perhitungan $f(x)$ dan Klasifikasi	86
Tabel 3.83 Model Evaluasi <i>Confusion Matriks</i>	86
Tabel 3.84 Perancangan Pengujian Perbandingan Jumlah Data <i>Training</i> dan Data <i>Testing</i>	88



Tabel 3.85 Skenario Pengujian Berdasarkan Nilai <i>Threshold</i> AHP	89
Tabel 3.86 Skenario Pengujian Berdasarkan Nilai Parameter <i>Lamda</i>	89
Tabel 3.87 Skenario Pengujian Nilai Konstanta Gamma.....	90
Tabel 3.88 Skenario Pengujian Terhadap Pengaruh Nilai <i>Epsilon</i>	90
Tabel 3.89 Skenario Pengujian Iterasi Maksimum	91
Tabel 3.90 Skenario Pengujian terhadap pengaruh nilai C terhadap akurasi.....	92
Tabel 4.1 Implementasi Perangkat Keras.....	94
Tabel 4.2 Lingkup Implementasi Perangkat Lunak	94
Tabel 5.1 Hasil Pengujian Rasio Data <i>Training</i> dan Data <i>Testing</i>	114
Tabel 5.2 Hasil Pengujian Nilai <i>Threshold</i> AHP	115
Tabel 5.3 Hasil Pengujian Nilai Lamda	117
Tabel 5.4 Hasil Pengujian Nilai Konstanta Gamma	118
Tabel 5.5 Hasil Pengujian Nilai Epsilon.....	120
Tabel 5.7 Hasil Pengujian Terhadap Pengaruh Nilai C.....	123



DAFTAR GAMBAR

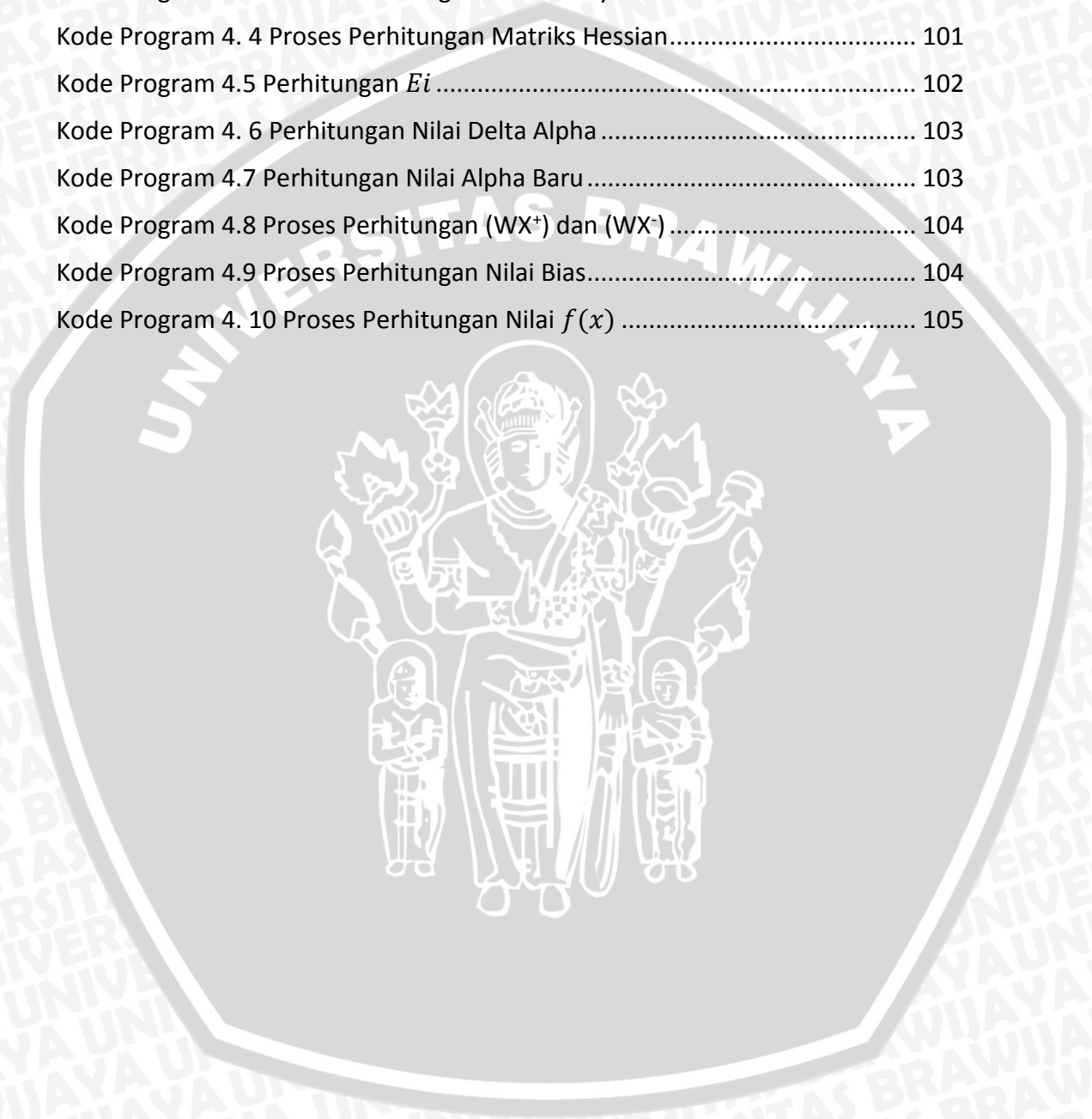
Gambar 2.1 Skematik DSS	10
Gambar 2.2 Fase-fase Proses Pengambilan Keputusan	13
Gambar 2.3 Konsep Dasar Penerapan Metode SVM	19
Gambar 2.4 Contoh Klasifikasi <i>One-against-all</i> untuk 3 kelas	23
Gambar 2.5 Klasifikasi Menggunakan Metode <i>One-Against-One</i>	24
Gambar 3.1 Diagram Alir Pelaksanaan Penelitian	28
Gambar 3.2 Diagram Alir Klasifikasi Penerima Beasiswa	31
Gambar 3.3 Diagram Alir Perhitungan Bobot AHP	32
Gambar 3.4 Diagram Alir Normalisasi AHP	33
Gambar 3.5 Diagram Alir Hitung Jumlah Perkolom Matrik Perbandingan	34
Gambar 3.6 Diagram Alir Seleksi fitur AHP	35
Gambar 3.7 Diagram Alir Normalisasi Dataset	36
Gambar 3.8 Diagram Alir Penghitungan Metode SVM	37
Gambar 3.9 Diagram Alir Perkalian Dengan Bobot Sub Kriteria Terpilih	38
Gambar 3.10 Diagram Alir Perhitungan <i>Kernel</i> SVM	39
Gambar 3.11 Diagram Alir Proses <i>Sequential Training</i> SVM	40
Gambar 3.12 Diagram Alir Perhitungan Matrix Hessian	41
Gambar 3.13 Diagram Alir Proses Perhitungan Nilai E_i	42
Gambar 3.14 Diagram Alir Perhitungan Maximum Delta Alpha	43
Gambar 3.15 Diagram Alir Proses <i>Testing</i>	44
Gambar 3.16 Proses Perhitungan Nilai Bias	45
Gambar 3.17 Proses Perhitungan Nilai $f(x)$	46
Gambar 3.18 Proses Perhitungan One-Against-All	47
Gambar 3.19 Perancangan Antarmuka Sistem	87
Gambar 4.1 Diagram Implementasi Sistem	93
Gambar 4.2 Antarmuka Konfigurasi Data Alternatif	106
Gambar 4.3 Antarmuka Konfigurasi Matriks Perbandingan	107
Gambar 4.4 Antarmuka Load Data Matrik Kriteria	107
Gambar 4.5 Hasil Perhitungan Sub Bobot Kriteria AHP	108
Gambar 4.6 Hasil Akhir Perhitungan AHP	108
Gambar 4.7 Load Data SVM	109

Gambar 4.8 Normalisasi Dataset SVM	109
Gambar 4.9 Antarmuka Perkalian Bobot Setelah Normalisasi	110
Gambar 4.10 Antarmuka Proses <i>Sequential Training</i>	110
Gambar 4.11 Hasil Perhitungan E_i Level 1	111
Gambar 4.12 Hasil Perhitungan E_i Level 2	111
Gambar 4.13 Antarmuka Hasil Perhitungan $f(x)$	112
Gambar 5.1 Diagram Pengujian dan Analisis	113
Gambar 5.2 Grafik Tingkat Akurasi Hasil Pengujian Rasio Perbandingan	114
Gambar 5.3 Grafik Pengujian Nilai Threshold	116
Gambar 5.4 Grafik Akurasi Terhadap Nilai Lamda	117
Gambar 5.5 Grafik Akurasi Terhadap Nilai Konstanta Gamma	119
Gambar 5.6 Grafik Akurasi Terhadap Nilai Epsilon	120
Gambar 5.7 Grafik Tingkat Akurasi Hasil Pengujian Iterasi Maksimum	122
Gambar 5.8 Grafik Akurasi Terhadap Pengaruh Nilai C	123



DAFTAR KODE PROGRAM

Kode Program 4.1 Proses Perhitungan AHP	97
Kode Program 4.2 Proses Normalisasi	99
Kode Program 4.3 Proses Perhitungan Kernel Polynomial	100
Kode Program 4. 4 Proses Perhitungan Matriks Hessian	101
Kode Program 4.5 Perhitungan Ei	102
Kode Program 4. 6 Perhitungan Nilai Delta Alpha	103
Kode Program 4.7 Perhitungan Nilai Alpha Baru	103
Kode Program 4.8 Proses Perhitungan (WX^+) dan (WX^-)	104
Kode Program 4.9 Proses Perhitungan Nilai Bias	104
Kode Program 4. 10 Proses Perhitungan Nilai $f(x)$	105



DAFTAR LAMPIRAN

Lampiran A Surat Rekomendasi Penelitian.....	128
Lampiran B Surat Persetujuan Menjadi Pakar	129
Lampiran C Formulir Pendaftaran Beasiswa.....	130
Lampiran D Inisialisasi Data	132
Lampiran E Nilai Perbandingan Kriteria Utama	133
Lampiran F Matrik Perbandingan Berpasangan Kriteria Utama.....	137
Lampiran G Nilai Perbandingan Sub Kriteria.....	138
Lampiran H Matrik Perbandingan Sub Kriteria	144
Lampiran I Struktur Hierarki AHP.....	147
Lampiran J Data Penerima Beasiswa GNOTA Kediri 2015	148
Lampiran K Dataset	156
Lampiran L Normalisasi Data Level 1	166
Lampiran M Perkalian Bobot Sub Kriteria Dengan Hasil Normalisasi Level 1.....	167
Lampiran N Perhitungan Kernel Polynomial Degree 2 Level 1	168
Lampiran O Normalisasi Data Level 2	169
Lampiran P Perkalian Bobot Sub Kriteria Dengan Hasil Normalisasi Level 2.....	170
Lampiran Q Perhitungan Kernel Polynomial Degree 2 Level 2.....	171
Lampiran R Hasil Pengujian Berdasarkan Nilai Threshold AHP.....	172
Lampiran S Hasil Pengujian	173



BAB 1 PENDAHULUAN

1.1 Latar belakang

Pendidikan merupakan kebutuhan pokok bagi seluruh masyarakat Indonesia. Manfaat pendidikan bagi masyarakat Indonesia sangat beragam seperti untuk karir atau pekerjaan. Disamping memberikan pengetahuan, pendidikan juga mampu membantu dalam kemajuan suatu bangsa dan menjadikan manusia lebih baik dan memiliki karakter. Undang-undang nomor 20 Tahun 2003 tentang sistem pendidikan nasional mengamanatkan bahwa pendidikan nasional harus mampu menjamin pemerataan kesempatan pendidikan dan peningkatan mutu serta relevansi pendidikan untuk menghadapi tantangan perubahan kehidupan lokal, nasional dan global. Seiring perkembangan zaman, diperlukan perbaikan dari sistem yang lama ke dalam bentuk sistem yang lebih baik lagi yang memiliki efisiensi dan efektifitas dalam sistem tersebut. Salah satu yang banyak dibutuhkan saat ini adalah sebuah sistem yang mampu membantu dalam proses pengambilan keputusan. Sistem pendukung keputusan berperan sebagai sistem yang dimaksudkan untuk mendukung para pengambil keputusan manajerial dalam situasi keputusan semi terstruktur (Turban, Aronson dan Liang, 2005).

Dalam penelitian ini diambil kasus pemilihan penerima beasiswa pada Lembaga GNOTA (Gerakan Nasional Orang Tua Asuh) Kediri sehingga nantinya kita dapat melakukan penyeleksian secara *realtime*, cepat dan akurat terhadap penerima beasiswa dengan suatu sistem penyeleksian yang mampu melakukan filterisasi secara otomatis. Lembaga GNOTA merupakan sebuah organisasi nirlaba, independen dan transparan. Lembaga ini merupakan gerakan inisiatif dari masyarakat untuk menjaga agar anak Indonesia mendapatkan pendidikan yang layak untuk masa depan yang lebih baik. GNOTA mendistribusikan paket bantuan pendidikan untuk membantu keluarga anak-anak kurang mampu agar mereka dapat menuntaskan pendidikannya.

Beberapa proses diperlukan dalam pengambilan keputusan untuk melakukan seleksi terhadap penerima beasiswa. Pada studi kasus Lembaga GNOTA Kediri, proses pengambilan keputusan penentuan penerima beasiswa selama ini dilakukan secara manual. Manual disini maksudnya adalah data pemohon beasiswa direkapitulasi oleh panitia penyeleksi, lalu panitia melakukan survei satu persatu terhadap data yang diajukan oleh pemohon. Setelah proses survei dilakukan maka panitia akan meninjau dan menganalisa kembali data pemohon beasiswa tersebut. Proses ini membutuhkan waktu yang lama, kurang lebih tiga bulan hanya untuk meninjau dan menganalisa data pemohon beasiswa yang cukup banyak. Cara ini menyebabkan penilaian terhadap penerima beasiswa dilakukan secara subjektif, yang berarti penilaian dari sudut pandang panitia survei saja. Selain itu permasalahan ketepatan penyaluran beasiswa GNOTA yang selama ini sering terjadi karena dalam

penentuan penerima beasiswa belum memiliki bobot secara pasti pada setiap kriteria dan sub kriteria yang dimiliki oleh pemohon beasiswa. Dengan demikian dibutuhkan sebuah sistem berbasis komputer sebagai alternatif solusi yang dapat membantu lembaga GNOTA menentukan pemohon beasiswa yang layak menerima beasiswa dari lembaga GNOTA Kediri agar beasiswa dapat disalurkan dengan sukses dan tepat sasaran. Dengan adanya aplikasi tersebut dapat memberikan informasi yang efektif dan efisien dalam pengambilan suatu keputusan untuk penerima beasiswa. Metode yang digunakan adalah metode AHP (*Analytical Hierarchy Process*) dan SVM (*Support Vector Machine*). Menurut Burgeois (2005) AHP umumnya digunakan dengan tujuan untuk menyusun prioritas dari berbagai alternatif atau pilihan yang ada, dan pilihan-pilihan tersebut bersifat kompleks atau multi kriteria.

Pada penelitian sebelumnya, Jian Liang Peng (2012) menggunakan metode AHP (*Analytical Hierarchy Process*) untuk pemilihan pemasok layanan logistik. Metode tersebut diangkat karena bisa memberikan nilai preferensi untuk setiap supplier. Pada penelitian tersebut menggunakan empat parameter penilaian seperti *cost*, *operating efficiency*, *service quality* dan *technology level*. Mohanty dan Bebartta (2011) melakukan penelitian yang membahas mengenai perbandingan klasifikasi antara metode KNN dan SVM untuk pengenalan karakter karakter bahasa Oriya (bahasa di negara bagian India) dimana pada penelitian tersebut menghasilkan akurasi 96.47% untuk metode KNN dan 98.9% dengan metode SVM, sehingga dapat diketahui klasifikasi dengan metode SVM lebih tinggi dibandingkan dengan metode KNN. Dr. S. Vijayarani (2015) menggunakan metode SVM dan Naive Bayes untuk prediksi penyakit liver. Kedua algoritma tersebut dibandingkan dan didasarkan pada akurasi kinerja faktor klasifikasi dan waktu pelaksanaan. Dari hasil percobaan klasifikasi dengan SVM dianggap yang terbaik karena memiliki akurasi tinggi dalam hal klasifikasi. Sedangkan *Naive Bayes* dianggap hanya unggul dalam waktu pelaksanaan saja. Yi Yang et al (2010) menggunakan metode AHP dan SVM untuk seleksi tempat pergantian gardu karena metode tersebut saling melengkapi untuk melakukan seleksi, dimana metode AHP kesulitan untuk memberikan klasifikasi kelayakan tempat pergantian gardu sehingga metode SVM digunakan sebagai proses awal untuk melakukan klasifikasi kelayakan tempat pergantian gardu.

Dari beberapa penelitian sebelumnya di atas, maka dalam skripsi ini diajukan judul "Penerapan Metode AHP dan SVM untuk Klasifikasi Penerima Beasiswa (Studi Kasus : Lembaga GNOTA Kediri)". Diharapkan dalam pemilihan penerima beasiswa digunakan metode AHP dan SVM untuk dapat menghasilkan *output* yang sesuai dengan keadaan atau kriteria yang ada sehingga dapat memudahkan Lembaga GNOTA dalam melakukan pengambilan keputusan dan meminimalkan kesalahan dalam pengambilan kebijakan.

1.2 Rumusan masalah

Rumusan masalah dalam penelitian ini adalah sebagai berikut :

1. Bagaimana implementasi dari metode AHP dan SVM untuk memberikan rekomendasi penerima beasiswa?
2. Bagaimana tingkat akurasi yang dihasilkan oleh sistem dari hasil implementasi metode AHP dan SVM?

1.3 Tujuan

Berdasarkan perumusan masalah yang telah diuraikan sebelumnya maka tujuan yang ingin dicapai dalam penelitian ini adalah :

1. mengimplementasikan metode AHP dan SVM untuk klasifikasi yang memberikan rekomendasi dalam pemilihan penerima beasiswa.
2. Mengukur tingkat akurasi dari implementasi metode AHP dan SVM untuk klasifikasi penerima beasiswa pada Lembaga GNOTA.

1.4 Manfaat

Manfaat yang diharapkan akan didapat dari penelitian ini adalah :

1. Sebagai salah satu alternatif dalam membantu melakukan pemilihan penerima beasiswa GNOTA Kediri.
2. Mempercepat proses seleksi penerima beasiswa dengan meningkatkan efisiensi waktu dan tenaga yang lebih baik.

1.5 Batasan masalah

Adapun batasan masalah pada penelitian ini adalah :

1. Pemilihan penerima beasiswa pada lembaga GNOTA Kediri berdasarkan 13 parameter yakni penghasilan orang tua, jumlah tanggungan, pekerjaan, daya listrik, biaya listrik, jumlah rumah, status rumah, status dinding, status lantai, PBB, jumlah motor, jumlah mobil dan jalur masuk PTN.
2. Jenis beasiswa yang diteliti adalah beasiswa alih jenjang dari SMA (Sekolah Menengah Atas) menuju PTN (Perguruan Tinggi Negeri).
3. Penelitian ini menggunakan 111 data uji dari penerima beasiswa GNOTA Kediri tahun 2015.

1.6 Sistematika pembahasan

Pembuatan penelitian ini dikerjakan dengan sistematika pembahasan sebagai berikut :

BAB I PENDAHULUAN

Berisi latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat dan sistematika pembahasan.

BAB II LANDASAN KEPUSTAKAAN

Bab ini membahas mengenai kajian pustaka jika ada penelitian sebelumnya yang digunakan sebagai acuan dalam mengerjakan penelitian penerapan metode AHP dan SVM pada klasifikasi penerima beasiswa lembaga GNOTA Kediri.

BAB III METODOLOGI DAN PERANCANGAN

Menguraikan tentang metode dan langkah kerja yang dilakukan dalam penelitian ini, perancangan sistem serta perhitungan manual.

BAB IV IMPLEMENTASI

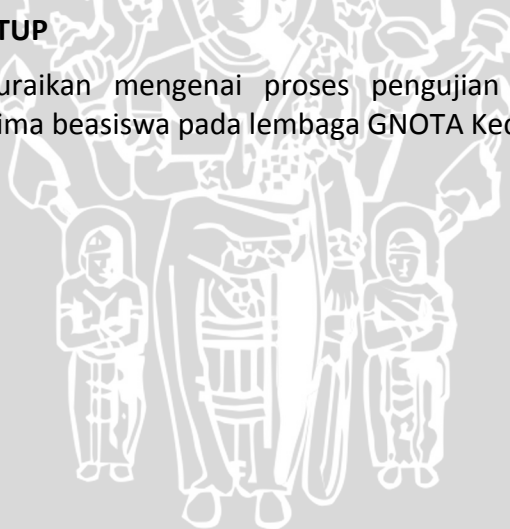
Membahas mengenai proses implementasi dari metode AHP dan SVM dalam melakukan klasifikasi penerima beasiswa pada Lembaga GNOTA Kediri.

BAB V PENGUJIAN DAN ANALISIS

Menguraikan tentang proses pengujian sistem klasifikasi penerima beasiswa. Proses yang diujikan adalah mengukur tingkat akurasi dari sistem.

BAB VI PENUTUP

Menguraikan mengenai proses pengujian sistem klasifikasi penerima beasiswa pada lembaga GNOTA Kediri.



BAB 2 LANDASAN KEPUSTAKAAN

Bab ini membahas tentang kajian pustaka dan dasar teori yang mendukung dalam penulisan skripsi mengenai Penerapan Metode AHP dan SVM untuk Klasifikasi Penerima Beasiswa (Studi Kasus : Lembaga GNOTA Kediri). Beberapa referensi diantaranya adalah Sistem Pendukung Keputusan, Klasifikasi, Beasiswa, *Analytical Hierarchy Process (AHP)*, *Support Vector Machine (SVM)*.

2.1 Kajian Pustaka

Sebagai bahan referensi dalam penelitian Penerapan Metode AHP dan SVM untuk Klasifikasi Penerima Beasiswa (Studi Kasus Lembaga GNOTA Kediri), berikut beberapa penelitian yang sudah ada menggunakan metode AHP dan SVM. Pada penelitian pertama membahas mengenai pemilihan pemasok layanan logistik. Pada penelitian tersebut permasalahan yang diangkat mengenai persaingan perusahaan yang semakin sengit diiringi dengan pesatnya kemajuan Teknologi Informasi sehingga memicu perusahaan harus lebih selektif dalam melakukan pemilihan pemasok layanan logistik. Akan tetapi, kurangnya metode yang efektif untuk memilih pemasok layanan logistik yang benar-benar sesuai untuk pengembangan usaha menyebabkan kegagalan logistik *outsourcing*. Metode AHP dipilih untuk penelitian di atas karena AHP dinilai lebih sesuai dalam hal evaluasi, disisi lain AHP menyediakan referensi rekomendasi bagi perusahaan untuk memilih pemasok layanan logistik yang sesuai.

Pada penelitian kedua yang dilakukan oleh Mohanty dan Bebartha (2011) membahas mengenai perbandingan metode SVM dan KNN untuk pengenalan karakter bahasa Oriya (bahasa dinegara bagian barat India). Pada penelitian tersebut menghasilkan nilai akurasi dengan metode SVM lebih tinggi jika dibandingkan dengan metode KNN yakni sebesar 98.9% untuk SVM dan 96.47% untuk KNN.

Pada penelitian ketiga membahas mengenai pemilihan letak transmisi dan stasiun transformasi pada negara bagian barat Mongolia. Dalam melakukan pemilihan letak transmisi dan stasiun transformasi, terdapat beberapa variabel yang digunakan mulai dari indeks situasi geologi dan geografi, indeks *engineering*, indeks faktor konstruksi, indeks ekonomi dan faktor lainnya yang menghasilkan letak transmisi dan stasiun transformasi dari Baotou ke Hohhot yang dikategorikan ideal dan non-ideal. Penelitian ini menghasilkan nilai akurasi sebesar 80%.

Pada penelitian keempat yang dilakukan oleh Dr. S. Vijayarani dan Mr. S. Dhayanand (2015) membahas mengenai prediksi penyakit hati menggunakan metode SVM dan *Naive Bayes*. Perbandingan algoritma ini dilakukan dan didasarkan pada akurasi kinerja faktor klasifikasi dan waktu pelaksanaan.

Berikut kajian psutaka yang menjelaskan beberapa penelitian yang telah dilakukan terkait dengan klasifikasi pada sistem pendukung keputusan yang ditunjukkan pada Tabel 2.1.

Tabel 2.1 Kajian Pustaka

No	Judul	Metode dan Objek	Hasil
1	<i>Selection of Logistics Outsourcing Serice Suppliers Based on AHP</i>	<p>Metode : AHP</p> <p>Objek : Nilai dari harga, <i>opeating</i>, efisiensi, teknologi yang digunakan.</p>	Nilai preferensi untuk setiap <i>supplier</i> (Nilai preferensi yang lebih besar menunjukkan bahwa suplier tersebut yang terpilih atau yang akan direkomendasikan menjadi pilihan terbaik).
2	<i>Peformance Comparison of SVM and KNN for Oriya Character Recognition.</i>	<p>Metode : SVM & KNN</p> <p>Objek : Tipe <i>image</i> dan <i>size</i> dari sampel seperti <i>Bold and Small</i>, <i>Bold and Big</i>, <i>Normal and Small</i>, <i>Normal and Bold</i>.</p>	Perbandingan metode SVM dan KNN dilakukan untuk pengenalan karakter bahasa Oriya dimana nilai akurasi yang dihasilkan oleh metode SVM sebesar 98.9% sedangkan metode KNN sebesar 96.47%,
3	<i>The Application of Sites Selection Based on AHP-SVM in 500KV Substation</i>	<p>Metode : AHP & SVM</p> <p>Objek : Evaluasi indeks yang komperhensif untuk pemilihan letak transmisi dan stasiun transformasi di negara bagian barat Mongolia.</p> <p>Parameter :</p> <ul style="list-style-type: none"> - <i>Geological and geography</i> - <i>Engineering indexes</i> - <i>Construction</i> 	<ul style="list-style-type: none"> - Data <i>sample</i> yang digunakan diklasifikasi dalam beberapa grup. - Pada penelitian ini metode AHP digunakan untuk penentuan bobot dengan 20 kriteria - Hasil dari klasifikasi pada setiap grup akan dihitung kembali untuk dievaluasi - Metode yang digunakan (AHP-SVM) memiliki efek lebih baik dari

		<p><i>factor indexes</i></p> <ul style="list-style-type: none"> - <i>Indexes economic</i> 	<p>klasifikasi dan dapat dipadukan untuk pemilihan gardu.</p> <ul style="list-style-type: none"> - Tingkat akurasi yang dihasilkan dari penerapan metode AHP-SVM ini sebesar 80%.
4	<p><i>Liver Disease Prediction using SVM and Naive Bayes Algorithm</i></p>	<p>Metode :</p> <ol style="list-style-type: none"> 1. SVM 2. Naive Bayes <p>Objek :</p> <p>Prediksi penyakit liver menggunakan metode SVM dan Naive Bayes</p>	<ul style="list-style-type: none"> - Perbandingan algoritma ini dilakukan dan didasarkan pada akurasi kinerja faktor klasifikasi dan waktu pelaksanaan. Dari hasil percobaan klasifikasi dengan SVM dianggap sebagai yang terbaik karena memiliki akurasi tinggi dalam klasifikasi yakni sebesar 80% sedangkan <i>Naive Bayes</i> hanya menghasilkan nilai akurasi sebesar 60%. - Untuk waktu eksekusi, <i>naive bayes</i> membutuhkan waktu minimum yakni 1670ms sedangkan SVM membutuhkan waktu 3210ms.
5	<p>Penerapan Metode AHP dan SVM untuk Klasifikasi Penerima Beasiswa (Studi Kasus : Lembaga GNOTA Kediri)</p>	<p>Metode :</p> <ol style="list-style-type: none"> 1. AHP 2. SVM <p>Objek :</p> <p>Data pemohon beasiswa lembaga GNOTA Kediri</p>	<ul style="list-style-type: none"> - Hasil penelitian ini menggunakan metode AHP dan SVM yang akan memberikan hasil klasifikasi penerima beasiswa pada Lembaga GNOTA Kediri. Pembobotan

	Tahun 2015.	dalam penelitian ini menggunakan metode AHP sedangkan untuk klasifikasi penerima beasiswa digunakan metode SVM.
--	-------------	---

Sumber : (Zhejiang Gongshang, 2012), (Mohanty dan Bebartta, 2011), (Yang-yi *et all* 2010) dan (Dr. S. Vijayarani dan Mr. S. Dhayanand, 2015)

2.2 Beasiswa

Beasiswa merupakan pendanaan yang diberikan oleh pemerintah, perusahaan swasta, kedutaan, universitas serta lembaga pendidik atau peneliti, atau dari kantor tempat bekerja seseorang. Beasiswa diberikan kepada yang berhak menerima berdasarkan klasifikasi, kualitas dan kompetensi penerima beasiswa (Ghafur, Abdul, 2008). Seperti halnya pada Lembaga GNOTA Kediri yang memiliki program pemberian beasiswa terhadap mahasiswa yang baru masuk ke Perguruan Tinggi Negeri (PTN). Disamping banyaknya pemohon beasiswa, banyaknya kriteria yang digunakan untuk proses seleksi tentu mempersulit tahap pemilihan penerima beasiswa.

Berdasarkan uraian di atas dapat disimpulkan bahwa pemberian beasiswa dilakukan berdasarkan proses seleksi agar beasiswa yang diberikan tepat sasaran dan diterima oleh pihak yang tepat. Dalam penelitian ini, kriteria-kriteria yang menjadi dasar Lembaga GNOTA Kediri mengambil keputusan pada pemberian beasiswa ada 13 kriteria seperti pada Tabel 2.2.

Tabel 2.2 Kriteria Penerimaan Beasiswa

No	Jenis Kriteria	Keterangan
1	Penghasilan orang tua/wali	Merupakan jumlah keseluruhan penghasilan orang tua/wali perbulannya
2	Jumlah tanggungan	Merupakan jumlah tanggungan yang masih dimiliki oleh orang tua/wali
3	Pekerjaan orang tua/wali	Merupakan jenis pekerjaan orang tua/wali
4	Daya listrik	Merupakan besarnya daya listrik yang digunakan
5	Biaya listrik	Merupakan biaya listrik bulan terakhir
6	Jumlah rumah	Merupakan jumlah rumah yang dimiliki oleh orang tua/wali
7	Status rumah	Kondisi status rumah yang ditempati

8	Kondisi dinding	Kondisi dinding rumah yang ditempati
9	Kondisi lantai	Kondisi lantai rumah yang ditempati
10	PBB	Besarnya pajak bumi dan bangunan terakhir
11	Jumlah motor	Jumlah motor yang dimiliki
12	Jumlah mobil	Jumlah mobil yang dimiliki
13	Program Beasiswa	Merupakan jalur masuk ke PTN

2.3 Gerakan Nasional Orang Tua Asuh (GNOTA)

Gerakan Nasional Orang Tua Asuh (GNOTA) merupakan suatu gerakan inisiatif masyarakat yang berfungsi untuk menjaga agar anak-anak Indonesia mendapatkan pendidikan dari sekolah dasar yang menjadi landasan dalam meraih masa depan yang lebih baik lagi. GNOTA memberikan bantuan terhadap masyarakat dengan mendistribusikan bantuan berupa donasi uang dan pendidikan untuk membantu anak-anak yang kurang mampu agar dapat melanjutkan pendidikan secara berkala. Pada Lembaga GNOTA Kediri, sebelum memberikan beasiswa mereka melakukan sosialisasi ke sekolah-sekolah yang ada di Kabupaten Kediri mengenai program beasiswa yang akan dibuka. Selanjutnya pemohon beasiswa mengirimkan berkas persyaratan pengajuan beasiswa ke Lembaga GNOTA Kediri dan akan dilakukan seleksi terhadap pemohon beasiswa tersebut. GNOTA Kab. Kediri membatasi penerima beasiswa dengan hanya menampung pemohon beasiswa yang berdomisili di Kabupaten Kediri. Beberapa program beasiswa yang diberikan yakni beasiswa alih jenjang dari SD ke SMP, beasiswa alih jenjang dari SMP ke SMA, dan beasiswa alih jenjang dari SMA ke Perguruan Tinggi Negeri (PTN). Dengan adanya GNOTA dapat membantu anak-anak Indonesia untuk melanjutkan pendidikannya ke jenjang yang lebih tinggi (GNOTA, 2015).

2.4 Sistem Pendukung Keputusan

Sistem Pendukung Keputusan merupakan sebuah sistem yang dimaksudkan untuk mendukung para pengambil keputusan manajerial dalam situasi keputusan semi terstruktur (Efraim, Aronson & Liang, 2005). Tujuan dari sistem pendukung keputusan yaitu agar memperluas kemampuan pengambil keputusan dalam proses pengambilan data atau informasi untuk mengambil keputusan serta menghasilkan solusi yang cepat, sehingga nantinya dapat digunakan untuk memperkuat pengambilan keputusan terhadap keputusan yang akan diambil.

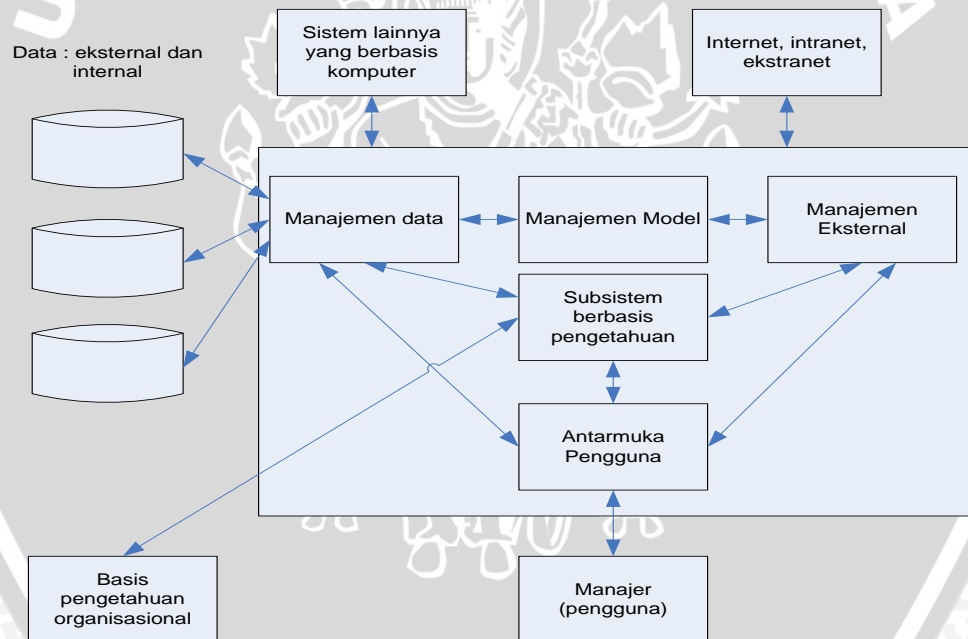
Sistem pendukung keputusan juga mampu menyelesaikan masalah dan keadaan yang dihadapi sehingga dapat mempercepat waktu, SDM, dan biaya yang dibutuhkan dalam memecahkan masalah terutama dalam berbagai permasalahan yang sangat kompleks. Sistem ini juga membantu dalam

merumuskan masalah dan keadaan yang dihadapi sehingga mampu menghemat waktu, tenaga dan biaya yang dibutuhkan untuk memecahkan suatu permasalahan terutama dalam berbagai permasalahan yang sangat kompleks.

2.4.1 Konsep Dasar Sistem Pendukung Keputusan

Konsep Sistem Pendukung Keputusan (*Decision Support System*) pertama kali menjadi area riset pada tahun 1970-an oleh Michael Scott Morton yang kemudian dikenal dengan istilah "*Management Decision System*". DSS adalah sistem yang interaktif berjalan pada sistem komputer untuk membantu permasalahan yang bersifat tidak terstruktur dan semi terstruktur, setiap permasalahan yang dapat diselesaikan dengan langkah strategis, sehingga memperoleh hasil yang maksimum yaitu keuntungan yang maksimum dan yang minimum. Sistem ini bukanlah sebagai pengganti dalam pengambilan keputusan yang sebenarnya, melainkan hanya sebagai alat bantu untuk menyelesaikan persoalan dalam pengambilan keputusan.

Sistem Pendukung Keputusan dibangun dari beberapa subsistem dan memiliki beberapa komponen SPK seperti pada Gambar 2.1.



Gambar 2.1 Skematik DSS

Sumber : Turban, Aronson & Liang (2005)

1. Subsistem Manajemen Data

Subsistem manajemen data yaitu subsistem manajemen yang memasukkan data dalam suatu database yang berisikan data yang relevan pada situasi dan yang dikelola oleh piranti lunak disebut juga sebagai database manajemen sistem (DBMS). Subsistem manajemen

data dapat saling terhubung dengan *warehouse* suatu perusahaan, pada repositori data perusahaan yang relevan dalam pengambilan keputusan.

2. Subsistem Manajemen Model

Subsistem manajemen model merupakan perangkat lunak yang mengatur model, statistik, ilmu manajemen, atau suatu model kuantitatif lainnya yang memberikan kapabilitas analisis dalam manajemen perangkat lunak secara tepat, bahasa dalam pemodelan untuk membangun suatu model-model secara sesuai keinginan juga dimasukkan. Perangkat lunak ini juga sering disebut sebagai sistem manajemen basis model (MBMS), yang mana komponen ini dapat dikoneksikan dalam penyimpanan suatu perusahaan internal maupun eksternal yang terdapat pada model. Sistem manajemen dan model metode soulusi bisa diterapkan pada sistem berbasis web, java yang berjalan pada sebuah server.

3. Subsistem antarmuka pengguna

Subsistem antarmuka pengguna dapat berkomunikasi dengan cara memerintahkan DSS melalui subsistem antarmuka pengguna. Pengguna merupakan dari bagian yang bisa dipertimbangkan dari sistem yang dibalut. Adapun sistem lainnya yang berbasis data komputer yaitu terdiri dari data eksternal dan data internal, penggunaan antarmuka menggunakan basis pengetahuan dari suatu organisasi dari para peneliti yang menegaskan bahwa beberapa kontribusi untuk dari DSS yang berasal dari interaksi yang sensitif antar muka pengguna dengan grafis yang sangat familiar dan konsisten bagi kebanyakan DSS lainnya.

4. Subsistem manajemen berbasis pengetahuan

Subsistem manajemen berbasis pengetahuan dapat mendukung semua subsistem lainnya atau bertindak secara dari suatu komponen yang independen, yang memberikan kecerdasan untuk memperbesar pengetahuan dari sistem pengambil keputusan. Subsistem ini juga saling terhubung dengan penyimpanan dari perusahaan yang kadang-kadang disebut sebagai dasar dari pengetahuan suatu organisasi. Banyak metode yang digunakan dalam kecerdasan buatan yang bisa diimplementasikan dalam sistem yang berbasis pengembangan web dan java serta mudah untuk terintegrasi dengan komponen DSS lainnya.

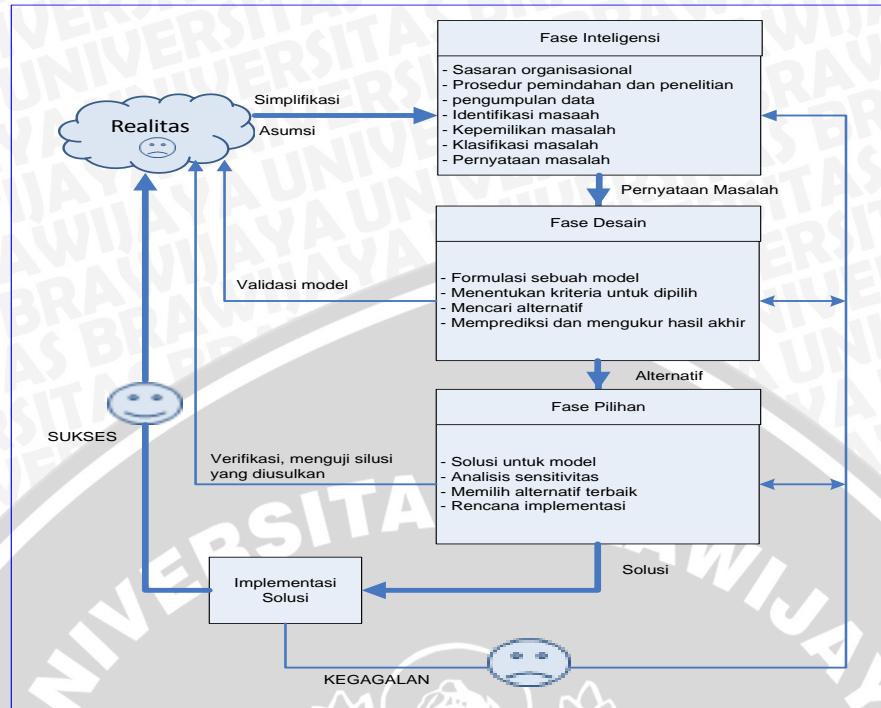
2.4.2 Karakteristik dan Kemampuan Sistem Pendukung Keputusan

Menurut Turban et al (2005) terdapat beberapa karakteristik dan kapabilitas kunci dari DSS meliputi :

1. Dukungan untuk pengambil keputusan, terutama pada situasi semi terstruktur dan tak terstruktur, dengan menyertakan penilaian manusia dan informasi terkomputerisasi. Masalah tersebut tidak dapat dipecahkan oleh sistem komputer lain atau oleh metode atau alat kuantitatif standar.
2. Dukungan untuk individu dan kelompok. Masalah yang kurang terstruktur sering memerlukan keterlibatan individu dari departemen dan tingkat organisasional yang berbeda bahkan dari organisasi lain.
3. Dukungan untuk keputusan independen dan atau sekuensial. Keputusan dapat dibuat satu kali beberapa kali atau berulang (dalam interval yang sama).
4. Mendukung seluruh fase dalam proses pengambilan keputusan yang meliputi inteligensi, desain, pilihan dan implementasi.
5. Mendukung berbagai proses dan model pengambilan keputusan.
6. Keefektifan pengambilan keputusan yang meningkat seperti akurasi, *timeliness*, dan kualitas) daripada efisiensinya.
7. Kontrol penuh oleh pengambil keputusan terhadap semua langkah proses pengambilan keputusan dalam memecahkan suatu masalah.
8. Sistem sederhana dapat dikembangkan dan dimodifikasi sendiri oleh pengguna akhir.
9. Biasanya model-model digunakan untuk menganalisis situasi pengambilan keputusan.
10. Terdapat berbagai akses yang disediakan untuk berbagai sumber data, seperti format dan tipe mulai dari sistem informasi geografis (SIG) maupun sistem berorientasi objek.
11. Berperan sebagai alat *stand alone* yang dapat digunakan oleh seorang pengambil keputusan pada satu lokasi atau didistribusikan di satu organisasi keseluruhan dan di beberapa organisasi sepanjang rantai persediaan.

2.4.3 Fase-fase Proses Pengambilan Keputusan

Menurut Simon (1977) mengatakan bahwa “proses pengambilan keputusan meliputi tiga fase utama yakni inteligensi, desain dan kriteria”. Kemudian ditambahkan fase yang keempat yaitu implementasi. Monitoring dapat dianggap sebagai fase kelima bentuk umpan balik. Akan tetapi menurut Turban et al (2005) memandang monitoring sebagai fase inteligensi yang diterapkan pada fase implementasi. Model Simon merupakan karakteristik yang paling kuat dan lengkap mengenai pengambilan keputusan rasional. Gambaran konseptual mengenai proses pengambilan keputusan ditunjukkan pada Gambar 2.2 berikut :



Gambar 2.2 Fase-fase Proses Pengambilan Keputusan

Sumber : Efraim, Aronson & Liang (2005)

Pada Gambar 2.2 merupakan fase-fase proses pengambilan keputusan yang terdiri dari beberapa fase sebagai berikut :

1. Fase Kecerdasan

Fase ini dimulai dengan identifikasi tujuan dan maksud dari organisasi. Selanjutnya permasalahan akan muncul yang kemudian akan diklasifikasikan. Permasalahan yang kompleks bisa didetailkan (*decompose*) menjadi sub *problem*. Akhir dari fase ini akan menghasilkan pernyataan "*problem*".

2. Fase Desain

Fase ini meliputi analisis, perancangan serta pengujian terhadap solusi-solusi yang layak atau memungkinkan. Pada fase ini pemodelan kondisi *problem* meliputi konstruksi, pengujian dan validasi. Jenis pemodelan meliputi pemodelan kualitatif dan kuantitatif (pemodelan matematis).

3. Fase Pemilihan

Fase ini meliputi pencarian, evaluasi dan rekomendasi sebuah solusi untuk model. Sebuah solusi untuk model merupakan sekumpulan spesifik nilai-nilai untuk variabel keputusan (ada alternatif-alternatif yang akan terseleksi). Solusi model disini tidak sama dengan solusi permasalahan. Solusi model merupakan solusi yang direkomendasikan untuk solusi permasalahan.

4. Fase implementasi



Jika seluruh tahapan telah dilewati maka selanjutnya pada tahap ini dilakukan penerapan terhadap rancangan sistem yang telah dibuat serta pelaksanaan alternatif dan tindakan yang dipilih pada fase pemilihan.

Dari keempat fase di atas, masing-masing dibagi atas dua tahap yakni tahap satu sampai dengan tiga adalah pengambilan keputusan, sedangkan tahap empat adalah untuk penyelesaian masalah.

2.5 Analytical Hierarchy Process (AHP)

Pada dasarnya tahap atau proses dari pengambilan keputusan merupakan memilih suatu alternatif. AHP umumnya digunakan dengan tujuan untuk menyusun prioritas dari berbagai alternatif pilihan yang ada dan pilihan-pilihan tersebut bersifat kompleks atau multi kriteria (Burgeois, 2005). Metode AHP dikembangkan pada tahun 1970 oleh Thomas L. Saaty. Pada awalnya metode AHP digunakan untuk memecakan masalah pada suatu intuisi yang kompleks dan yang tidak memiliki struktur kedalam bagian-bagiannya.

Beberapa kelebihan penggunaan metode AHP adalah sebagai berikut (Suryadi & Ramdhani, 1988) :

1. Struktur yang berbentuk hirarki sebagai konsekuensi dari kriteria yang dipilih sampai pada sub kriteria yang paling dalam.
2. Memperhatikan validitas sampai dengan batas toleransi inkonsistensi berbagai kriteria dan alternatif yang dipilih oleh para pengambil keputusan.
3. Memperhitungkan daya tahan atau ketahanan keluaran analisis sensitifitas pembuat keputusan.

Selain itu metode AHP mempunyai kemampuan untuk memecahkan masalah yang multi objektif dan multi kriteria berdasarkan pada perbandingan preferensi dari pembuatan keputusan yang sangat komperhensif.

2.5.1 Langkah-langkah Penggunaan Metode AHP

Dalam penyelesaian dengan metode AHP langkah-langkahnya sebagai berikut (Saaty, T. L. 1991) :

1. Mendefnisikan permasalahan dan menentukan tujuan.
2. Menyusun masalah ke dalam suatu struktur hierarki sehingga permasalahan yang kompleks dapat ditinjau dari sisi yang detail dan terukur.
3. Menghitung nilai prioritas untuk tiap elemen masalah pada setiap hierarki. Prioritas ini dihasilkan dari suatu matriks perbandingan berpasangan antara seluruh elemen pada tingkat hierarki yang sama.

Tabel 2.3 Susunan Matriks Perbandingan Berpasangan

	A1	A2	A3
A1	1	A_{12}	A_{13}
A2	A_{ij}	1	A_{23}
A3	A_{ij}	A_{ij}	1

Sumber : Saaty, T. L (1991)

- Melakukan pengujian konsistensi terhadap perbandingan antar elemen yang ditempatkan pada tingkat hierarki untuk digunakan dalam pertimbangan perhitungan perangkingan akhir.

Tabel 2.4 Intensitas Kepentingan Perangkingan Akhir

Intensitas Kepentingan	Keterangan
1	Kedua elemen sama pentingnya
3	Elemen yang satu sedikit lebih penting daripada elemen lainnya
5	Elemen yang satu lebih penting daripada elemen lainnya
7	Satu elemen jelas mutlak penting daripada elemen lainnya
9	Satu elemen mutlak penting daripada elemen lainnya
2,4,6,8	Nilai rata-rata tengah dua nilai pertimbangan yang berdekatan
Kebalikan	Nilai kebalikan $A_{(i,j)} = \frac{1}{A_{(j,i)}}$ dimana A adalah matrik perbandingan berpasangan antar elemen. Baik kriteria, sub kriteria maupun alternatif tujuan.

Sumber : (Saaty, T. L. 1991. Pengambilan Keputusan Bagi Para Pemimpin)

Jika A_{ij} adalah elemen dari baris ke-i dan kolom ke-j pada matrik berpasangan, maka rumus perhitungan untuk mengisi kolom A_{ij} adalah dengan Persamaan 2.1 berikut :

$$A_{ij} = \frac{1}{A_{ji}} \tag{2.1}$$



5. Berdasarkan pertimbangan terhadap perbandingan berpasangan disintetis untuk memperoleh keseluruhan prioritas. Proses sintetis dengan menjumlahkan nilai-nilai pada setiap kolom matriks, membagi setiap nilai dari kolom dengan total kolom yang bersangkutan untuk memperoleh normalisasi matriks dengan Persamaan 2.2 berikut :

$$\text{Nilai elemen baru} = \frac{A_{ij}}{\text{Jumlah perkolom tiap kriteria}} \quad (2.2)$$

6. Pembobotan dengan menjumlahkan nilai-nilai dari setiap baris dan membaginya dengan jumlah kriteria dengan Persamaan 2.3 berikut :

$$\text{Bobot Prioritas} = \frac{\text{jumlah baris nilai elemen baru}}{\text{banyak kriteria}} \quad (2.3)$$

7. Menghitung konsistensi untuk mengetahui seberapa baik konsistensi yang ada. Untuk menghitung konsistensi maka dilakukan dengan mengalikan nilai pada kolom pertama dengan prioritas relatif elemen pertama, nilai kolom kedua dengan prioritas relatif elemen kedua dan seterusnya, kemudian tiap baris dijumlahkan untuk mendapatkan nilai λ_{\max} seperti pada Persamaan 2.4.

$$\lambda_{\max} = \frac{\sum \text{Bobot Prioritas}}{n} \quad (2.4)$$

Keterangan :

Ax merupakan perkalian antara masing-masing kriteria pada matrik perbandingan dengan bobot prioritas.

n merupakan banyaknya kriteria yang digunakan.

8. Menghitung nilai *Consistency Index* (CI) menggunakan Persamaan 2.5 berikut :

$$CI = \frac{(\lambda_{\max} - n)}{(n-1)} \quad (2.5)$$

Keterangan :

CI merupakan *Consistency Index* yang akan dihitung.

λ_{\max} merupakan nilai yang didapat dari penghitungan sebelumnya.

n merupakan banyaknya kriteria yang kita gunakan.

9. Menghitung *Consistency Ratio* (CR) menggunakan Persamaan 2.6 berikut :

$$CR = \frac{CI}{IR} \quad (2.6)$$

Keterangan :

CI merupakan *Consistency Index*

IR merupakan *Index Random Consistency* yang didapat dari tabel Ratio Index.

- Memeriksa konsistensi hierarki berdasarkan Tabel 2.5 *Ratio Index (RI)*. Jika nilainya < 0,1 maka hasil perhitungan bisa dinyatakan benar atau konsisten.

Tabel 2.5 Ratio Index

<i>N</i>	1	2	3	4	5	6	7
<i>RI</i>	0.00	0.00	0.58	0.90	1.12	1.24	1.32
<i>N</i>	8	9	10	11	12	13	14
<i>RI</i>	1.41	1.45	1.49	1.51	1.48	1.56	1.57
<i>N</i>	15	16	17	18	19	20	21
<i>RI</i>	1.59	1.5978	1.608	1.618	1.665	1.634	1.6409
<i>N</i>	22	23	24	25	26	27	28
<i>RI</i>	1.6470	1.6526	1.6577	1.6624	1.6667	1.6706	1.6743
<i>N</i>	29	30	31	32	33	34	35
<i>RI</i>	1.6777	1.6809	1.6838	1.6867	1.6893	1.6917	1.6940
<i>N</i>	36	37	38	39			
<i>RI</i>	1.6962	1.6982	1.7002	1.7020			

Sumber : (Alonso,Jose Antonio & Lamata, M. Teresa 2006)

2.6 Normalisasi

Normalisasi dilakukan untuk mendapatkan nilai data dengan ukuran yang lebih kecil sehingga dapat mewakili data asli (Jain, Yogendra Kumar & Bhandare, Santosh Kumar 2011). Tujuan normalisasi adalah untuk menghindari beberapa atribut yang memiliki nilai besar mempengaruhi atribut lain yang diukur dengan skala yang lebih kecil. Terdapat beberapa metode untuk normalisasi data, termasuk *min-max normalization*, *z-score normalization* dan *normalization by decimal scaling*.

Min Max Normalization

Min-max normalization menggunakan transformasi linier pada data asli. Persamaan 2.7 merupakan metode *min-max normalization*.

$$d' = \frac{[d - \min(p)]}{[\max(p) - \min(p)]} \times (BA - BB) + BB \quad (2.7)$$

Dimana :

$\min(p)$ = nilai minimum dari atribut yang digunakan

$\max(p)$ = nilai maksimum dari atribut yang digunakan.

d = merupakan data yang akan dinormalisasi.

BA = merupakan batas atas *interval*.

BB = merupakan batas bawah *interval*.

Jika menggunakan *interval* [0,1] maka Persamaan 2.7 dapat disederhanakan seperti Persamaan 2.8.

$$d' = \frac{d - \min(p)}{\max(p) - \min(p)} \quad (2.8)$$

2.7 Klasifikasi

Klasifikasi adalah bentuk dari analisis data dengan membuat suatu model atau fungsi untuk menggambarkan kelas data penting (Han, Kamber, & Pei, 2012). Model atau *classifier* dibangun untuk memprediksi suatu kelas. Dalam proses klasifikasi terdapat dua data yang berbeda yakni data *training* dan data *testing*. Data *training* menjadi acuan untuk membuat aturan pada klasifikasi data. Data *testing* digunakan untuk uji coba pada data yang belum mempunyai kelas berdasarkan aturan pada data *training*.

Berikut merupakan tahapan klasifikasi :

1. *Learning Step*

Tahap *learning* merupakan bentuk analisis data dengan membangun model atau *classifier*. Pada tahap ini suatu model klasifikasi dibangun dengan menggunakan data yang telah diketahui kelasnya untuk perkiraan.

2. *Classification Step*

Pada tahap ini suatu model digunakan untuk memprediksi kelas dengan memberikan label pada data yang akan digunakan.

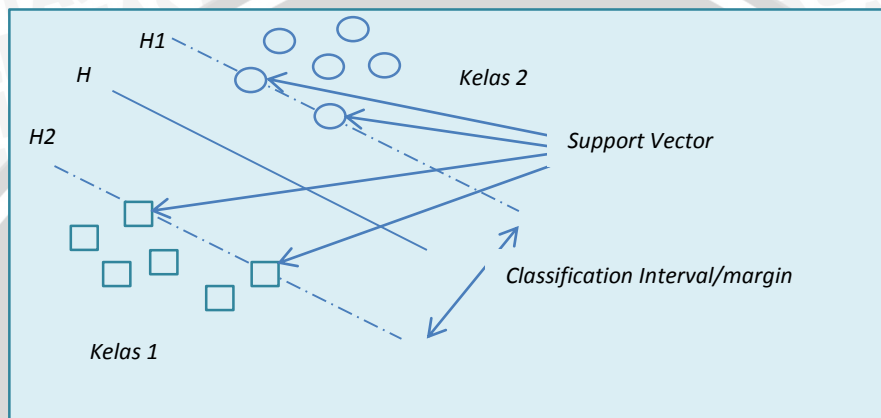
2.8 Support Vector Machine (SVM)

Support Vector Machine (SVM) merupakan seperangkat metode pembelajaran mesin pada teori statistik Vladimir Vapnik (Kartal Hasan Basri, Cebi Ferhan. 2013). Metode SVM merupakan salah satu metode yang digunakan untuk pemilihan variabel dan proses klasifikasi data (Novianti Fourina Ayu, Purnami Santi Wulan. 2012). Dalam teknik pemecahan masalah menggunakan metode SVM dilakukan dengan menentukan suatu *hyperplane* terbaik. *Hyperplane* merupakan suatu bidang pemisah yang terletak di tengah antara dua set objek dari dua kelas.

SVM dapat melakukan proses generalisasi untuk menentukan pola tertentu berdasarkan data *testing* dan data *training*. SVM memiliki struktur yang sederhana sehingga cepat dalam menyelesaikan permasalahan. Metode SVM memiliki kemampuan menemukan fungsi pemisah (*classifier*) yang optimal (Damayanti Fitri, Arifin Agus Zainal. 2010). Metode SVM dalam penerapannya

dapat digunakan hanya untuk klasifikasi dua kelas dan selalu memberikan hasil yang sama dalam setiap kali melakukan proses *running*.

Berdasarkan karakteristiknya, metode SVM dibagi menjadi dua yakni *linier* dan *non-linier* (Yang Yi, Du Quishi & Zhao Jinying. 2010). SVM yang dikembangkan menggunakan prinsip *linier* adalah dengan memisahkan kedua *class* secara terpisah oleh *hyperplane* yaitu dengan teknik *soft margin*. Sedangkan untuk menyelesaikan permasalahan *non-linier* yakni dengan menerapkan fungsi *kernel trick* pada ruang berdimensi tinggi (Racman Farizi, Purnami Santi Wulan. 2012).



Gambar 2.3 Konsep Dasar Penerapan Metode SVM

Sumber : Yang Yi Du, Quishi, Zhao Jinyang (2010)

Pada Gambar 2.3 dijelaskan tentang dasar penerapan SVM dengan dua dimensi, simbol bulat dan persegi mewakili dua jenis sampel yakni H_1 dan H_2 yang dipisahkan oleh *hyperplane* (H) disebut dengan *classification interval* atau margin vektor terdekat yang memisahkan *hyperplane* disebut dengan *support vector*.

2.8.1 Support Vector Machine Linier

Pada klasifikasi linier metode SVM dibagi menjadi dua jenis yakni *separable* dan *non separable* (Rachman Fairizi, Purnami Santi Wulan. 2012). SVM melakukan klasifikasi himpunan data *training* berpasangan dari dua kelas (Novianti Fourina Ayu, Purnami Santi Wulan. 2012). Klasifikasi *linier* data merupakan data yang dapat dipisahkan secara *linier*. Misalnya diketahui $x_i = \{x_1, \dots, x_n\}$ adalah titik pada dataset dan $y_i = y \in \{+1, -1\}$ adalah kelas pada x_i . Bidang pemisah yang terbaik adalah yang dapat memisahkan semua set data x_i . Bidang pemisah pertama membatasi kelas pertama sedangkan bidang pemisah kedua membatasi kelas kedua sehingga diperoleh Persamaan 2.9 dan Persamaan 2.10.

$$(w \cdot x_i + b) \geq +1 \text{ untuk } y_i = +1 \tag{2.9}$$

$$(w \cdot x_i + b) \leq -1 \text{ untuk } y_i = -1 \tag{2.10}$$

Dimana :

w = vektor bobot

x_i = data ke- i , untuk $i = 1, 2, \dots, n$

b = bilangan skalar yang menyatakan nilai bias

y_i = target atau kelas (*sample* positif (+1) dan *sample* negatif (-1))

Dimana variabel w merupakan bidang normal antara bidang pemisah terhadap pusat koordinat, dan variabel b adalah posisi bidang relatif terhadap pusat koordinat. Jarak antara *hyperplane* dan titik terdekat untuk menemukan *margin* terbesar adalah dengan memaksimalkan fungsi $\frac{1}{\|w\|}$ atau dengan meminimalkan $\|w\|^2$. Bidang pemisah atau *hyperplane* kedua kelas dijelaskan dengan Persamaan 2.11.

$$y_i(w \cdot x_i + b) \geq 1, i = 1, 2, \dots, n \quad (2.11)$$

Hal ini dapat dirumuskan sebagai *Quadratic Programming* dengan mencari titik minimal. *Hyperplane* terbaik dengan *margin* terbesar dapat dirumuskan menjadi masalah optimasi constrain pada Persamaan 2.12.

$$\text{Min} = \frac{1}{2} \|w\|^2 \quad (2.12)$$

Dengan syarat $y_i\{w \cdot x_i + b\} \geq 1$, persamaan 2.12 untuk menyelesaikan permasalahan optimasi yang dapat diselesaikan dengan menggunakan fungsi *Lagrange*. Fungsi *Lagrange* yang digunakan adalah *Lagrange Multiplier* dapat dilihat pada Persamaan 2.13.

$$\text{Min}L(w, b, \alpha) = \frac{1}{2} \|w\|^2 + \sum_{i=1}^n \alpha_i [y_i(w \cdot x_i + b) - 1] \quad (2.13)$$

Dimana α_i merupakan nilai bobot dari setiap titik data. Notasi ini bernilai nol atau positif ($\alpha_i \geq 0$), dengan meminimalkan L terhadap w dan b , maka untuk menghitung w dan b berdasarkan Persamaan 2.14.

$$w = \sum_{i=1}^n \alpha_i y_i x_i \quad b = -\frac{1}{2} (w \cdot x^+ + w \cdot x^-) \quad (2.14)$$

Dimana x^+ adalah nilai dari salah satu *support vector* dari *class positive*. Sedangkan x^- adalah nilai dari salah satu *support vector* dari *class negative*. Persamaan 2.12 dapat dimodifikasi menjadi fungsi maksimal untuk mencari *hyperplane* terbaik dapat dirumuskan dalam persamaan 2.15.

$$\text{Max} \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j x_i \cdot x_j \quad (2.15)$$

Dengan batasan

$$\alpha_i \geq 0 (i=1,2,\dots,n) \text{ dan } \sum_{i,j=1}^n \alpha_i y_i = 0$$

Hasil dari perhitungan menggunakan persamaan tersebut diperoleh nilai α_i bernilai positif yang nanti akan digunakan untuk menemukan w . Terdapat α_i untuk setiap data *training* yang memiliki nilai $\alpha_i > 0$ adalah *support vector*, sedangkan sisanya memiliki nilai $\alpha_i = 0$. Dengan demikian, fungsi keputusan yang dihasilkan hanya dipengaruhi oleh *support vector*. Untuk mendapatkan klasifikasi yang optimal data *testing* x dapat digunakan persamaan 2.16. hasil klasifikasi dapat dirumuskan dengan Persamaan 2.17.

$$f(x) = (w \cdot x + b) \text{ atau } \sum_{i=1}^m \alpha_i y_i x_i x + b \tag{2.16}$$

$$\text{Fungsi Klasifikasi} = \text{sign}(f(x)) \tag{2.17}$$

Keterangan variabel m adalah jumlah *support vector* atau titik data yang memiliki $\alpha_i > 0$, dan notasi x merupakan data yang akan diklasifikasikan.

2.8.2 Support Vector Machine Non-Linier

Persamaan *non-linier* dapat ditransformasikan dalam persamaan *linier* dengan menggunakan dimensi yang tinggi (Kartal Hasan Basri, Cebi Ferhan. 2013). Sebuah fungsi *kernel* dapat didefinisikan menjadi *input kernel trick*. *Kernel trick* merupakan bagian dari pembelajaran dalam metode SVM, dimana untuk mengetahui fungsi *kernel* tanpa harus mengetahui wujud dari fungsi *non-linier*. *Kernel trick* dapat dirumuskan pada Persamaan 2.18.

$$K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j) \tag{2.18}$$

Dimana :

$$K(x_i, x_j) = \text{fungsi kernel}$$

$$\phi(x_i) = \text{transformasi data ke-}i$$

Notasi x_i, x_j dengan *dot product* dapat diganti dengan simbol K . Pada dasarnya untuk mendapatkan *hyperplane* yang optimal dapat menggunakan model *quadratic problem* yang dijelaskan sebelumnya, seperti pada Persamaan 2.19 berikut :

$$\min \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n \alpha_i y_i K(x_i, x_j) \tag{2.19}$$

Dengan batasan sebagai berikut :

$$\sum_{i=1}^n \alpha_i y_i = 0, 0 \leq \alpha_i \leq C \text{ dan } i = 1, 2, \dots, n$$



Nilai $K(x, y)$ merupakan fungsi *kernel* yang menunjukkan pemetaan *non linier* pada *feature space*. Solusi yang diperoleh untuk fungsi keputusan klasifikasi optimal dapat dilihat pada Persamaan 2.20.

$$f(x) = \text{sign}\left(\sum_{i=1}^m \alpha_i y_i K(x, x_i) + b\right) \quad (2.20)$$

Dimana :

$$K(x_i, x_j) = \text{fungsi kernel } K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j)$$

$$y_i = \text{target/kelas (sample positif (+1) dan sample negatif (-1))}$$

$$m = \text{jumlah support vector}$$

$$b = \text{bilangan skalar yang menyatakan bias}$$

Masing-masing *kernel* dapat digunakan dalam percobaan untuk menentukan parameter kernel dan memberikan keakuratan yang terbaik dalam proses klasifikasi. Proses SVM dimulai dengan perhitungan fungsi *kernel*. Berikut adalah beberapa fungsi *kernel* yang dapat digunakan.

- a. Fungsi *linier* didefinisikan pada Persamaan 2.21.

$$K(x, x_i) = x \cdot x_i \quad (2.21)$$

- b. Fungsi *polynomial* didefinisikan pada Persamaan 2.22.

$$K(x, x_i) = (x \cdot x_i)^d \quad (2.22)$$

- c. Fungsi *gaussian RBF* didefinisikan pada Persamaan 2.23

$$K(x, x_i) = \exp\left(-\frac{\|x - x_i\|^2}{2\sigma^2}\right) \quad (2.23)$$

- d. Fungsi *sigmoid* didefinisikan pada Persamaan 2.24.

$$K(x, x_i) = \tanh(\sigma(x, x_i) + C) \quad (2.24)$$

Kernel linier digunakan ketika data yang akan diklasifikasikan dapat terpisahkan dengan sebuah garis atau *hyperplane*. Sedangkan kernel *non-linier* digunakan ketika data hanya dapat dipisahkan dengan garis lengkung atau sebuah bidang pada ruang yang berdimensi tinggi.

2.9 Multi-Class SVM

Vapnik yang pertama kali memperkenalkan SVM, dimana pada saat itu SVM hanya mampu mengklasifikasikan data ke dalam dua kelas (*binary classifier*). SVM dapat diterapkan pada pengklasifikasian data yang sifatnya *multi-class* dengan menggunakan dua pendekatan yang umum dilakukan. Pendekatan pertama yakni dengan cara penggabungan beberapa SVM biner, sedangkan pendekatan kedua dengan cara menggabungkan semua data yang terdiri dari beberapa kelas ke dalam permasalahan optimasi.

Dalam permasalahan optimasi yang harus diselesaikan, pendekatan yang kedua jauh lebih rumit dalam penggunaannya. Beberapa metode yang digunakan pada pendekatan yang pertama yakni metode *one-against-all*, *one-against-one* dan *Directed Acyclic Graph Support Vector Machine* (DAGSVM). Pada penelitian ini akan menggunakan metode *one-against-all* untuk memecahkan permasalahan pengklasifikasian dengan *multi-class SVM*.

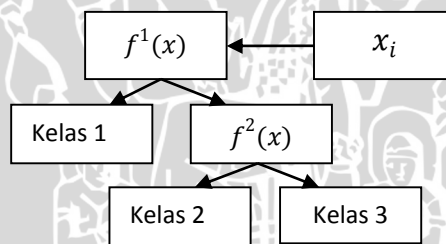
2.9.1 Metode One-Against-All

Permasalahan *multi class* dalam penelitian ini akan dipecahkan dengan metode *one-against-all*. Metode ini dibangun berdasarkan sejumlah k SVM biner (k adalah jumlah kelas). SVM ke- i dilatih dengan seluruh data *sample* yang terdapat pada ke- i dengan memiliki label kelas positif (+1) dan label kelas negatif (-1). Apabila terdapat permasalahan klasifikasi dengan tiga buah kelas, untuk proses *training* pada kelas tersebut akan menggunakan tiga buah SVM biner seperti pada Tabel 2.6.

Tabel 2.6 Metode One-against-all dengan 3 Kelas

$y_i = 1$	$y_i = -1$	Hipotesis
Kelas 1	Bukan kelas 1	$f^1(x) = (w^1)x + b^1$
Kelas 2	Kelas 3	$f^2(x) = (w^2)x + b^2$

Sumber : Hsu and Lin (2002) dan Sembiring (2007)



Gambar 2.4 Contoh Klasifikasi One-against-all untuk 3 kelas

Sumber : Hsu and Lin (2002) dan Sembiring (2007)

2.9.2 Metode One-Against-One

Metode *one-against-one* ini dibangun berdasarkan $\frac{k(k-1)}{2}$ buah model klasifikasi SVM biner. Metode ini dilatih dari dua kelas yakni kelas ke- i dan kelas ke- j , dimana pencarian solusi dilakukan untuk persoalan optimasi konstrain. Dalam melakukan pengujian terdapat beberapa metode, dimana pengujian dilaksanakan setelah keseluruhan $\frac{k(k-1)}{2}$ selesai dibangun. Salah satu dari metode tersebut adalah metode *voting* seperti yang ditunjukkan pada Tabel 2.7 dan Gambar 2.5.

Tabel 2.7 Metode One-against-one dengan 6 SVM Biner

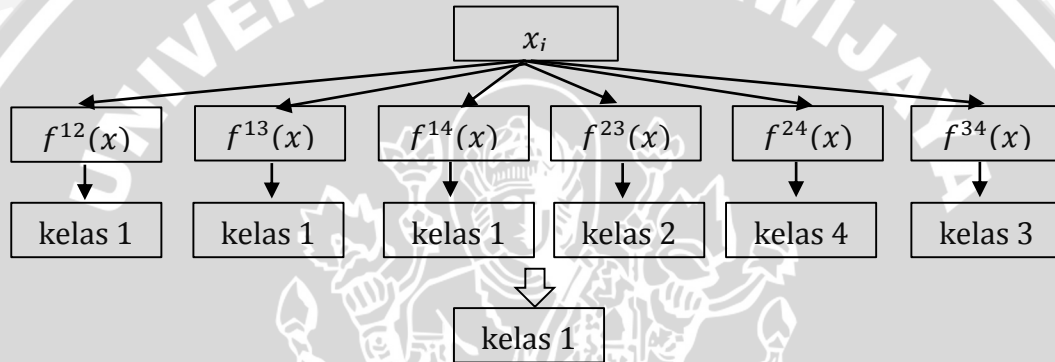
$y_i = 1$	$y_i = -1$	Hipotesis
-----------	------------	-----------



Kelas 1	Bukan kelas 2	$f^{12}(x) = (w^{12})x + b^{12}$
Kelas 1	Bukan kelas 3	$f^{13}(x) = (w^{13})x + b^{13}$
Kelas 1	Bukan kelas 4	$f^{14}(x) = (w^{14})x + b^{14}$
Kelas 2	Bukan kelas 3	$f^{23}(x) = (w^{23})x + b^{23}$
Kelas 2	Bukan kelas 4	$f^{24}(x) = (w^{24})x + b^{24}$
Kelas 3	Bukan kelas 4	$f^{34}(x) = (w^{34})x + b^{34}$

Sumber : Hsu and Lin (2002) dan Sembiring (2007)

Pada Tabel 2.7 merupakan metode *One-against-one* dengan 6 SVM biner dimana pada fungsi f^{12} sampai f^{14} digunakan untuk memisahkan kelas 1 ketika nilai dari $y_i = 1$ adalah kelas 1 dan ketika nilai dari $y_i = -1$ bukan kelas dari fungsi pada digit terakhir.



Gambar 2.5 Klasifikasi Menggunakan Metode *One-Against-One*

Sumber : Hsu and Lin (2002) dan Sembiring (2007)

2.10 Metode Sequential Training pada SVM

Hyperplane yang optimal pada SVM dapat ditemukan dengan merumuskannya ke dalam *quadratic problem* dan diselesaikan dengan analisa numerik. Metode ini dikembangkan oleh Vijayakumar untuk mencari nilai α , dapat dijelaskan dalam tahapan berikut (Vijayakumar, 1999) :

1. Setelah perhitungan *kernel* selesai dilakukan, selanjutnya dilakukan inialisasi untuk parameter SVM misalnya nilai $\alpha_i = 0$.
2. Lakukan penghitungan komponen matriks pada Persamaan 2.25.

$$D_{ij} = y_i y_j (K(x_i, x_j) + \lambda^2) \tag{2.25}$$

Notasi x merupakan data ke- i dan data ke- j . Notasi y merupakan kelas dari data ke- i dan data ke- j . $K(x_i, x_j)$ merupakan fungsi kernel yang digunakan. Lakukan 1 dan langkah 2 untuk $i, j = 1, 2, \dots, n$.

3. Untuk setiap $i = 1, 2, \dots, n$ dihitung menggunakan Persamaan 2.26 sampai dengan 2.28.



$$a) \quad E_i = \sum_{j=1}^n \alpha_i D_{ij} \quad (2.26)$$

$$b) \quad \delta\alpha_i = \min\{\max[\gamma(1-E_i), -\alpha_i], C - \alpha_i\} \quad (2.27)$$

$\delta\alpha_i$ merupakan variabel tunggal, bukan bentuk perlakuan dari δ dan α_i yang ditunjukkan pada Persamaan 2.28.

$$c) \quad \alpha_i = \alpha_i + \delta\alpha_i \text{ (memperbaharuni nilai } \alpha_i \text{)} \quad (2.28)$$

Notasi γ adalah parameter untuk mengontrol kecepatan proses *learning* (*learning rate*). Nilai konstanta untuk parameter γ , dapat dilihat pada Persamaan 2.29.

$$\gamma = \frac{CLR}{\max_{\{i\}} D_{ii}} \quad (2.29)$$

$\max_{\{i\}} D_{ii}$ merupakan nilai maksimum yang diperoleh dari nilai diagonal pada matriks D_{ii} . *CLR* merupakan *constant learning rate*. Dengan menggunakan nilai konstanta untuk nilai yang diinputkan oleh pengguna dalam pengujian 0.01. Fungsi dari $\delta\alpha_i$ merupakan fungsi konvergensi untuk memantau perubahan suatu fungsi dari *Lagrange Multiplier*. Jika data latih telah mencapai nilai konvergen maksimum ($|\delta\alpha_i| < \varepsilon$), dan ketika nilai maksimum dari iterasi mencapai nilai yang ditentukan maka iterasi akan dihentikan.

4. Proses tersebut diulangi sampai α mencapai nilai konvergen. Konvergen dapat didefinisikan dari tingkat perubahan pada nilai α .
5. Mendapatkan nilai *Support Vector* dapat diketahui dengan $SV = (\alpha_i > ThresholdSV)$. Nilai *Threshold SV* tersebut adalah $ThresholdSV \geq 0$.

2.11 Evaluasi

Evaluasi dilakukan untuk mengetahui tingkat akurasi dari hasil klasifikasi dengan cara menghitung jumlah *record* uji yang kelasnya diprediksi secara tepat (akurat). Penelitian ini menggunakan *classification accuracy*, dimana *classification accuracy* adalah ukuran suatu ketepatan dari kejadian-kejadian yang diinginkan (Purnami dan Embong, 2009). Berikut persamaan dari *classification accuracy* seperti pada Persamaan 2.30 :

$$Classification Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \times 100\% \quad (2.30)$$

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). Persamaan 2.30 dapat ditentukan

dengan menggunakan nilai yang sudah ditentukan pada *confusion matrix*. *Confusion matrix* merupakan klasifikasi aktual dan prediksi pada sistem klasifikasi. Adapapun *confusion matrix* 2x2 ditunjukkan pada Tabel 2.8 berikut :

Tabel 2.8 Confusion Matrix 2x2

	Predicted Class	
	Positive	Negative
Actual Class Positive	True Positive (TP)	False Negative (FN)
Actual Class Negative	False Positive (FP)	True Negative (TN)

Sumber : Hermaduanti dan Kusumadewi (2008)

Adapun istilah pada Persamaan 2.30 untuk proses klasifikasi adalah sebagai berikut :

- TP merupakan jumlah nilai positif yang diklasifikasikan positif.
- TN merupakan jumlah nilai negatif yang diklasifikasikan sebagai negatif.
- FP merupakan nilai negatif yang diklasifikasikan sebagai positif.
- FN merupakan nilai positif yang diklasifikasikan sebagai negatif.

Sedangkan Persamaan 2.30 juga dapat digunakan untuk *confusion matrix* 3x3 yang ditunjukkan pada Tabel 2.9 berikut :

Tabel 2.9 Confusion Matrix 3x3

		Predicted Class		
		Class 1	Class 2	Class 3
Actual Class	Class 1	X_{11}	X_{12}	X_{13}
	Class 2	X_{21}	X_{22}	X_{23}
	Class 3	X_{31}	X_{32}	X_{33}

Sumber : Hermaduanti dan Kusumadewi (2008)

Keterangan :

- TP (*True Positive*)

Apabila *actual class* dan sistem sama, maka akan menghasilkan *positive class* jika :

- *Actual class "A"* sistem juga memutuskan kelas "A"
- *Actual class "B"* sistem juga memutuskan kelas "B"
- *Actual class "C"* sistem juga memutuskan kelas "C"

Untuk rumus TP dapat dilihat pada Persamaan 2.31.

$$TP = X_{11} + X_{12} + X_{13} \tag{2.31}$$



b. TN (*True Negative*)

Apabila *actual class* dan sistem sama menghasilkan *negative class* jika :

- *Actual class* bukan *class "A"* sistem juga memutuskan bukan *class "A"*
- *Actual class* bukan *class "B"* sistem juga memutuskan bukan *class "B"*
- *Actual class* bukan *class "C"* sistem juga memutuskan bukan *class "C"*

Untuk rumus TN dapat dilihat pada Persamaan 2.32.

$$TN = (X_{22} + X_{33}) + (X_{11} + X_{33}) + (X_{11} + X_{22}) \quad (2.32)$$

c. FP (*False Positive*)

Apabila *actual class* negatif tetapi sistem memutuskan positif *class* jika :

- *Actual class* bukan *class "A"* tetapi sistem memutuskan *class "A"*
- *Actual class* bukan *class "B"* tetapi sistem memutuskan *class "B"*
- *Actual class* bukan *class "C"* tetapi sistem memutuskan *class "C"*

Untuk rumus FP dapat dilihat pada Persamaan 2.33.

$$FP = (X_{21} + X_{31}) + (X_{12} + X_{32}) + (X_{13} + X_{23}) \quad (2.33)$$

d. FN (*False Negative*)

Apabila *actual class* positif tetapi sistem memutuskan negatif *class* jika :

- *Actual class "A"* tetapi sistem memutuskan bukan *class "A"*
- *Actual class "B"* tetapi sistem memutuskan bukan *class "B"*
- *Actual class "C"* tetapi sistem memutuskan bukan *class "C"*

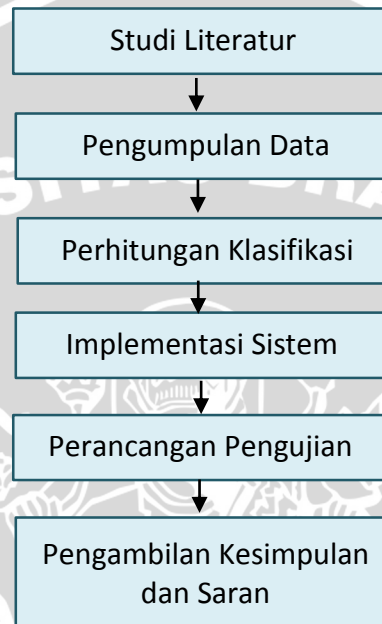
Untuk rumus FN dapat dilihat pada Persamaan 2.34.

$$FN = (X_{12} + X_{13}) + (X_{21} + X_{23}) + (X_{31} + X_{32}) \quad (2.34)$$

Dimana kelas A merupakan kelas "Sangat Layak", kelas B merupakan kelas "Layak" dan kelas C merupakan kelas "Tidak Layak".

BAB 3 METODOLOGI DAN PERANCANGAN

Pada bab ini dijelaskan mengenai langkah-langkah dan algoritma yang digunakan dalam penyusunan skripsi klasifikasi penerima beasiswa dengan metode AHP dan SVM. Langkah-langkah yang akan dilakukan untuk penelitian ini meliputi studi literatur, pengumpulan data, perhitungan klasifikasi, implementasi sistem, perancangan pengujian dan evaluasi seperti yang terlihat pada Gambar 3.1.



Gambar 3.1 Diagram Alir Pelaksanaan Penelitian

3.1 Studi Literatur

Studi literatur dilakukan dengan cara mengumpulkan dan mempelajari literatur-literatur yang berkaitan dengan penelitian. Teori-teori pendukung untuk penelitian ini diperoleh dari jurnal, buku dan penelitian sebelumnya yang memiliki topik pembahasan yang sama atau berhubungan dengan skripsi. Literatur yang digunakan dalam penelitian ini adalah yang berkaitan dengan Sistem Pendukung Keputusan, Metode SVM, Metode AHP, Pemrograman menggunakan bahasa pemrograman Java dan DBMS MySQL yang didapatkan dari jurnal, buku, *ebook*, penelitian sebelumnya dan sumber pustaka lain.

3.2 Pengumpulan Data

Pengumpulan data bertujuan untuk memperoleh data pemohon beasiswa pada Lembaga GNOTA yang digunakan dalam perancangan klasifikasi penerima beasiswa. Pengumpulan data pada penelitian ini dilakukan dengan beberapa tahap :

1. Wawancara

Wawancara dilakukan terhadap pegawai Lembaga GNOTA yang menangani alur pemberian beasiswa untuk mendapatkan data mengenai parameter yang digunakan saat melakukan seleksi beserta data pemohon beasiswa pada Lembaga GNOTA Kediri.

2. Observasi

Penelitian ini dilakukan secara langsung agar dapat memperoleh data yang dibutuhkan dalam menentukan kategori yang akan menjadi bahan pertimbangan dalam pemilihan penerima beasiswa untuk menghasilkan perhitungan dengan tingkat akurasi yang lebih baik.

3. Studi Pustaka

Metode ini dilakukan untuk menunjang dari metode sebelumnya yaitu observasi yang telah dilakukan dalam pengumpulan data dan mencari suatu referensi yang saling berhubungan dengan penelitian yang dilakukan ini.

Bentuk data yang didapat dalam penelitian ini berdasarkan metode pengumpulan data yang dilakukan, ditunjukkan pada Tabel 3.1.

Tabel 3.1 Bentuk Data Penelitian

No	Kebutuhan Data	Sumber Data	Metode	Kegunaan Data
1	Data pemohon beasiswa	Lembaga GNOTA Kediri	Permohonan data pemohon beasiswa	Sebagai data utama yang akan dilakukan seleksi
2	Data penerima beasiswa	Lembaga GNOTA Kediri	Permohonan data penerima beasiswa	Data penerima beasiswa Lembaga GNOTA yang sudah diketahui layak/ tidak menerima beasiswa sebagai data latih dalam menentukan nilai bobot kepentingan kriteria
3	Data parameter penilaian	Lembaga GNOTA Kediri	Permohonan data parameter penilaian	Sebagai acuan untuk parameter penilaian dalam proses seleksi

3.3 Perhitungan Klasifikasi

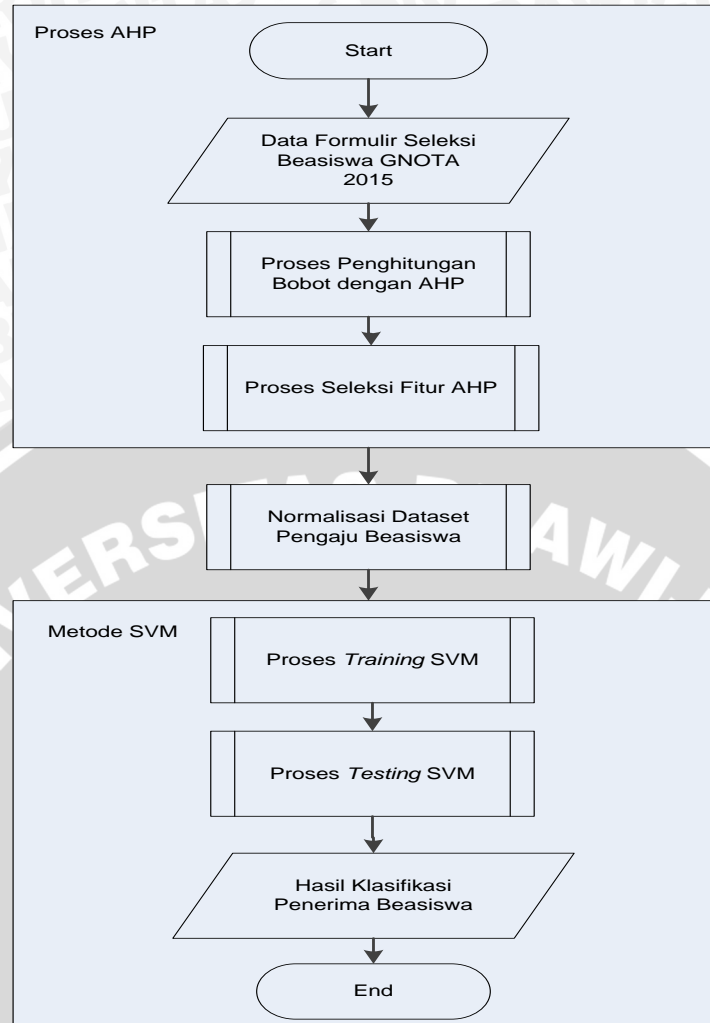
Perhitungan klasifikasi merupakan yang akan menjelaskan desain dari sistem secara keseluruhan, baik dari segi model maupun arsitektur yang digunakan beserta model perhitungan yang digunakan. Perhitungan klasifikasi dibuat berdasarkan hasil yang telah diperoleh dalam tahap pengumpulan data.

3.3.1 Deskripsi Umum

Sistem yang akan dibangun merupakan suatu sistem yang menerapkan metode AHP dan SVM dalam hal klasifikasi penerima beasiswa pada Lembaga GNOTA Kediri. Dalam penelitian ini menggunakan data yang didapatkan secara langsung dari lembaga GNOTA Kediri dengan melakukan interview dan observasi lapangan. Data penerima beasiswa akan diseleksi berdasarkan kriteria yang telah ditetapkan. Pada tahap ini metode AHP berperan dalam pembobotan untuk menentukan kriteria yang akan digunakan. Selanjutnya, kriteria yang terpilih dari hasil pembobotan dengan metode AHP akan digunakan untuk proses klasifikasi dengan metode SVM. Sekumpulan *dataset* penerima beasiswa lembaga GNOTA akan dinormalisasi sehingga menghasilkan nilai yang lebih kecil yang akan mempermudah dalam proses komputasi. Setelah dilakukan normalisasi data maka tahap selanjutnya adalah proses *training* menggunakan metode *sequential training* pada SVM untuk menemukan bidang pemisah pada *dataset* penerima beasiswa. Selanjutnya data tersebut akan diuji untuk mendapatkan hasil klasifikasi.

3.3.1 Model Perancangan Sistem

Model perancangan sistem akan digambarkan dalam bentuk diagram alir yang menggambarkan aliran proses dari komponen-komponen sistem. Diagram ini menjelaskan cara kerja sistem yang mulai dari masukan sampai keluaran yang dihasilkan. Model perancangan sistem berupa diagram blok dapat dilihat pada Gambar 3.2.



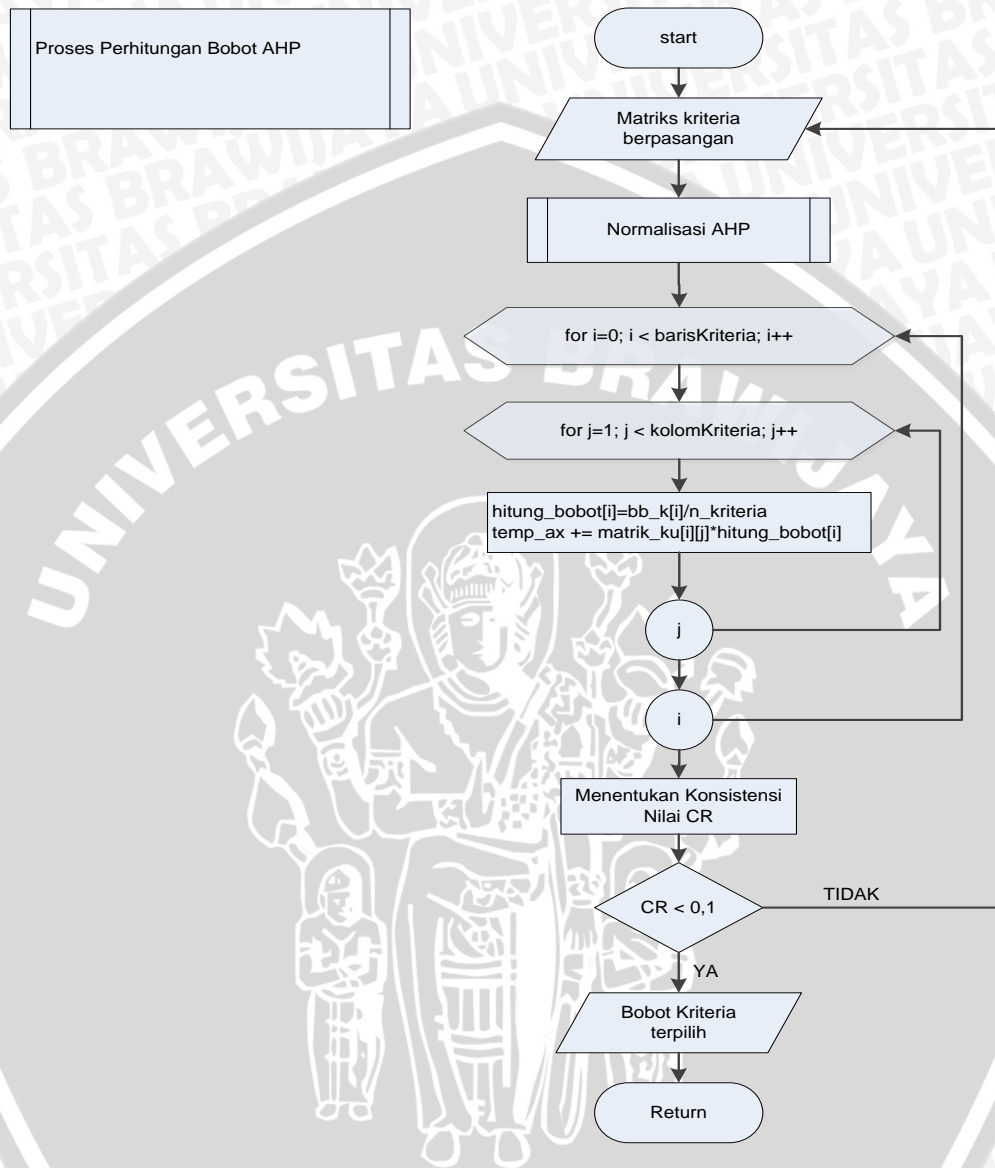
Gambar 3.2 Diagram Alir Klasifikasi Penerima Beasiswa

Pada Gambar 3.2 menggambarkan mengenai proses klasifikasi penerima beasiswa dengan metode AHP-SVM. Proses tersebut diawali dengan memberikan inputan berupa data parameter penilaian sebagai bahan acuan untuk proses seleksi penerima beasiswa. Data parameter penilaian tersebut akan dihitung menggunakan metode AHP untuk mendapatkan nilai bobot masing-masing kriteria maupun sub kriteria. Selanjutnya dilakukan seleksi fitur AHP dengan menggunakan nilai *threshold*. Kemudian proses normalisasi dataset pemohon beasiswa dan melakukan pemilihan terhadap dataset tersebut yang akan digunakan untuk data *training* dan data *testing* dalam proses klasifikasi menggunakan metode SVM. Untuk metode SVM sendiri dimulai dari proses *training* dengan metode *sequential training* untuk mendapatkan parameter SVM. Proses *testing* dilakukan setelah tahapan *training* selesai dilakukan. *Output* yang dihasilkan berupa hasil klasifikasi penerima beasiswa yang akan dievaluasi untuk mengukur nilai akurasi yang dihasilkan oleh sistem.



3.3.1.1 Poses Perhitungan Bobot Kriteria dengan AHP

Proses perhitungan bobot kriteria dengan metode AHP seperti pada Gambar 3.3 berikut :



Gambar 3.3 Diagram Alir Perhitungan Bobot AHP

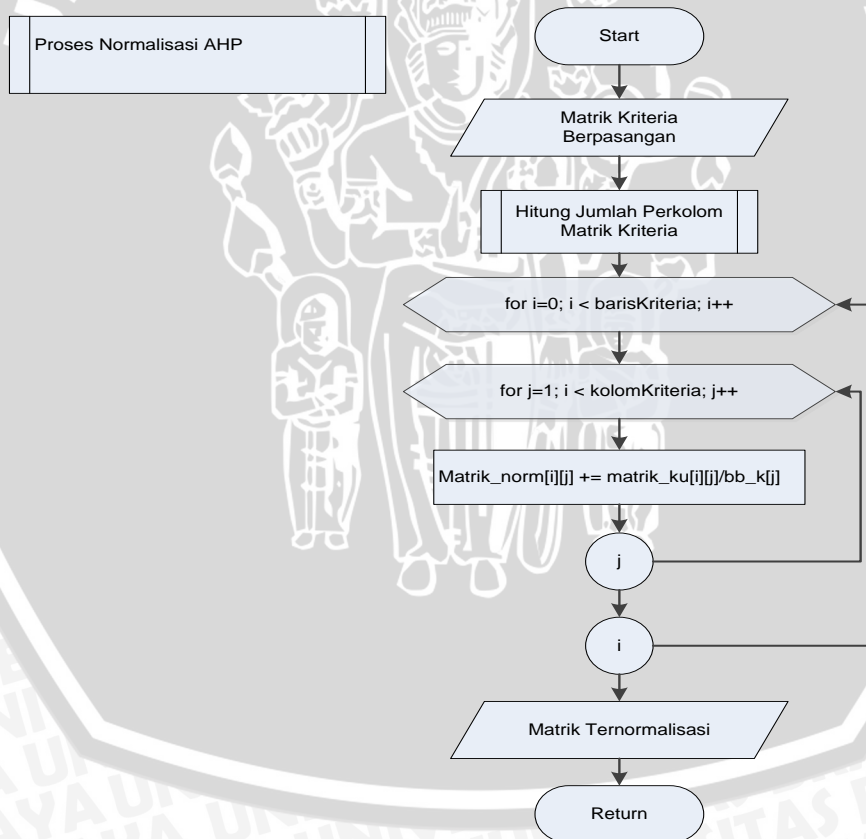
Pada Gambar 3.3 merupakan diagram alir perhitungan bobot dengan metode AHP dengan tahapan sebagai berikut :

1. Memberikan inputan berupa matrik perbandingan berpasangan pada kriteria yang digunakan.
2. Melakukan normalisasi matrik perbandingan berpasangan dan menjumlahkan setiap baris hasil normalisasi matriks perbandingan berpasangan.

3. Melakukan penghitungan bobot kriteria dengan mengambil hasil penjumlahan setiap baris matrik perbandingan berpasangan dan membaginya dengan banyak kriteria yang digunakan.
4. Menghitung nilai AX dengan menjumlahkan hasil setiap perkalian matrik perbandingan berpasangan dengan bobot kriteria.
5. Menentukan konsistensi CR (*Consistency Ratio*) seperti pada Persamaan 2.4.
6. Melakukan pengecekan hasil penghitungan nilai CR. Apabila hasil penghitungan nilai CR kurang dari 0.1 maka perhitungan tersebut dikatakan konsisten. Jika nilai CR yang dihasilkan lebih dari 0.1 maka akan dilakukan penghitungan ulang mulai dari matrik perbandingan berpasangan.
7. Selanjutnya menentukan nilai *threshold* untuk menentukan kriteria terpilih yang akan digunakan untuk proses SVM selanjutnya.
8. Hasil akhir dalam perhitungan ini berupa bobot kriteria.

3.3.1.2 Proses Normalisasi AHP

Proses normalisasi AHP seperti yang terdapat pada Gambar 3.4.



Gambar 3.4 Diagram Alir Normalisasi AHP

Berikut merupakan tahapan proses normalisasi AHP seperti pada Gambar 3.4 :

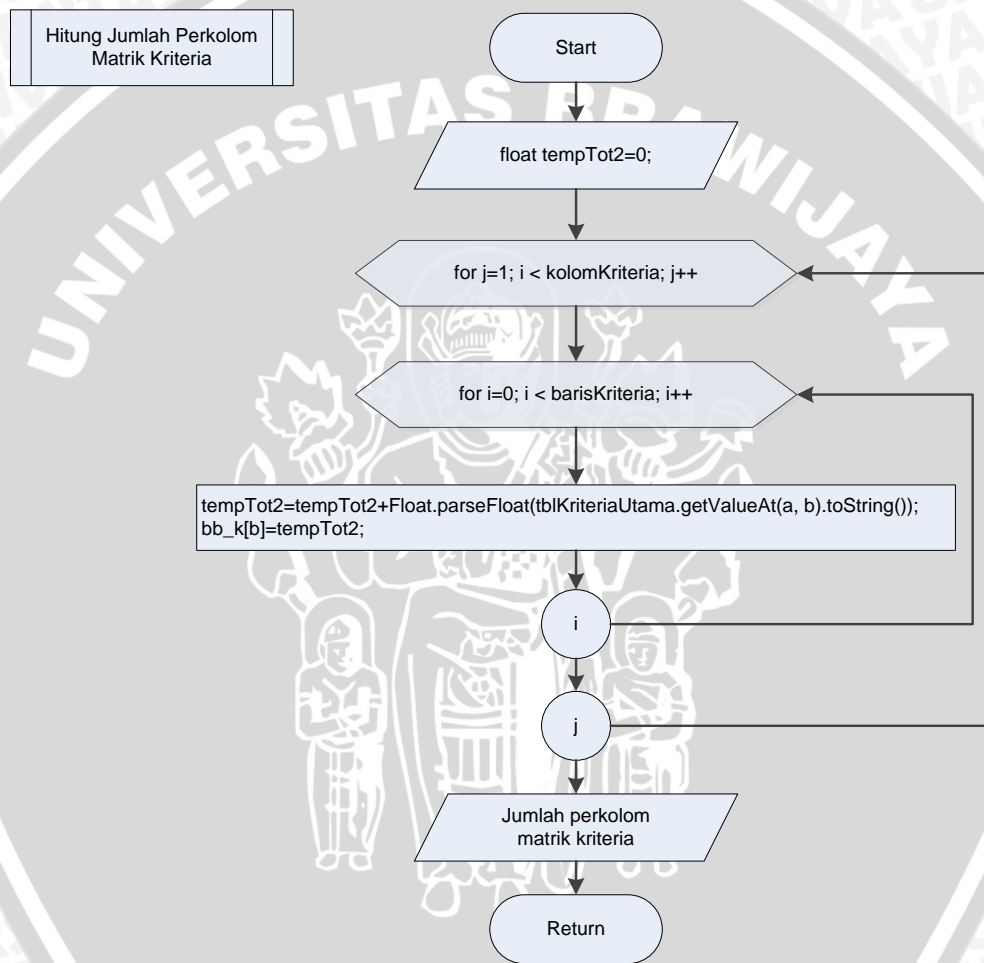
1. Memberikan masukan berupa matrik kriteria berpasangan.



2. Perhitungan normalisasi AHP dilakukan sebanyak jumlah baris dan kolom matrik kriteria berpasangan.
3. Melakukan normalisasi AHP.
4. Setelah seluruh perulangan dilakukan, selanjutnya akan diberikan *output* berupa matrik yang telah ternormalisasi.

3.3.1.3 Proses Hitung Jumlah Perkolom Matriks Perbandingan

Proses perhitungan jumlah perkolom matrik perbandingan seperti yang terdapat pada Gambar 3.5.



Gambar 3.5 Diagram Alir Hitung Jumlah Perkolom Matrik Perbandingan

Berikut merupakan tahapan perhitungan jumlah perkolom matrik perbandingan seperti pada Gambar 3.5.

1. Diberikan *input* berupa inialisasi variabel untuk menampung hasil dari jumlah perkolom matrik perbandingan.
2. Melakukan perulangan dalam perhitungan penjumlahan setiap kolom matrik perbandingan sebanyak *j* yang mewakili kolom matrik

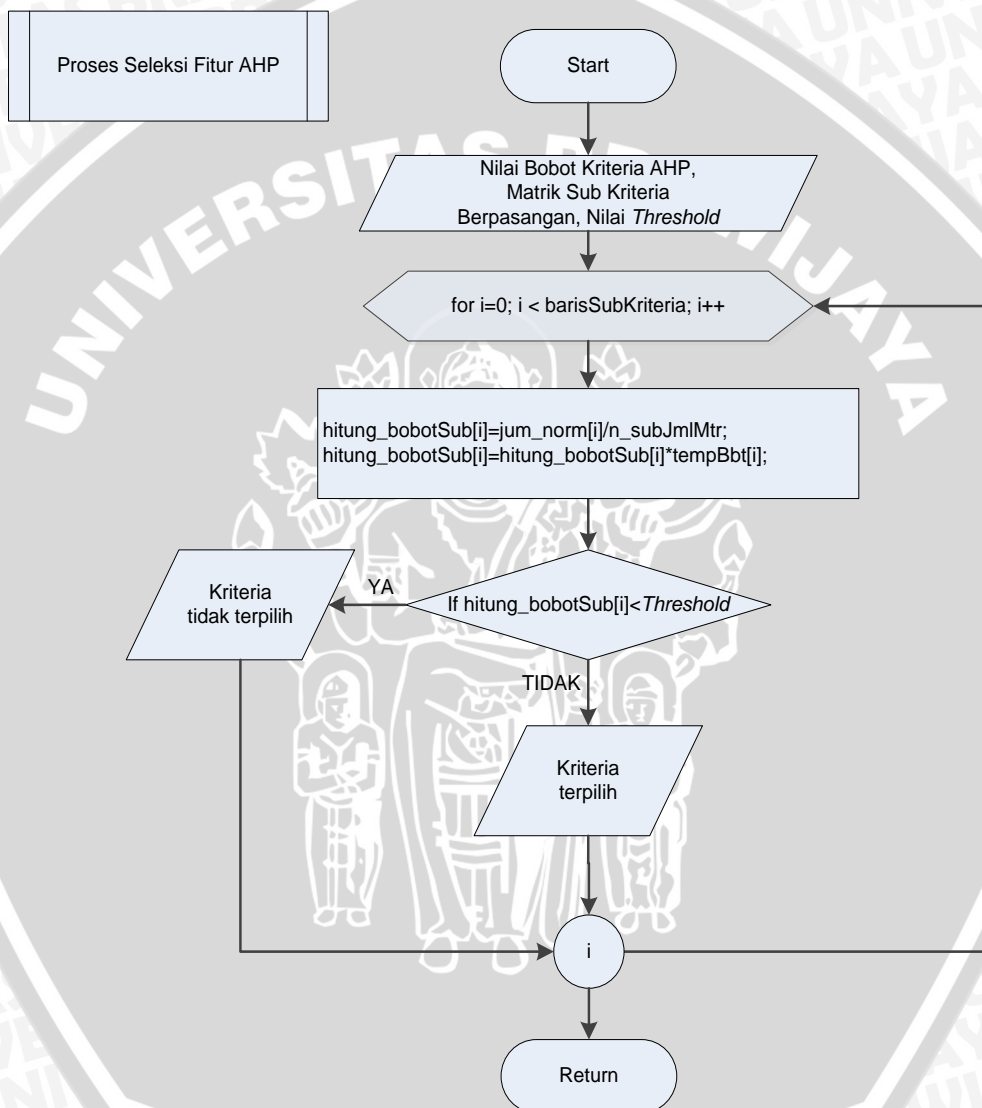


perbandingan, dan i yang mewakili sebanyak baris matrik perbandingan.

3. *Output* yang dihasilkan berupa hasil dari penjumlahan perkolom matrik perbandingan.

3.3.1.4 Proses Seleksi Fitur AHP dengan Threshold

Proses seleksi fitur AHP dengan menggunakan nilai *threshold* seperti pada Gambar 3.6.



Gambar 3.6 Diagram Alir Seleksi fitur AHP

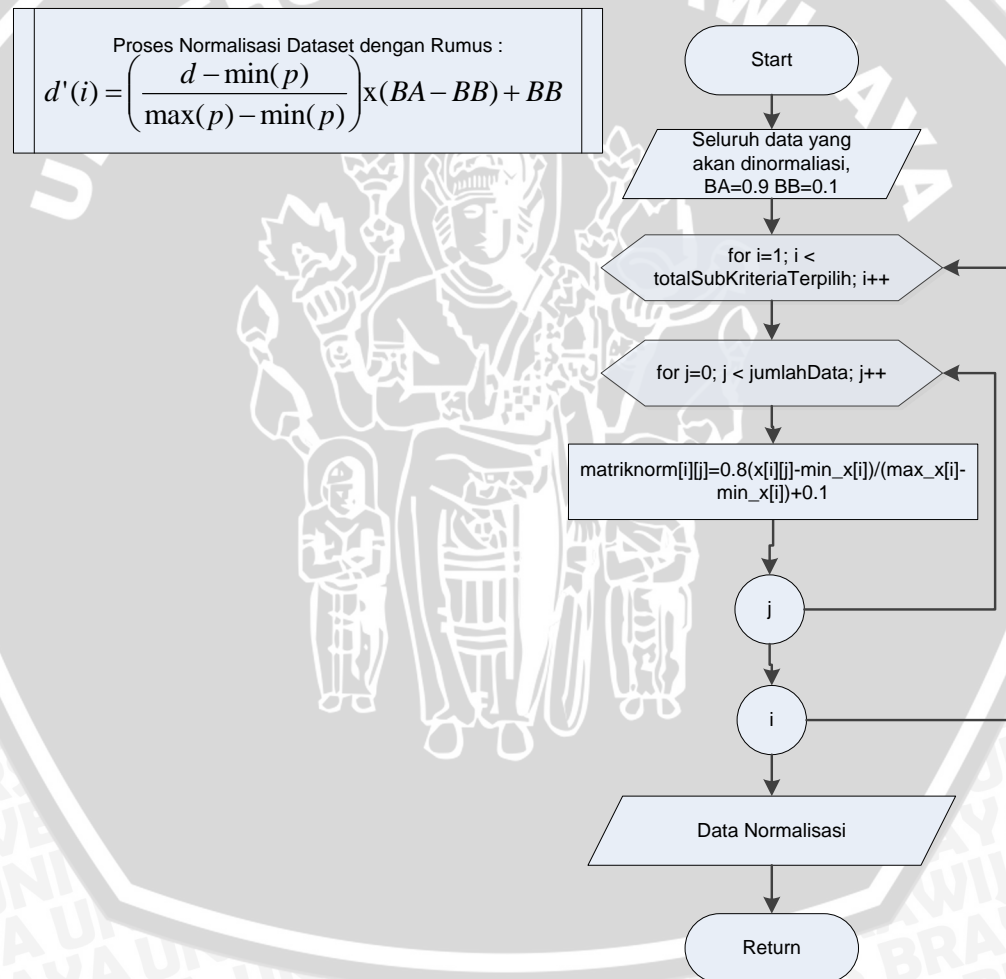
Berikut merupakan diagram alir seleksi fitur menggunakan AHP seperti pada Gambar 3.6.

1. Diawali dengan *input* berupa nilai *threshold* yang akan digunakan untuk melakukan seleksi fitur, nilai bobot kriteria utama AHP dan matrik perbandingan dari sub kriteria AHP.

2. Selanjutnya menghitung bobot dari setiap sub kriteria yakni mengalikannya dengan bobot kriteria utama AHP. Dilakukan perulangan sebanyak jumlah sub kriteria.
3. Selanjutnya terdapat sebuah *decision* untuk melakukan seleksi fitur. Jika nilai bobot dari sub kriteria kurang dari nilai *threshold* yang ditentukan, maka akan masuk ke dalam kriteria tidak terpilih. Jika nilai bobot dari sub kriteria lebih dari sama dengan nilai *threshold* yang ditentukan maka akan masuk ke dalam kriteria terpilih.
4. *Output* yang dihasilkan berupa sub kriteria terpilih dan sub kriteria tidak terpilih yang ditentukan berdasarkan besaran nilai *threshold* AHP.

3.3.1.5 Proses Normalisasi Dataset

Proses normalisasi dataset seperti yang terdapat pada Gambar 3.7.



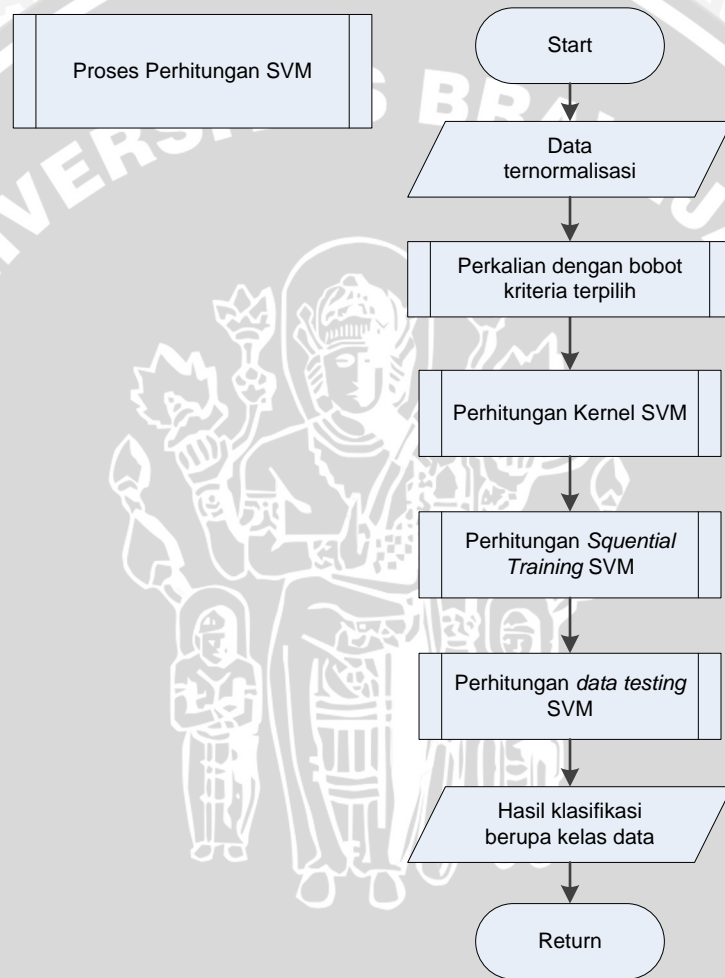
Gambar 3.7 Diagram Alir Normalisasi Dataset

Berikut merupakan tahap dari proses normalisasi dataset seperti yang digambarkan pada Gambar 3.7.

1. Melakukan *input* seluruh *dataset* pemohon beasiswa yang akan dinormalisasi.
2. Perhitungan normalisasi diawali dengan melakukan perulangan sebanyak sub kriteria terpilih dan sebanyak jumlah data.
3. Melakukan normalisasi
4. Hasil akhir berupa data yang telah ternormalisasi.

3.3.1.6 Proses Perhitungan Metode SVM

Tahapan yang dilakukan untuk proses perhitungan dengan metode SVM seperti pada Gambar 3.8.



Gambar 3.8 Diagram Alir Penghitungan Metode SVM

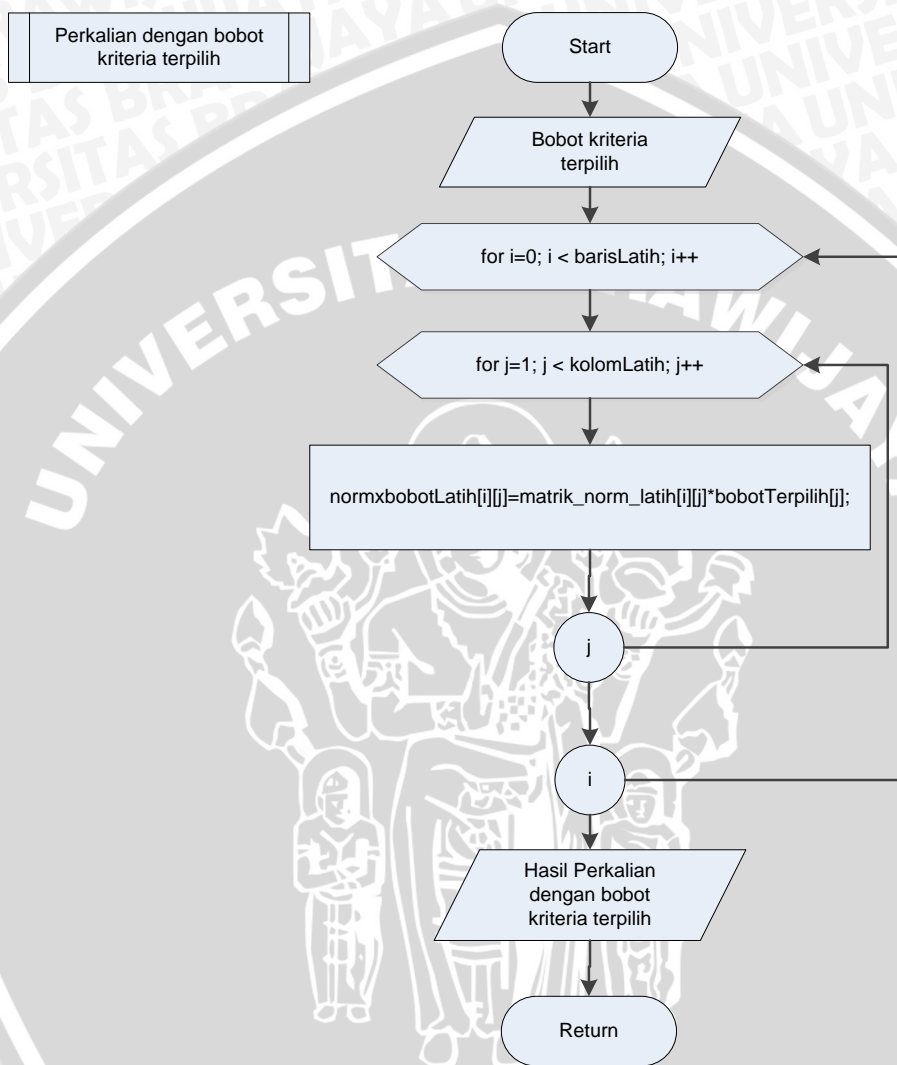
Pada Gambar 3.8 mengenai tahap yang dilakukan dalam proses penghitungan dengan metode SVM dapat dijelaskan sebagai berikut :

1. Memberikan *input* berupa dataset yang telah dinormalisasi.
2. Dataset yang telah dinormalisasi dikalikan dengan nilai bobot kriteria terpilih.
3. Melakukan penghitungan *kernel*.
4. Melakukan penghitungan data *training* dengan metode *Sequential Training SVM*.

5. Selanjutnya melakukan penghitungan data *testing*.
6. Hasil akhir berupa klasifikasi kelas data.

3.3.1.7 Proses Perkalian dengan Bobot Sub Kriteria Terpilih

Proses perkalian dengan bobot sub kriteria terpilih seperti pada Gambar 3.9.



Gambar 3.9 Diagram Alir Perkalian Dengan Bobot Sub Kriteria Terpilih

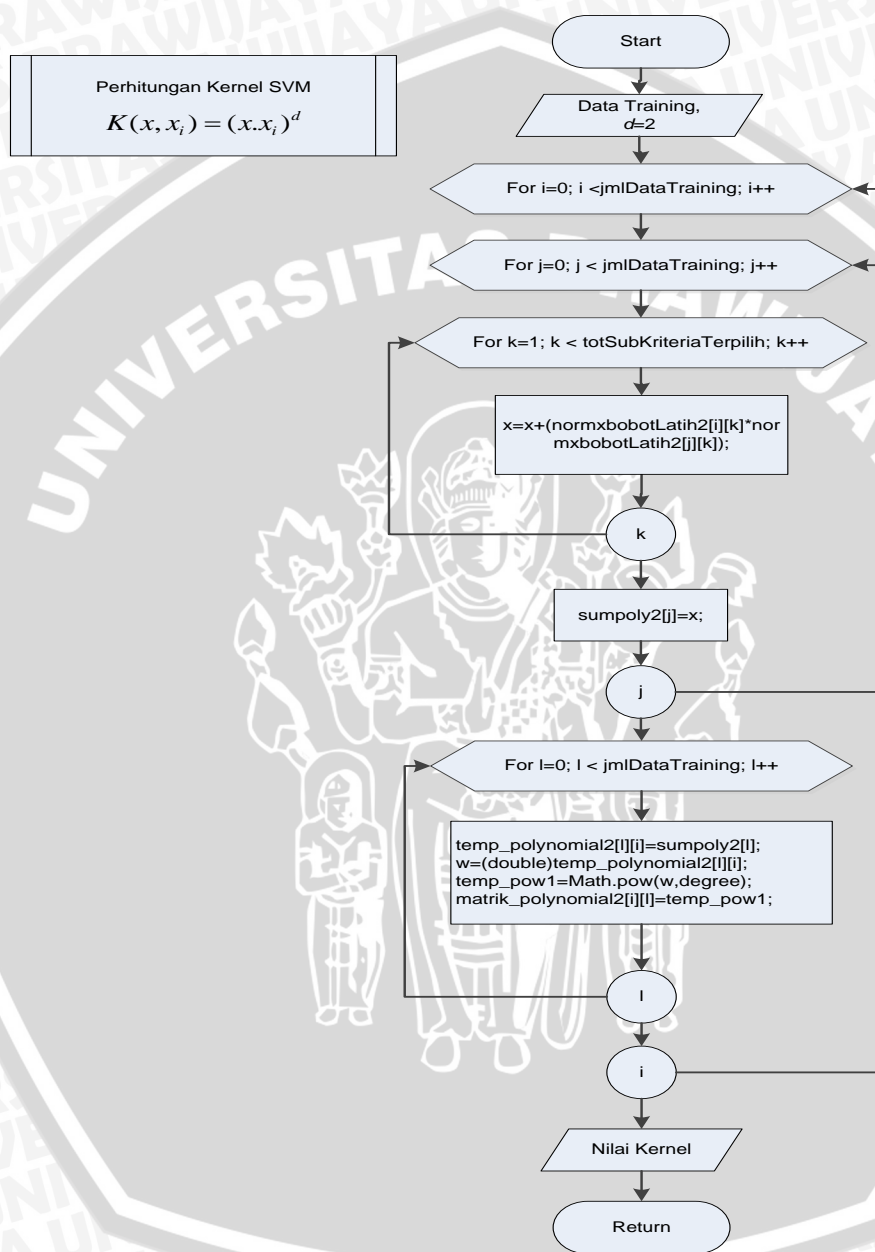
Berikut merupakan tahapan perkalian dengan bobot sub kriteria terpilih seperti pada Gambar 3.9.

1. Diberikan *input* berupa bobot kriteria terpilih yang telah diseleksi menggunakan nilai *threshold* yang ditentukan.
2. Dilakukan perulangan sebanyak baris data latih dan kolom data latih.
3. Melakukan penghitungan hasil dari normalisasi *dataset* yang dikalikan dengan bobot sub kriteria terpilih.

4. *Output* yang diberikan berupa hasil perkalian matrik ternormalisasi dengan bobot sub kriteria terpilih.

3.3.1.8 Proses Perhitungan Kernel SVM

Tahapan yang dilakukan untuk proses perhitungan *kernel* SVM dapat dilihat pada Gambar 3.10.



Gambar 3.10 Diagram Alir Perhitungan *Kernel* SVM

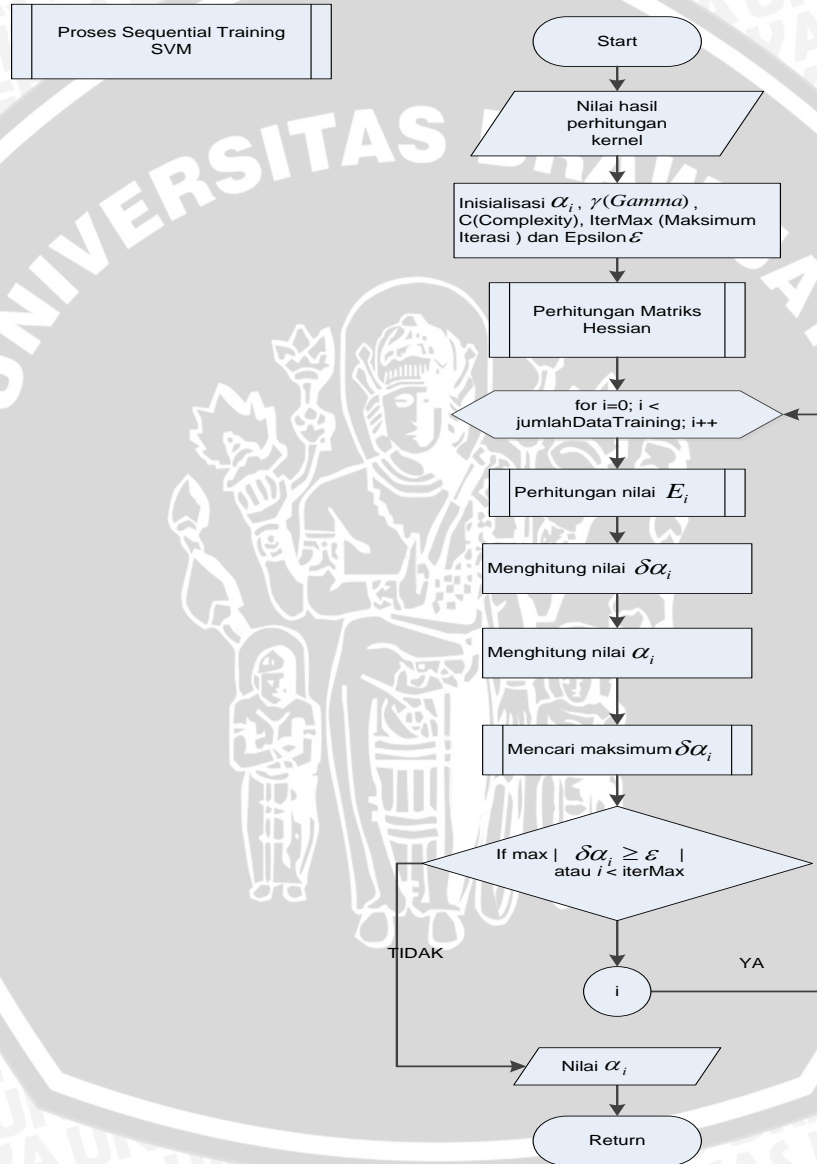
Pada Gambar 3.10 merupakan tahapan yang dilakukan untuk perhitungan kernel SVM yang akan dijelaskan sebagai berikut :

1. Penghitungan *kernel* SVM diawali dengan memberikan inputan berupa data *training* yang telah dinormalisasi.

2. Perulangan dilakukan sebanyak jumlah data *training* dan sebanyak sub kriteria terpilih.
3. Selanjutnya dilakukan proses perhitungan *kernel* menggunakan *kernel Polynomial*.
4. Hasil akhir berupa nilai kernel.

3.3.1.9 Perhitungan dengan Metode Sequential Training SVM

Proses perhitungan dengan metode *Sequential Training* seperti pada Gambar 3.11.



Gambar 3.11 Diagram Alir Proses Sequential Training SVM

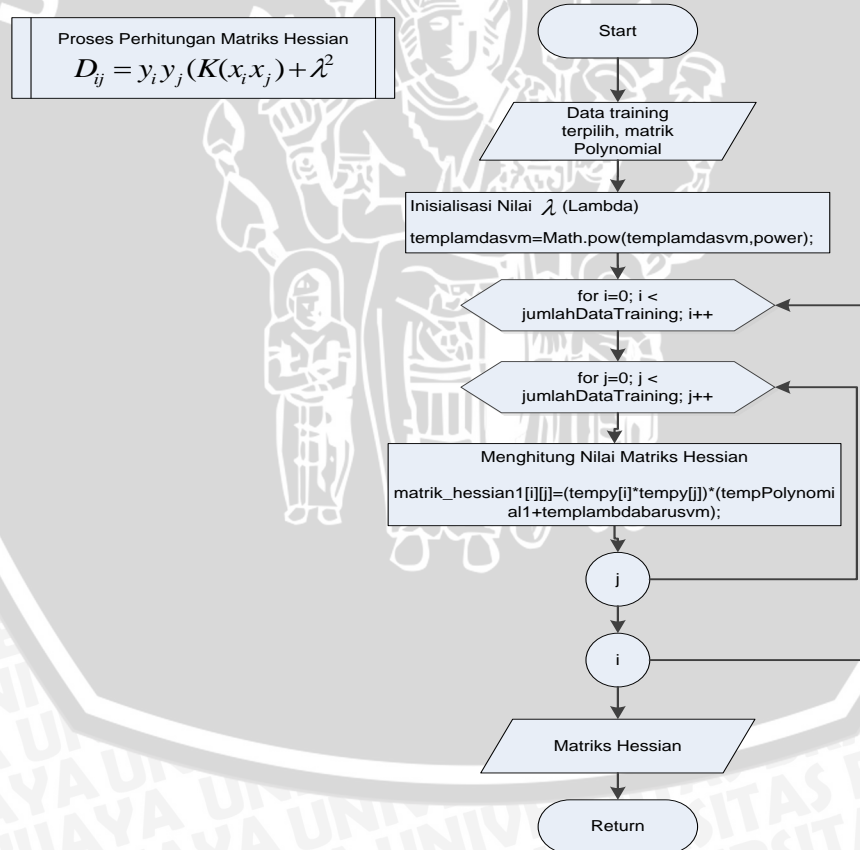
Tahapan perhitungan dengan Metode *Sequential Training* pada Gambar 3.11 akan dijelaskan sebagai berikut :

1. Memberikan inputan berupa hasil perhitungan *kernel*.

2. Inisialisasi parameter SVM seperti nilai α_i , γ (Gamma), ϵ (Epsilon), IterMax dan nilai C(Complexity).
3. Perhitungan Matrik Hessian seperti pada Persamaan 2-24.
4. Melakukan perulangan sebanyak jumlah data *training*.
5. Perhitungan nilai E_i seperti pada Persamaan 2.25.
6. Menghitung nilai $\delta\alpha_i$ seperti pada Persamaan 2.26.
7. Menghitung nilai α_i seperti pada Persamaan 2.27.
8. Iterasi dilakukan sampai mencapai nilai konvergen yakni jika maksimum ($|\delta\alpha_i| < \epsilon$) atau mencapai nilai iterMax.
9. Hasil akhir berupa nilai dari variabel α_i yang akan digunakan untuk menentukan *Support Vector*.

3.3.1.10 Perhitungan Matriks Hessian

Tahapan dalam perhitungan Matriks Hessian seperti pada Gambar 3.12.



Gambar 3.12 Diagram Alir Perhitungan Matrix Hessian

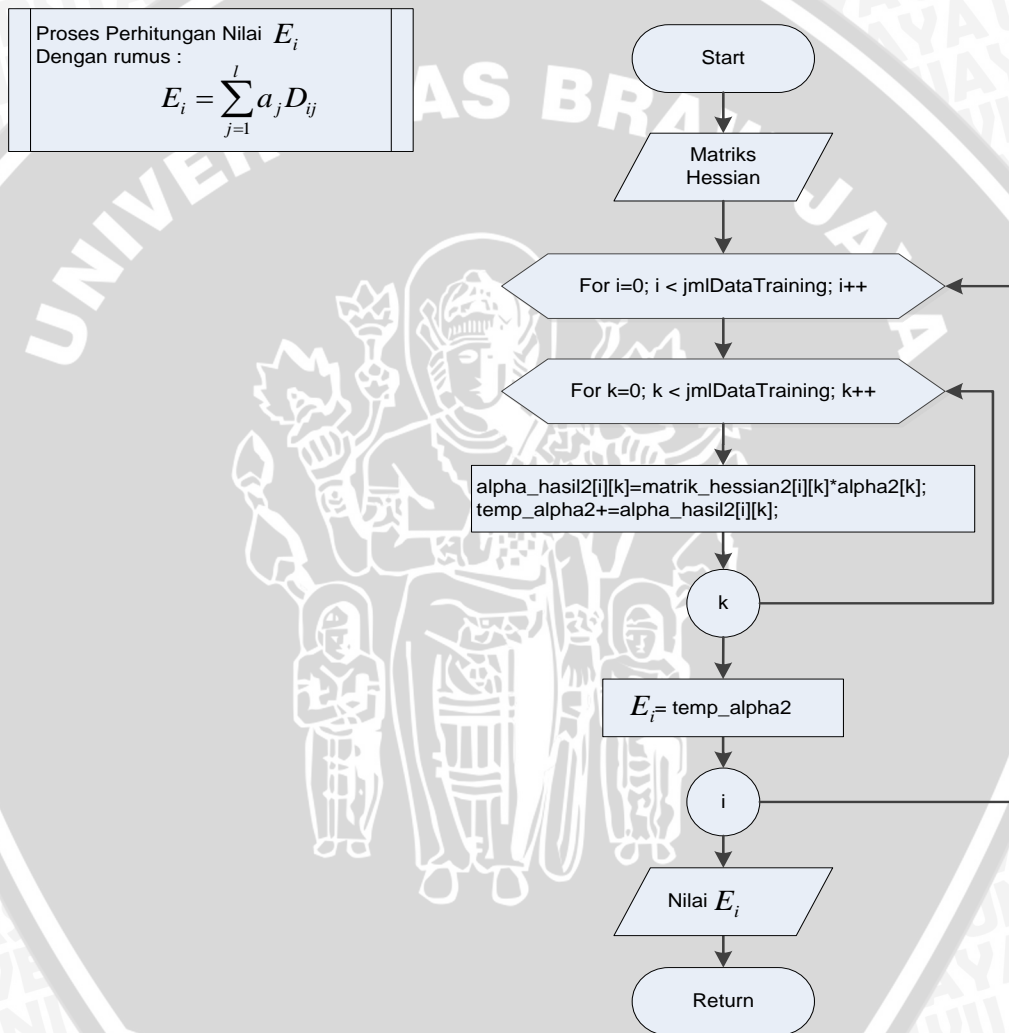
Tahapan dalam perhitungan Matriks Hessian seperti pada Gambar 3.12 dijelaskan sebagai berikut :



1. Memberikan inputan berupa data *training* terpilih.
2. Melakukan inialisasi λ (*Lamda*).
3. Proses penghitungan Matriks Hessian seperti pada Persamaan 2.24.
4. Melakukan perulangan penghitungan Matriks Hessian sebanyak jumlah data training.
5. Jika seluruh perulangan telah selesai dilakukan maka akan menghasilkan *output* berupa nilai Matrks Hessian.

3.3.1.11 Perhitungan Nilai E_i

Proses perhitungan nilai E_i seperti yang ditunjukkan pada Gambar 3.13.



Gambar 3.13 Diagram Alir Proses Perhitungan Nilai E_i

Penjelasan mengenai proses perhitungan nilai E_i pada Gambar 3.13 sebagai berikut :

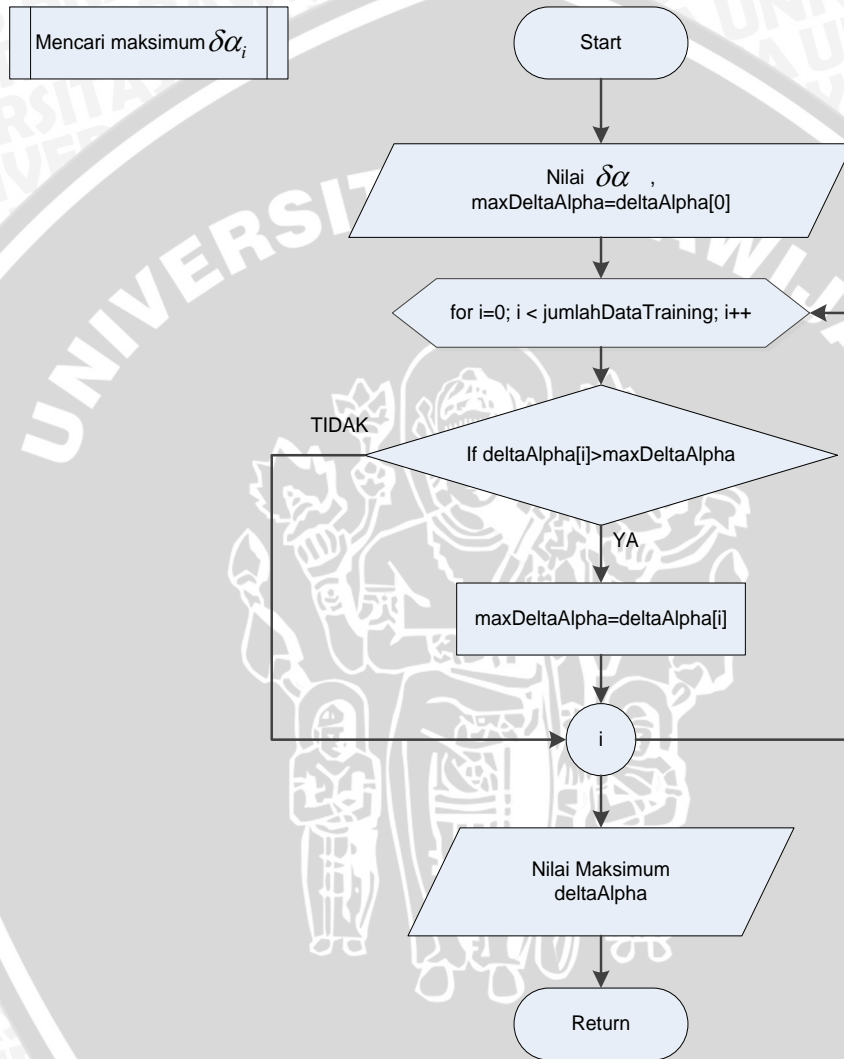
1. Memberikan input berupa data Matriks Hessian.
2. Inialisasi nilai $\alpha_i = 0$.



3. Melakukan penghitungan nilai E_i seperti pada Persamaan 2.25.
4. Melakukan perulangan sebanyak jumlah data *training*.
5. Setelah seluruh perulangan dilakukan maka akan menghasilkan *output* berupa nilai E_i .

3.3.1.12 Proses Perhitungan Nilai Maksimum Delta Alpha

Proses perhitungan nilai dari $\max(|\delta\alpha_i|)$ seperti pada Gambar 3.14.



Gambar 3.14 Diagram Alir Perhitungan Maximum Delta Alpha

Berikut merupakan tahap proses perhitungan nilai maksimum $\delta\alpha$ seperti pada Gambar 3.14.

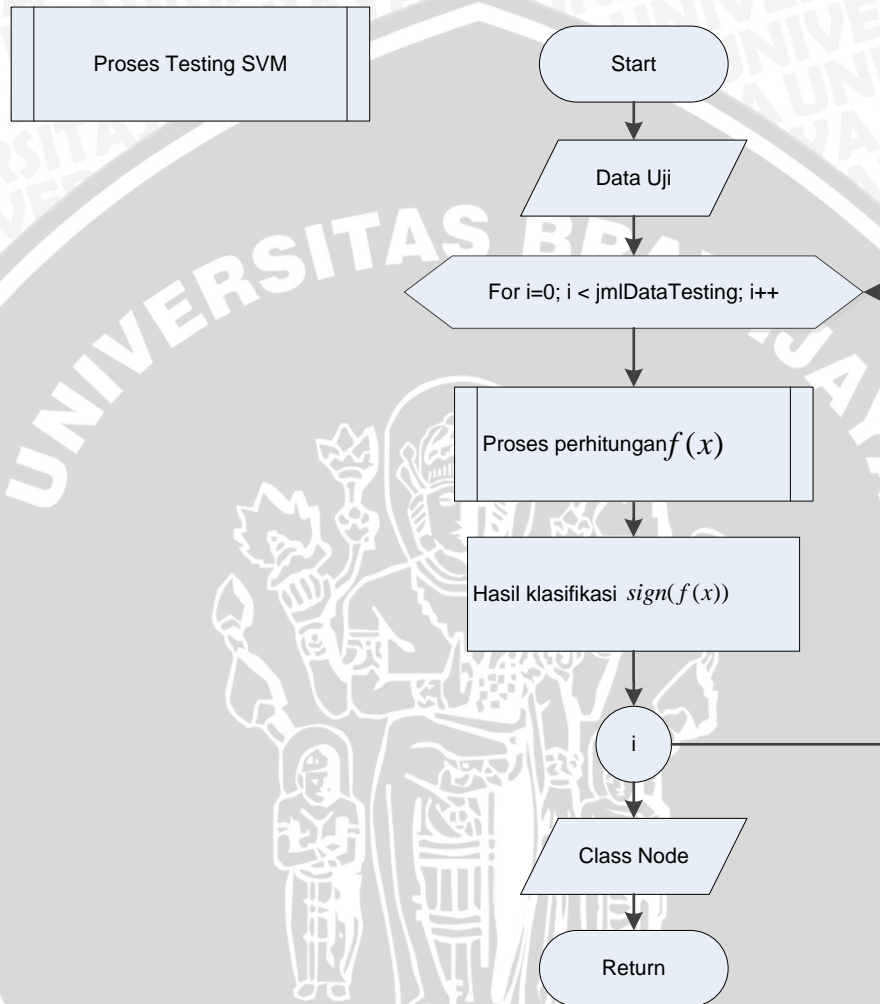
1. Diberikan suatu *inputan* berupa nilai $\delta\alpha$ dan inisialisasi variabel untuk menampung hasil maksimum $\delta\alpha$.
2. Dilakukan perulangan sebanyak jumlah data *training*.
3. Terdapat suatu *decission* yang akan melakukan pengecekan untuk setiap isi *array* yang menampung nilai $\delta\alpha$. Jika nilai $\delta\alpha$ pada perulangan ke-*i*

lebih besar dari nilai $\delta\alpha$ sebelumnya, maka nilai $\delta\alpha$ ke- i yang akan diambil sebagai nilai maksimum $\delta\alpha$ yang baru. Begitupun seterusnya.

4. *Output* yang dihasilkan berupa nilai maksimum $\delta\alpha$.

3.3.1.13 Proses *Testing* dengan Metode SVM

Proses perhitungan data *testing* dapat dilihat seperti pada Gambar 3.15.



Gambar 3.15 Diagram Alir Proses *Testing*

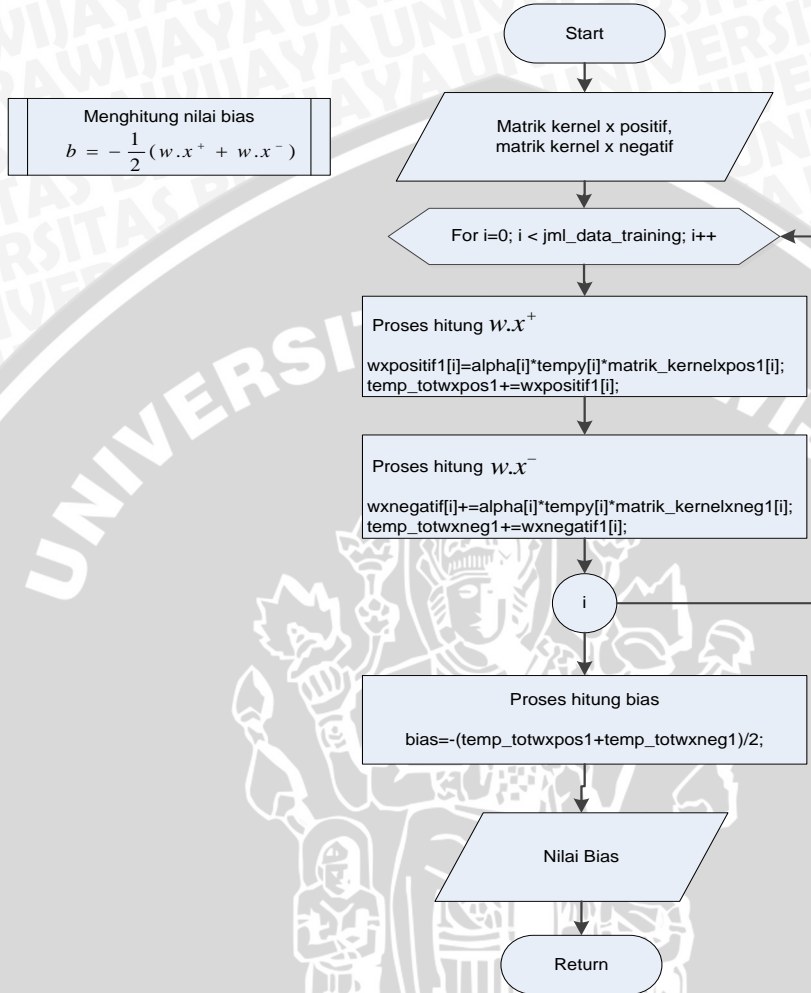
Berikut merupakan tahapan dari proses data *testing* seperti pada Gambar 3.15:

1. Memberikan *input* berupa data bobot kriteria
2. Melakukan pengujian data dengan fungsi *hyperplane* yang dihasilkan dari perhitungan metode *Sequential Training*.
3. Melakukan penghitungan fungsi $f(x)$.
4. Selanjutnya menghitung hasil klasifikasi dengan fungsi $sign(f(x))$.
5. Proses perulangan terhadap proses 3 dan 4 sebanyak jumlah data *testing*.

6. Hasil akhir berupa *predicted class* atau *class node* dari setiap data *testing*.

3.3.1.14 Proses Perhitungan Nilai Bias

Proses perhitungan nilai bias seperti pada Gambar 3.16.



Gambar 3.16 Proses Perhitungan Nilai Bias

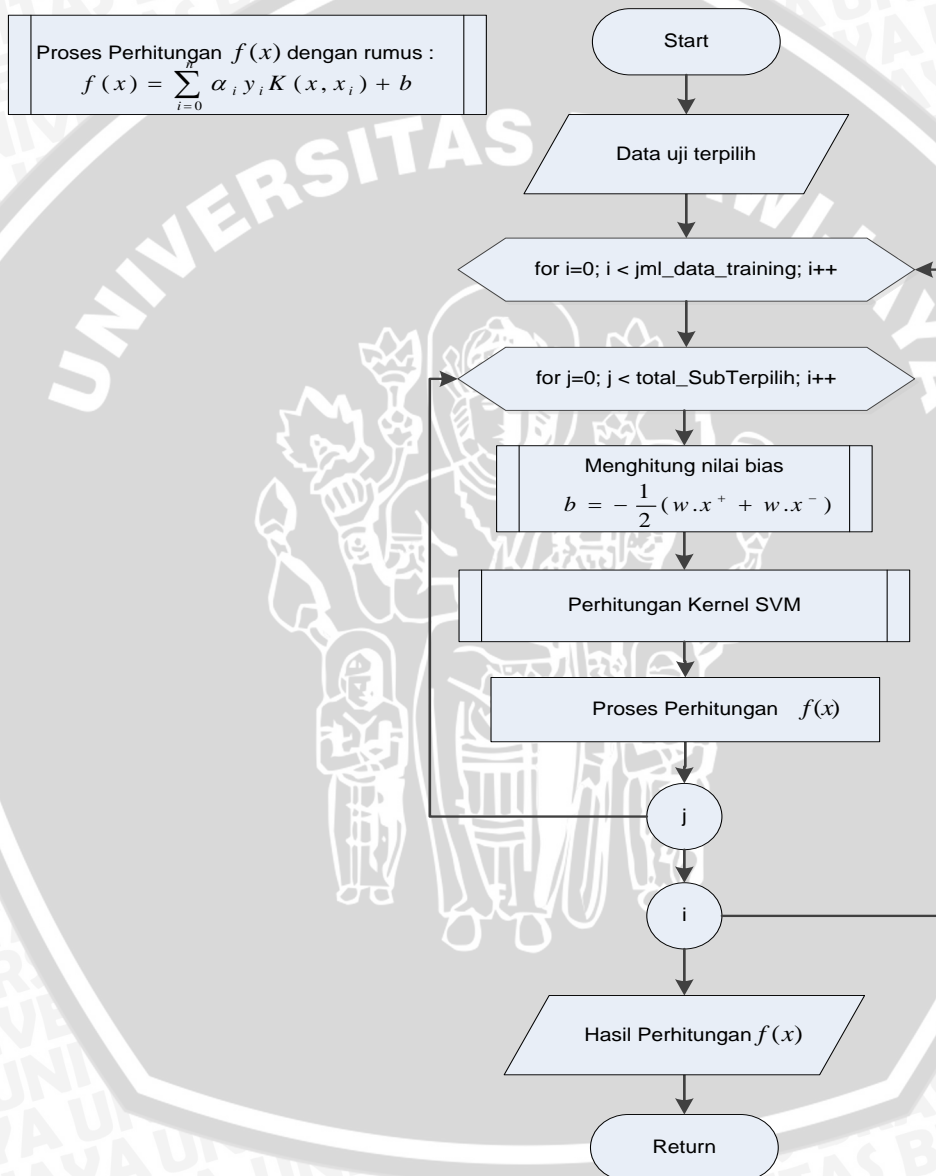
Berikut merupakan tahap proses perhitungan nilai bias seperti pada Gambar 3.16.

1. Diberikan *input* berupa matrik kernel x positif dan matrik kernel x negatif.
2. Dilakukan perulangan sejumlah data *training*.
3. Selanjutnya dilakukan perhitungan untuk mencari nilai $w \cdot x^+$ dan $w \cdot x^-$ dan menampung hasil penjumlahan dari $w \cdot x^+$ dan hasil penjumlahan dari $w \cdot x^-$ ke dalam suatu variabel.
4. Selanjutnya menghitung nilai *bias*.
5. *Output* yang dihasilkan berupa nilai bias.

3.3.1.15 Proses Perhitungan Nilai $f(x)$

Proses perhitungan nilai $f(x)$ dapat dilihat pada Gambar 3.17. berikut merupakan tahap dari proses perhitungan nilai $f(x)$ pada Gambar 3.17 :

1. Memberikan input berupa data uji terpilih.
2. Perhitungan nilai $K(x_i, x_j)$ terhadap data testing.
3. Perhitungan nilai $f(x) = \alpha_i y_i K(x_i, x_j)$ dimana y_i adalah kelas dari data testing.
4. Output berupa nilai $f(x)$ dari data testing.

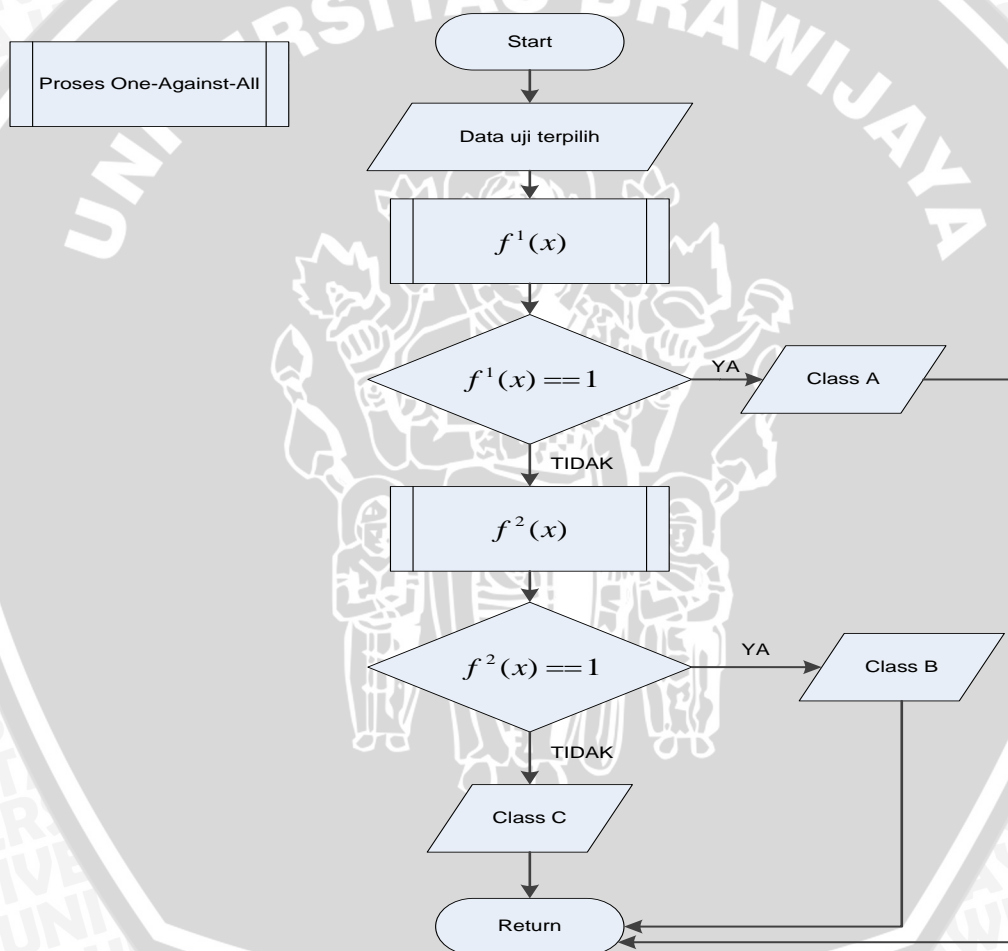


Gambar 3.17 Proses Perhitungan Nilai $f(x)$

3.3.1.16 Proses Perhitungan One-Against-All

Proses perhitungan *one-against-all* digunakan untuk mengatasi kelemahan metode SVM yang hanya dapat melakukan klasifikasi dua kelas (+1, -1)

sehingga dapat menjadi *multi class SVM* karena dalam penelitian ini menggunakan tiga kelas untuk klasifikasi yakni kelas A untuk kategori sangat layak, kelas B untuk kategori layak dan kelas C untuk kategori tidak layak. Dalam pemisahan menjadi tiga kelas akan dibuat dua fungsi klasifikasi. Fungsi yang pertama adalah untuk memisahkan antara kelas A (+1) lalu kelas B dan C (-1). Fungsi kedua untuk memisahkan kelas B (+1) dan kelas C (-1). Dalam penggunaan kedua fungsi tersebut, jika hasil klasifikasi pada fungsi pertama bernilai positif (+1) maka data tersebut akan masuk ke dalam kelas A. Sedangkan jika pada fungsi pertama bernilai (-1) maka akan dilanjutkan perhitungan ke dalam fungsi kedua untuk dilakukan pengecekan jika pada fungsi kedua bernilai (+1) maka akan dimasukkan ke dalam kelas B dan jika pada fungsi kedua bernilai (-1) maka akan dimasukkan ke dalam kelas C. Untuk proses perhitungan *one-against-all* dapat dilihat pada Gambar 3.13.



Gambar 3.18 Proses Perhitungan One-Against-All

3.3.2 Perhitungan Manual

Perhitungan manual menggunakan data *sample* yang didapatkan dari lembaga GNOTA Kediri. Data tersebut selanjutnya akan dinormalisasi. Data *sample* akan digunakan sebagai dataset *training* dan dataset *testing*.

Pada Tabel 3.2 merupakan sampel *dataset* yang akan digunakan untuk *dataset training* dan *dataset testing*. Secara keseluruhan sampel *dataset* dapat dilihat pada Lampiran K. NP merupakan inisial dari nomor peserta. P merupakan inisial dari nama sub kriteria. kelas merupakan kelas aktual dari data tersebut dimana kelas A merupakan kategori untuk penerima beasiswa “Sangat Layak”, B untuk kategori “Layak” dan C untuk kategori “Tidak Layak”.

Tabel 3.2 Sample Dataset

NO	NP	P1	P2	P3	P4	P5	...	P47	P48	P49	P50	kelas
1	211	0	4	0	0	0	...	0	0	0	1	A
2	174	0	4	0	0	0	...	0	0	2	0	A
...
110	91	0	0	0	2	0	...	0	0	0	1	C
111	198	0	0	0	0	1	...	0	0	0	1	C

3.3.2.1 Perhitungan Manual Proses AHP

Perhitungan AHP diawali dengan penentuan nilai matriks perbandingan berpasangan seperti yang terlihat pada Tabel 3.3. C merupakan inisial dari nama kriteria utama. Struktur hierarki AHP secara keseluruhan dapat dilihat pada Lampiran I. Berikut merupakan contoh pengitungan matriks perbandingan berpasangan seperti pada Persamaan 2.1 :

$$\frac{1}{C_{32}} = \frac{1}{4} = 0.25$$

Tabel 3.3 Matriks Perbandingan Berpasangan

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13
C1	1	6	4	3	3	5	5	4	4	4	5	6	4
C2	0.16	1	4	3	3	4	4	3	3	3	2	3	3
C3	0.25	0.25	1	3	3	5	5	5	5	5	3	3	5
C4	0.33	0.33	0.33	1	3	4	4	4	3	3	2	2	4
C5	0.33	0.33	0.33	0.33	1	1	2	3	3	3	2	2	3
C6	0.2	0.25	0.20	0.25	1	1	4	3	3	4	5	5	6
C7	0.2	0.25	0.20	0.25	0.5	0.25	1	1	1	1	2	2	2
C8	0.25	0.33	0.20	0.25	0.33	0.33	1	1	2	1	3	4	2
C9	0.25	0.33	0.20	0.33	0.33	0.33	1	0.5	1	2	3	4	1
C10	0.25	0.33	0.20	0.33	0.33	0.25	1	1	0.5	1	3	4	2
C11	0.2	0.5	0.33	0.5	0.5	0.2	0.5	0.33	0.33	0.33	1	3	5
C12	0.16	0.33	0.33	0.5	0.5	0.2	0.5	0.25	0.25	0.25	0.33	1	6
C13	0.25	0.33	0.20	0.25	0.33	0.16	0.5	0.5	1	0.5	0.2	0.16	1
Σ	3.85	10.58	11.53	13.00	16.83	21.73	29.50	26.58	27.08	28.08	31.53	39.17	44

Keterangan :

- C1 = Penghasilan
- C2 = Jumlah Tanggungan
- C3 = Pekerjaan
- C4 = Daya Listrik
- C5 = Biaya Listrik
- C6 = Jumlah Rumah
- C7 = Status Rumah
- C8 = Kondisi Dinding
- C9 = Kondisi Lantai
- C10 = PBB
- C11 = Jumlah Motor
- C12 = Jumlah Mobil
- C13 = Program Beasiswa

Setelah didapatkan nilai dari matriks perbandingan berpasangan, selanjutnya adalah melakukan normalisasi matriks perbandingan berpasangan dengan menggunakan Persamaan 2.2. Hasil dari normalisasi matriks perbandingan berpasangan dapat dilihat pada Tabel 3.4.

Berikut merupakan contoh penghitungan untuk normalisasi matriks.

$$Norm_{11} = \frac{A_{11}}{\text{jumlah perkolom tiap kriteria}}$$

$$Norm_{11} = \frac{1}{3.85} = 0,26$$

A_{11} merupakan nilai bobot pada matrik perbandingan berpasangan pada baris pertama kolom pertama. Setelah didapatkan nilai hasil dari normalisasi matriks perbandingan berpasangan, selanjutnya dijumlahkan perbaris seluruh hasil dari normalisasi seperti contoh perhitungan berikut :

$$\begin{aligned} \sum norm &= 0.26 + 0.57 + 0.35 + 0.23 + 0.18 + 0.23 + 0.17 + 0.15 + 0.15 \\ &+ 0.14 + 0.16 + 0.15 + 0.09 \\ &= 2.83 \end{aligned}$$

Tabel 3.4 Normalisasi Matrik

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	Σ
C1	0.26	0.57	0.35	0.23	0.18	0.23	0.17	0.15	0.15	0.14	0.16	0.15	0.09	2.83
C2	0.04	0.09	0.35	0.23	0.18	0.18	0.14	0.11	0.11	0.11	0.06	0.08	0.07	1.75
C3	0.06	0.02	0.09	0.23	0.18	0.23	0.17	0.19	0.18	0.18	0.10	0.08	0.11	1.82
C4	0.09	0.03	0.03	0.08	0.18	0.18	0.14	0.15	0.11	0.11	0.06	0.05	0.09	1.30
C5	0.09	0.03	0.03	0.03	0.06	0.05	0.07	0.11	0.11	0.11	0.06	0.05	0.07	0.86
C6	0.05	0.02	0.02	0.02	0.06	0.05	0.14	0.11	0.11	0.14	0.16	0.13	0.14	1.14
C7	0.05	0.02	0.02	0.02	0.03	0.01	0.03	0.04	0.04	0.04	0.06	0.05	0.05	0.46
C8	0.06	0.03	0.02	0.02	0.02	0.02	0.03	0.04	0.07	0.04	0.10	0.10	0.05	0.59
C9	0.06	0.03	0.02	0.03	0.02	0.02	0.03	0.02	0.04	0.07	0.10	0.10	0.02	0.56
C10	0.06	0.03	0.02	0.03	0.02	0.01	0.03	0.04	0.02	0.04	0.10	0.10	0.05	0.54

C11	0.05	0.05	0.03	0.04	0.03	0.01	0.02	0.01	0.01	0.01	0.03	0.08	0.11	0.48
C12	0.04	0.03	0.03	0.04	0.03	0.01	0.02	0.01	0.01	0.01	0.01	0.03	0.14	0.40
C13	0.06	0.03	0.02	0.02	0.02	0.01	0.02	0.02	0.04	0.02	0.01	0.00	0.02	0.28
Σ	1	1	1	1	1	1	1	1	1	1	1	1	1	

Selanjutnya, melakukan penghitungan bobot setiap kriteria dengan membagi hasil penjumlahan baris normalisasi dengan banyaknya kriteria seperti pada Persamaan 2.3. Hasil perhitungan bobot prioritas dapat dilihat pada Tabel 3.5. Berikut contoh penghitungan bobot untuk setiap kriteria.

$$C1 = \frac{2,83}{13} = 0,22$$

Tabel 3.5 Bobot Kriteria

C	Kriteria	Bobot
C1	Penghasilan	0.22
C2	Jumlah Tanggungan	0.13
C3	Pekerjaan	0.14
C4	Daya Listrik	0.10
C5	Biaya Listrik	0.07
C6	Jumlah Rumah	0.09
C7	Status Rumah	0.04
C8	Kondisi Dinding	0.05
C9	Kondisi Lantai	0.04
C10	PBB	0.04
C11	Jumlah Motor	0.04
C12	Jumlah Mobil	0.03
C13	Program Beasiswa	0.02

Selanjutnya melakukan penghitungan untuk mendapatkan nilai Ax dengan mengalikan setiap bobot kriteria dengan matriks kriteria berpasangan. Berikut adalah contoh penghitungan Ax untuk kriteria Penghasilan :

$$\begin{aligned}
 Ax &= (1 \times 0.22) + (6 \times 0.013) + (4 \times 0.14) + (3 \times 0.10) + (3 \times 0.07) + \\
 &\quad (5 \times 0.09) + (5 \times 0.04) + (4 \times 0.05) + (4 \times 0.04) + (4 \times 0.04) + \\
 &\quad (5 \times 0.04) + (6 \times 0.03) + (4 \times 0.02) \\
 &= 3.67
 \end{aligned}$$

Hasil penghitungan Ax terhadap seluruh kriteria dapat dilihat pada Tabel 3.6.



Tabel 3.6 Hasil Perhitungan Nilai Ax

C	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13
Bobot	0.22	0.13	0.14	0.10	0.07	0.09	0.04	0.05	0.04	0.04	0.04	0.03	0.02
<i>Ax</i>	3.67	2.34	2.30	1.61	1.01	1.32	0.53	0.69	0.65	0.63	0.56	0.45	0.33

Selanjutnya melakukan penghitungan untuk mendapatkan nilai λ_{maks} dengan menjumlahkan hasil bagi nilai *Ax* dengan bobot kriteria. Berikut hasil penghitungan untuk mendapatkan nilai λ_{maks} seperti pada Persamaan 2.4.

$$\lambda_{maks} = \frac{\left(\frac{3.67}{0.22} + \frac{2.34}{0.13} + \frac{2.30}{0.14} + \frac{1.61}{0.10} + \frac{1.01}{0.07} + \frac{1.32}{0.09} + \frac{0.53}{0.04} + \frac{0.69}{0.05} + \frac{0.65}{0.04} + \frac{0.63}{0.04} + \frac{0.56}{0.03} + \frac{0.45}{0.03} + \frac{0.33}{0.02} \right)}{13} = 14.526$$

Berikutnya melakukan perhitungan untuk mendapatkan nilai *Consistency Index (CI)* seperti pada Persamaan 2.5. Berikut adalah contoh penghitungan untuk mendapatkan nilai *CI*.

$$CI = \frac{(14.526 - 13)}{13 - 1} = 0.127$$

Setelah mendapatkan nilai *CI*, selanjutnya menghitung *Consistency Ratio (CR)* dengan menggunakan Persamaan 2.6. Berikut contoh penghitungan nilai *CR*.

$$CR = \frac{0,127}{1,56} = 0,08$$

Karena hasil dari nilai *CR* < 0.1 maka penghitungan tersebut dikatakan konsisten. Jika lebih dari 0.1 maka dikatakan hasil penghitungan tidak konsisten dan harus mengulangi perhitungan dari matriks perbandingan kriteria.

Setelah mendapatkan bobot untuk masing-masing kriteria maka selanjutnya adalah menghitung bobot untuk masing-masing sub kriteria. P merupakan inisial untuk setiap sub kriteria seperti pada Lampiran I.

1. Kriteria penghasilan (C1)

Sub kriteria penghasilan terdiri dari penghasilan Rp.0 (P1), Rp.1 s/d Rp.500.000 (P2), Rp.500.001 sd Rp.1000.000 (P3), Rp.1000.001 sd Rp.2.000.000 (P4), Rp.>2.000.000 (P5). Untuk matrik perbandingan berpasangan sub kriteria penghasilan seperti pada Tabel 3.7.

Tabel 3.7 Matrik Perbandingan Kriteria Penghasilan

	P1	P2	P3	P4	P5
P1	1.00	2.00	4.00	5.00	7.00
P2	0.50	1.00	4.00	4.00	6.00
P3	0.25	0.25	1.00	3.00	3.00



P4	0.20	0.25	0.33	1.00	2.00
P5	0.14	0.17	0.33	0.50	1.00
JUMLAH	2.09	3.67	9.67	13.50	19.00

Selanjutnya dilakukan normalisasi terhadap matriks perbandingan sub kriteria penghasilan seperti pada Persamaan 2.2.

$$Norm_{11} = \frac{1}{2.09} = 0.48$$

Hasil dari penghitungan normalisasi matriks perbandingan sub kriteria penghasilan dapat dilihat pada Tabel 3.8.

Tabel 3.8 Hasil Normalisasi Sub Kriteria Penghasilan

	P1	P2	P3	P4	P5	Σ
P1	0.48	0.55	0.41	0.37	0.37	2.18
P2	0.24	0.27	0.41	0.30	0.32	1.54
P3	0.12	0.07	0.10	0.22	0.16	0.67
P4	0.10	0.07	0.03	0.07	0.11	0.38
P5	0.07	0.05	0.03	0.04	0.05	0.24
JUMLAH	1	1	1	1	1	5

Dalam menentukan bobot dari sub kriteria penghasilan, dilakukan perkalian bobot kriteria penghasilan dengan bobot dari sub kriteria penghasilan. Hasil dari perhitungan tersebut seperti pada Tabel 3.9.

Berikut merupakan penghitungan untuk bobot kriteria P1.

$$P1 = \frac{2.18}{5} = 0.44$$

$$P1 = 0.44 \times 0.22 = 0.95$$

Tabel 3.9 Matriks Perbandingan Kriteria Penghasilan

Bobot Kriteria (C1)		0,22
P1	0.44	0.095
P2	0.31	0.067
P3	0.13	0.029
P4	0.08	0.016
P5	0.05	0.010
JUMLAH	1,00	

2. Kriteria Jumlah Tanggungan (C2)

Sub kriteria jumlah tanggungan terdiri dari >5 (P6), 3 sd 5 (P7), 0 sd 3 (P8).

Matrik perbandingan berpasangan untuk kriteria jumlah tanggungan seperti pada Tabel 3.10.



Tabel 3.10 Bobot Sub Kriteria Jumlah Tanggungan

	P6	P7	P8
P6	1	3	4
P7	0.33	1	3
P8	0.25	0.33	1
JUMLAH	1.58	4.33	8.00

Selanjutnya dilakukan normalisasi terhadap matriks perbandingan sub kriteria jumlah tanggungan seperti pada Persamaan 2.2.

$$Norm_{11} = \frac{1}{1.58} = 0.631$$

Hasil dari penghitungan normalisasi matriks perbandingan sub kriteria jumlah tanggungan dapat dilihat pada Tabel 3.11.

Tabel 3.11 Hasil Normalisasi Sub Kriteria Jumlah Tanggungan

	P6	P7	P8	JUMLAH
P6	0.631578947	0.692307692	0.5	1.82388664
P7	0.210526316	0.230769231	0.375	0.816295547
P8	0.157894737	0.076923077	0.125	0.359817814
JUMLAH	1	1	1	3

Dalam menentukan bobot dari sub kriteria jumlah tanggungan, dilakukan perkalian bobot kriteria jumlah tanggungan dengan bobot dari sub kriteria jumlah tanggungan. Hasil dari penghitungan tersebut seperti pada Tabel 3.12. berikut merupakan contoh penghitungan bobot kriteria untuk sub kriteria P6.

$$P6 = \frac{1.82388664}{3} = 0.607962213$$

$$P6 = 0.607962213 \times 0.13 = 0.081928881$$

Tabel 3.12 Bobot Sub Kriteria Jumlah Tanggungan

BOBOT KRITERIA (C2)		0,13
P6	0.607962213	0.081928881
P7	0.272098516	0.036667948
P8	0.119939271	0.016162995
JUMLAH	1	

3. Sub Kriteria Pekerjaan (C3)

Sub kriteria pekerjaan terdiri dari Tidak Bekerja (P9), Buruh (P10), Karyawan (P11), Wiraswasta (P12), Pengusaha (P13). Matriks



perbandingan berpasangan untuk sub kriteria pekerjaan dapat dilihat pada Tabel 3.13.

Tabel 3.13 Matrik Perbandingan Kriteria Pekerjaan

	P9	P10	P11	P12	P13
P9	1.00	3.00	5.00	6.00	7.00
P10	0.33	1.00	2.00	4.00	5.00
P11	0.20	0.50	1.00	3.00	4.00
P12	0.17	0.25	0.33	1.00	2.00
P13	0.14	0.20	0.25	0.50	1.00
JUMLAH	1.84	4.95	8.58	14.50	19.00

Selanjutnya dilakukan normalisasi terhadap matriks perbandingan sub kriteria pekerjaan seperti pada Persamaan 2.2.

$$Norm_{11} = \frac{1}{1.84} = 0.54$$

Hasil dari penghitungan normalisasi matriks perbandingan sub kriteria pekerjaan dapat dilihat pada Tabel 3.14.

Tabel 3.14 Hasil Normalisasi Sub Kriteria Pekerjaan

	P9	P10	P11	P12	P13	JUMLAH
P9	0.54	0.61	0.58	0.41	0.37	2.51
P10	0.18	0.20	0.23	0.28	0.26	1.15
P11	0.11	0.10	0.12	0.21	0.21	0.74
P12	0.09	0.05	0.04	0.07	0.11	0.35
P13	0.08	0.04	0.03	0.03	0.05	0.23
JUMLAH	1	1	1	1	1	

Dalam menentukan bobot dari sub kriteria pekerjaan, dilakukan perkalian bobot kriteria pekerjaan dengan bobot dari sub kriteria pekerjaan. Hasil dari penghitungan tersebut seperti pada Tabel 3.15. Berikut contoh penghitungan bobot sub kriteria pekerjaan untuk sub kriteria P9.

$$P9 = \frac{2.51}{5} = 0.502687$$

$$P9 = 0.502687 \times 0.14 = 0.070373$$

Tabel 3.15 Bobot Sub Kriteria Pekerjaan

BOBOT KRITERIA		0,14
P9	0.502687	0.070373
P10	0.230986	0.032337
P11	0.148693	0.020816



P12	0.070802	0.009912
P13	0.046833	0.006556
JUMLAH	1	

4. Kriteria Daya Listrik (C4)

Sub kriteria daya listrik terdiri dari 0w (P14), 220w (P15), 450w (P16), 900w (P17), 1399w (P18), 2200w (P19). Matriks perbandingan berpasangan untuk kriteria daya listrik seperti pada Tabel 3.16.

Tabel 3.16 Matriks Perbandingan Kriteria Daya Listrik

	P14	P15	P16	P17	P18	P19
P14	1.00	2.00	2.00	4.00	5.00	7.00
P15	0.50	1.00	3.00	4.00	6.00	6.00
P16	0.50	0.33	1.00	3.00	4.00	5.00
P17	0.25	0.25	0.33	1.00	3.00	3.00
P18	0.20	0.17	0.25	0.33	1.00	2.00
P19	0.14	0.17	0.20	0.33	0.50	1.00
JUMLAH	2.59	3.92	6.78	12.67	19.50	24.00

Selanjutnya dilakukan normalisasi terhadap matriks perbandingan sub kriteria daya listrik seperti pada Persamaan 2.2.

$$Norm_{11} = \frac{1}{2.59} = 0.39$$

Hasil dari penghitungan normalisasi matriks perbandingan sub kriteria daya listrik dapat dilihat pada Tabel 3.17.

Tabel 3.17 Hasil Normalisasi Sub Kriteria Daya Listrik

	P14	P15	P16	P17	P18	P19	JUMLAH
P14	0.39	0.51	0.29	0.32	0.26	0.29	2.06
P15	0.19	0.26	0.44	0.32	0.31	0.25	1.76
P16	0.19	0.09	0.15	0.24	0.21	0.21	1.08
P17	0.10	0.06	0.05	0.08	0.15	0.13	0.57
P18	0.08	0.04	0.04	0.03	0.05	0.08	0.32
P19	0.06	0.04	0.03	0.03	0.03	0.04	0.22
JUMLAH	1.00	1.00	1.00	1.00	1.00	1.00	6.00

Dalam menentukan bobot dari sub kriteria daya listrik, dilakukan perkalian bobot kriteria daya listrik dengan bobot dari sub kriteria daya listrik.

Hasil dari penghitungan tersebut seperti pada Tabel 3.18. Berikut contoh penghitungan bobot sub kriteria daya listrik untuk sub kriteria P14.

$$P_{14} = \frac{2.06}{5} = 0.343$$

$$P_{14} = 343 \times 0.10 = 0.034$$

Tabel 3.18 Bobot Sub Kriteria Daya Listrik

BOBOT KRITERIA		0,10
P14	0.343	0.034
P15	0.294	0.029
P16	0.179	0.018
P17	0.095	0.009
P18	0.053	0.005
P19	0.036793	0.003666

5. Sub kriteria Biaya Listrik (C5)

Sub kriteria biaya listrik terdiri dari Rp.0 (P20), Rp.1 sd 50.000 (P21), Rp. 50.001 sd Rp.100.000 (P22), Rp.100.001 sd Rp.150.000 (P23), Rp.>150.000 (P24). Matriks perbandingan berpasangan untuk kriteria biaya listrik seperti pada Tabel 3.19.

Tabel 3.19 Matrik Perbandingan Kriteria Biaya Listrik

	P20	P21	P22	P23	P24
P20	1.00	3.00	5.00	6.00	7.00
P21	0.33	1.00	3.00	4.00	5.00
P22	0.20	0.33	1.00	3.00	3.00
P23	0.17	0.25	0.33	1.00	2.00
P24	0.14	0.20	0.33	0.50	1.00
JUMLAH	1.84	4.78	9.67	14.50	18.00

Selanjutnya dilakukan normalisasi terhadap matriks perbandingan sub kriteria biaya listrik seperti pada Persamaan 2.2.

$$Norm_{11} = \frac{1}{1.84} = 0.39$$

Hasil dari penghitungan normalisasi matriks perbandingan sub kriteria biaya listrik dapat dilihat pada Tabel 3.20.

Tabel 3.20 Hasil Normalisasi Sub Kriteria Biaya Listrik

	P20	P21	P22	P23	P24	JUMLAH
P20	0.543	0.627	0.517	0.414	0.389	2.490
P21	0.181	0.209	0.310	0.276	0.278	1.254
P22	0.109	0.070	0.103	0.207	0.167	0.655

P23	0.090	0.052	0.034	0.069	0.111	0.357
P24	0.078	0.042	0.034	0.034	0.056	0.244
JUMLAH	1	1	1	1	1	

Dalam menentukan bobot dari sub kriteria biaya listrik, dilakukan perkalian bobot kriteria biaya listrik dengan bobot dari sub kriteria biaya listrik. Hasil dari penghitungan tersebut seperti pada Tabel 3.21. Berikut contoh penghitungan bobot sub kriteria biaya listrik untuk sub kriteria P20.

$$P_{20} = \frac{2.490}{5} = 0.497947$$

$$P_{20} = 0.497947 \times 0.07 = 0.032901$$

Tabel 3.21 Bobot Sub Kriteria Biaya Listrik

BOBOT KRITERIA		0,07
P20	0.497947	0.032901
P21	0.250784	0.01657
P22	0.131045	0.008659
P23	0.071453	0.004721
P24	0.04877	0.003222

6. Kriteria Jumlah Rumah (C6)

Sub kriteria jumlah rumah terdiri dari 0 (P25), 1 (P26), >1 (P27). Matriks kriteria berpasangan sub kriteria jumlah rumah seperti pada Tabel 3.22.

Tabel 3.22 Matriks Perbandingan Kriteria Jumlah Rumah

	P25	P26	P27
P25	1.00	5.00	6.00
P26	0.20	1.00	3.00
P27	0.17	0.33	1.00
JUMLAH	1.37	6.33	10.00

Selanjutnya dilakukan normalisasi terhadap matriks perbandingan sub kriteria jumlah rumah seperti pada Persamaan 2.2.

$$Norm_{11} = \frac{1}{1.37} = 0.39$$

Hasil dari penghitungan normalisasi matriks perbandingan sub kriteria jumlah rumah dapat dilihat pada Tabel 3.23.



Tabel 3.23 Hasil Normalisasi Sub Kriteria Jumlah Rumah

	P25	P26	P27	JUMLAH
P25	0.731707317	0.789473684	0.6	2.121181001
P26	0.146341463	0.157894737	0.3	0.6042362
P27	0.12195122	0.052631579	0.1	0.274582798
JUMLAH	1	1	1	

Dalam menentukan bobot dari sub kriteria jumlah rumah, dilakukan perkalian bobot kriteria jumlah rumah dengan bobot dari sub kriteria jumlah rumah. Hasil dari penghitungan tersebut seperti pada Tabel 3.24. Berikut contoh penghitungan bobot sub kriteria jumlah rumah untuk sub kriteria P25.

$$P_{25} = \frac{2.121181001}{3} = 0.40$$

$$P_{25} = 0.40 \times 0.07 = 0,02$$

Tabel 3.24 Bobot Sub Kriteria Jumlah Rumah

BOBOT KRITERIA		0,07
P25	0,40	0,02
P26	0,37	0,02
P27	0,23	0,01

7. Kriteria Status Rumah (C7)

Sub kriteria status rumah terdiri dari kontrak (P28), milik ortu (P29) dan milik sendiri (P30). Matriks perbandingan berpasangan kriteria status rumah seperti pada Tabel 3.25.

Tabel 3.25 Matriks Perbandingan Kriteria Status Rumah

	P28	P29	P30
P28	1,00	2,00	3,00
P29	0,50	1,00	2,00
P30	0,33	0,50	1,00
JUMLAH	1,83	3,50	6,00

Selanjutnya dilakukan normalisasi terhadap matriks perbandingan sub kriteria status rumah seperti pada Persamaan 2.2.

$$Norm_{11} = \frac{1}{1.83} = 0.76$$



Hasil dari penghitungan normalisasi matriks perbandingan sub kriteria status rumah dapat dilihat pada Tabel 3.26.

Tabel 3.26 Hasil Normalisasi Sub Kriteria Status Rumah

	P28	P29	P30	JUMLAH
P28	0.76	0.83	0.54	2.14
P29	0.13	0.14	0.38	0.65
P30	0.11	0.03	0.08	0.21
JUMLAH	1	1	1	

Dalam menentukan bobot dari sub kriteria status rumah, dilakukan perkalian bobot kriteria status rumah dengan bobot dari sub kriteria status rumah. Hasil dari penghitungan tersebut seperti pada Tabel 3.27. Berikut hasil penghitungan bobot sub kriteria status rumah untuk sub kriteria P28.

$$P28 = \frac{2.14}{3} = 0.09$$

$$P28 = 0.71 \times 0.09 = 0.06$$

Tabel 3.27 Bobot Sub Kriteria Status Rumah

BOBOT KRITEIRA		0.09
P28	0.71	0.06
P29	0.20	0.02
P30	0.09	0.01

8. Kriteria Kondisi Dinding (C8)

Sub kriteria kondisi dinding terdiri dari bambu (P31), papan kayu (P32) dan tembok (P33). Matriks perbandingan kriteria kondisi dinding dapat dilihat pada Tabel 3.28.

Tabel 3.28 Matriks Perbandingan Kriteria Kondisi Dinding

	P31	P32	P33
P31	1.00	5.00	7.00
P32	0.20	1.00	4.00
P33	0.14	0.25	1.00
JUMLAH	1.34	6.25	12.00

Selanjutnya dilakukan normalisasi terhadap matriks perbandingan sub kriteria kondisi dinding seperti pada Persamaan 2.2.

$$Norm_{11} = \frac{1}{1.34} = 0.74$$

Hasil dari penghitungan normalisasi matriks perbandingan sub kriteria kondisi dinding dapat dilihat pada Tabel 3.29.

Tabel 3.29 Hasil Normalisasi Sub Kriteria Kondisi Dinding

	P31	P32	P33	JUMLAH
P31	0.74	0.80	0.58	2.13
P32	0.15	0.16	0.33	0.64
P33	0.11	0.04	0.08	0.23
JUMLAH	1	1	1	3

Dalam menentukan bobot dari sub kriteria kondisi dinding, dilakukan perkalian bobot kriteria kondisi dinding dengan bobot dari sub kriteria kondisi dinding. Hasil dari penghitungan tersebut seperti pada Tabel 3.30. Berikut hasil penghitungan bobot sub kriteria kondisi dinding untuk sub kriteria P31.

$$P31 = \frac{2.13}{3} = 0.70$$

$$P31 = 0.70 \times 0.05 = 0.03$$

Tabel 3.30 Bobot Sub Kriteria Kondisi Dinding

BOBOT KRITERIA	0,05	
P31	0.709338	0.03229
P32	0.21409	0.00975
P33	0.076572	0.00349

9. Kriteria Kondisi Lantai (C9)

Sub kriteria kondisi lantai terdiri dari Tanah (P34), Tegel (P35) dan Keramik (P36). Matriks perbandingan berpasangan kriteria kondisi lantai dapat dilihat pada Tabel 3.31.

Tabel 3.31 Matriks Perbandingan Kriteria Kondisi Lantai

	P34	P35	P36
P34	1.00	5.00	6.00
P35	0.20	1.00	4.00
P36	0.17	0.25	1.00
JUMLAH	1.37	6.25	11.00



Selanjutnya dilakukan normalisasi terhadap matriks perbandingan sub kriteria kondisi lantai seperti pada Persamaan 2.2.

$$Norm_{11} = \frac{1.37}{1} = 0.731707317$$

Hasil dari penghitungan normalisasi matriks perbandingan sub kriteria kondisi lantai dapat dilihat pada Tabel 3.32.

Tabel 3.32 Hasil Normalisasi Sub Kriteria Kondisi Lantai

	P34	P35	P36	JUMLAH
P34	0.731707317	0.8	0.545454545	2.077161863
P35	0.146341463	0.16	0.363636364	0.669977827
P36	0.12195122	0.04	0.090909091	0.25286031
JUMLAH	1	1	1	3

Dalam menentukan bobot dari sub kriteria kondisi lantai, dilakukan perkalian bobot kriteria kondisi lantai dengan bobot dari sub kriteria kondisi lantai. Hasil dari penghitungan tersebut seperti pada Tabel 3.33. Berikut hasil penghitungan bobot sub kriteria kondisi lantai untuk sub kriteria P34.

$$P34 = \frac{2.077161863}{3} = 0.692387$$

$$P34 = 0.692387 \times 0.04 = 0.02958$$

Tabel 3.33 Bobot Sub Kriteria Kondisi Lantai

BOBOT KRITERIA		0.04
P34	0.692387	0.02958
P35	0.223326	0.00954
P36	0.084287	0.0036

10. Kriteria PBB(C10)

Sub kriteia PBB terdiri dari Rp.0 sd Rp.50.000 (P37), Rp.50.001 sd Rp.100.000 (P38), Rp. 100.001 sd Rp. 200.000 (P39), Rp. > 200.000 (P40). Matriks perbandingan berpasangan untuk kriteria PBB dapat dilihat pada Tabel 3.34.

Tabel 3.34 Matriks Perbandingan Kriteria PBB

	P37	P38	P39	P40
P37	1.00	3.00	4.00	6.00
P38	0.33	1.00	3.00	4.00



P39	0.25	0.33	1.00	3.00
P40	0.17	0.25	0.33	1.00
JUMLAH	1.75	4.58	8.33	14.00

Selanjutnya dilakukan normalisasi terhadap matriks perbandingan sub kriteria PBB seperti pada Persamaan 2.2.

$$Norm_{11} = \frac{1}{1.75} = 0.57$$

Hasil dari penghitungan normalisasi matriks perbandingan sub kriteria PBB dapat dilihat pada Tabel 3.35.

Tabel 3.35 Hasil Normalisasi Sub Kriteria PBB

	P37	P38	P39	P40	JUMLAH
P37	0.57	0.65	0.48	0.43	2.13
P38	0.19	0.22	0.36	0.29	1.05
P39	0.14	0.07	0.12	0.21	0.55
P40	0.10	0.05	0.04	0.07	0.26
JUMLAH	1	1	1	1	4

Dalam menentukan bobot dari sub kriteria PBB, dilakukan perkalian bobot kriteria PBB dengan bobot dari sub kriteria PBB. Hasil dari penghitungan tersebut seperti pada Tabel 3.36. Berikut hasil penghitungan bobot sub kriteria PBB untuk sub kriteria P36.

$$P37 = \frac{2.13}{4} = 0.53$$

$$P37 = 0.53 \times 0.04 = 0.02$$

Tabel 3.36 Bobot Sub Kriteria PBB

BOBOT KRITERIA	0,04	
P37	0.53	0.02
P38	0.26	0.01
P39	0.14	0.01
P40	0.07	0.003

11. Kriteria Jumlah Motor (C11)

Sub kriteria jumlah motor terdiri dari 0 (P41), 1 (P42), 2 (P43), 3(P44) dan >3 (P45). Untuk matriks perbandingan berpasangan jumlah motor dapat dilihat pada Tabel 3.37.



Tabel 3.37 Matriks Perbandingan Kriteria Jumlah Motor

	P41	P42	P43	P44	P45
P41	1.00	3.00	3.00	4.00	5.00
P42	0.33	1.00	2.00	3.00	4.00
P43	0.33	0.50	1.00	1.00	3.00
P44	0.25	0.33	1.00	1.00	3.00
P45	0.20	0.25	0.33	0.33	1.00
JUMLAH	2.12	5.08	7.33	9.33	16.00

Selanjutnya dilakukan normalisasi terhadap matriks perbandingan sub kriteria jumlah motor seperti pada Persamaan 2.2.

$$Norm_{11} = \frac{1}{2.12} = 0.47$$

Hasil dari penghitungan normalisasi matriks perbandingan sub kriteria jumlah motor dapat dilihat pada Tabel 3.38.

Tabel 3.38 Hasil Normalisasi Sub Kriteria Jumlah Motor

	P41	P42	P43	P44	P45	JUMLAH
P41	0.47	0.59	0.41	0.43	0.31	2.21
P42	0.16	0.20	0.27	0.32	0.25	1.20
P43	0.16	0.10	0.14	0.11	0.19	0.69
P44	0.12	0.07	0.14	0.11	0.19	0.61
P45	0.09	0.05	0.05	0.04	0.06	0.29
JUMLAH	1.00	1.00	1.00	1.00	1.00	5.00

Dalam menentukan bobot dari sub kriteria jumlah motor, dilakukan perkalian bobot kriteria jumlah motor dengan bobot dari sub kriteria jumlah motor. Hasil dari penghitungan tersebut seperti pada Tabel 3.28. Berikut hasil penghitungan bobot sub kriteria jumlah motor untuk sub kriteria P41.

$$P41 = \frac{2.21}{5} = 0.442553$$

$$P41 = 0.442553 \times 0.04 = 0.016377$$

Tabel 3.39 Bobot Sub Kriteria Jumlah Motor

BOBOT KRITERIA		0,04
P41	0.442553	0.016377

P42	0.239671	0.008869
P43	0.137369	0.005083
P44	0.122938	0.004549
P45	0.057467	0.002127

12. Kriteria Jumlah Mobil (C12)

Sub kriteria jumlah mobil terdiri dari 0 mobil (P46), 1 mobil (P47) dan >1 mobil (P48). Matrik perbandingan kriteria jumlah mobil dapat dilihat pada Tabel 3.40.

Tabel 3.40 Matrik Perbandingan Sub Kriteria Jumlah Mobil

	P46	P47	P48
P46	1	2	3
P47	0.5	1	2
P48	0.33	0.5	1
JUMLAH	1.83	3.5	6

Selanjutnya dilakukan normalisasi terhadap matriks perbandingan sub kriteria jumlah mobil seperti pada Persamaan 2.2.

$$Norm_{11} = \frac{1}{1.83} = 0.55$$

Hasil dari penghitungan normalisasi matriks perbandingan sub kriteria jumlah mobil dapat dilihat pada Tabel 3.41.

Tabel 3.41 Normalisasi Sub Kriteria Jumlah Mobil

	P46	P47	P48	JUMLAH
P46	0.55	0.57	0.50	1.62
P47	0.27	0.29	0.33	0.89
P48	0.18	0.14	0.17	0.49
JUMLAH	1	1	1	3

Dalam menentukan bobot dari sub kriteria jumlah mobil, dilakukan perkalian bobot kriteria jumlah mobil dengan bobot dari sub kriteria jumlah mobil. Hasil dari penghitungan tersebut seperti pada Tabel 3.42. Berikut hasil penghitungan bobot sub kriteria jumlah motor untuk sub kriteria P46.

$$P_{46} = \frac{1.62}{3} = 0.54$$

$$P_{46} = 0.54 \times 0.03 = 0.02$$



Tabel 3.42 Bobot Sub Kriteria Jumlah Mobil

BOBOT KRITERIA		0,03
P46	0.54	0.02
P47	0.30	0.01
P48	0.16	0.01

13. Kriteria Program Beasiswa (C13)

Sub kriteria program beasiswa terdiri dari Bidik Misi (P49) dan Lainnya (P50). Matriks perbandingan kriteria program beasiswa seperti pada Tabel 3.43.

Tabel 3.43 Matriks Perbandingan Sub Kriteria Program Beasiswa

	P49	P50
P49	1	1
P50	1	1
JUMLAH	2	2

Selanjutnya dilakukan normalisasi terhadap matriks perbandingan sub kriteria program beasiswa seperti pada Persamaan 2.2.

$$Norm_{11} = \frac{1}{2} = 0.50$$

Hasil dari penghitungan normalisasi matriks perbandingan sub kriteria program beasiswa dapat dilihat pada Tabel 3.44.

Tabel 3.44 Hasil Normalisasi Sub Kriteria Program Beasiswa

	P49	P50	JUMLAH
P49	0.50	0.50	1.00
P50	0.50	0.50	1.00
JUMLAH	1	1	2

Dalam menentukan bobot dari sub kriteria program beasiswa, dilakukan perkalian bobot kriteria program beasiswa dengan bobot dari sub kriteria program beasiswa. Hasil dari penghitungan tersebut seperti pada Tabel 3.45. Berikut hasil penghitungan bobot sub kriteria program beasiswa untuk sub kriteria P49.

$$P49 = \frac{1}{2} = 0.50$$

$$P49 = 0.50 \times 0.02 = 0.011$$



Tabel 3.45 Bobot Sub Kriteria Program Beasiswa

BOBOT KRITERIA		0,02
P49	0.500	0.011
P50	0.500	0.011

Setelah penghitungan seluruh sub kriteria dilakukan, selanjutnya dilakukan reduksi data untuk memperoleh kriteria tertentu berdasarkan nilai yang ditetapkan. Nilai *threshold* yang ditetapkan dalam perhitungan ini adalah 0.015. Nilai bobot dari sub kriteria yang lebih dari sama dengan *threshold* akan digunakan untuk proses selanjutnya seperti pada Tabel 3.46. Sedangkan nilai bobot dari sub kriteria yang berada di bawah *threshold* tidak akan digunakan. Tabel 3.47 menunjukkan nilai bobot seluruh kriteria dan sub kriteria.

Tabel 3.46 Nilai Bobot Seluruh Kriteria dan Sub Kriteria

KRITERIA	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
BOBOT	0.095	0.067	0.029	0.016	0.010	0.082	0.037	0.016	0.070	0.032
KRITERIA	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20
BOBOT	0.021	0.010	0.007	0.034	0.029	0.018	0.009	0.005	0.004	0.033
KRITERIA	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30
BOBOT	0.017	0.009	0.005	0.003	0.062	0.018	0.008	0.025	0.008	0.003
KRITERIA	P31	P32	P33	P34	P35	P36	P37	P38	P39	P40
BOBOT	0.032	0.010	0.003	0.030	0.010	0.004	0.022	0.011	0.006	0.003
KRITERIA	P41	P42	P43	P44	P45	P46	P47	P48	P49	P50
BOBOT	0.016	0.009	0.005	0.005	0.002	0.017	0.009	0.005	0.011	0.011

Tabel 3.47 Kriteria Setelah Reduksi

KRITERIA	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
BOBOT	0.095	0.067	0.029	0.016	NO	0.082	0.037	0.016	0.070	0.032
KRITERIA	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20
BOBOT	0.021	NO	NO	0.034	0.029	0.018	NO	NO	NO	0.033
KRITERIA	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30
BOBOT	0.017	NO	NO	NO	0.062	0.018	NO	0.025	NO	NO
KRITERIA	P31	P32	P33	P34	P35	P36	P37	P38	P39	P40
BOBOT	0.032	NO	NO	0.030	NO	NO	0.022	NO	NO	NO
KRITERIA	P41	P42	P43	P44	P45	P46	P47	P48	P49	P50
BOBOT	0.016	NO	NO	NO	NO	0.017	NO	NO	NO	NO

3.3.2.2 Perhitungan Manual Proses Normalisasi

Pada Tabel 3.2 merupakan data yang akan dilakukan Normalisasi, yang terdiri dari 12 data sampel. Data sampel terdiri dari 9 data *training* dan 3 data *testing*. Hasil dari normalisasi pada data *training* dan data *testing* dapat dilihat pada Tabel 3.48 dan 3.49. Berikut contoh penghitungan normalisasi data *training* untuk $p_{(1,1)}$ sesuai dengan Persamaan 2.7. Pada proses normalisasi akan digunakan *interval* [0.1,0.9] yang bertujuan untuk menghindari hasil bernilai nol, sehingga hasil minimum bernilai 0.1 dan maksimum bernilai 0.9. Hasil dari normalisasi data *training* dan data *testing* secara keseluruhan pada level 1 dapat dilihat pada Lampiran L.

$$p'_{11} = \left(\frac{0 - 0}{5 - 0} \right) \times (0.9 - 0.1) + 0.1 = 0.1$$

Tabel 3.48 Hasil Normalisasi Data Training

NP	P1	P2	P3	P4	P6	P34	P37	P41	P46	CLASS
211	0.1	0.7	0.1	0.1	0.1	0.1	0.9	0.1	0.9	A
174	0.1	0.7	0.1	0.1	0.1	0.1	0.9	0.9	0.9	A
187	0.1	0.7	0.1	0.1	0.1	0.1	0.9	0.1	0.9	A
86	0.1	0.1	0.6	0.1	0.1	0.1	0.1	0.1	0.9	B
256	0.1	0.1	0.6	0.1	0.1	0.1	0.1	0.1	0.9	B
166	0.1	0.1	0.6	0.1	0.1	0.1	0.1	0.1	0.9	B
233	0.1	0.1	0.1	0.1	0.1	0.1	0.9	0.1	0.9	C
265	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.9	C
183	0.1	0.7	0.1	0.1	0.1	0.1	0.1	0.1	0.9	C

Tabel 3.49 Hasil Normalisasi Data Testing

NP	P1	P2	P3	P4	P6	P34	P37	P41	P46	CLASS
135	0.1	0.7	0.1	0.1	0.1	0.1	0.9	0.1	0.9	A
226	0.1	0.1	0.58	0.1	0.1	0.1	0.1	0.1	0.9	B
185	0.1	0.1	0.58	0.1	0.1	0.1	0.1	0.1	0.9	C

Setelah dilakukan normalisasi maka dilakukan perkalian terhadap bobot yang didapatkan dari hasil perhitungan AHP. Hasil dari perkalian normalisasi data *training* dan normalisasi data *testing* terhadap bobot kriteria yang telah dihitung dengan metode AHP dapat dilihat pada Tabel 3.50 dan 3.51. Berikut contoh penghitungan hasil normalisasi data *training* dikali bobot sub kriteria untuk data nomor peserta 211 pada sub kriteria P1 :

$$X_{1,1} = 0.1 \times 0.095 = 0.01$$

Dimana 0.1 merupakan hasil normalisasi data untuk nomor peserta 211 pada sub kriteria P1, sedangkan 0,095 merupakan hasil perhitungan bobot sub kriteria P1. Hasil perhitungan normalisasi data *training* yang telah dikalikan dengan sub kriteria secara lengkap dapat dilihat pada Lampiran M.

Tabel 3.50 Perkalian Bobot Sub Kriteria dengan Hasil Normalisasi Data Training

NP	P1	P2	P3	P4	P6	P34	P37	P41	P46	KELAS
211	0.01	0.05	0.003	0.002	0.008	0.003	0.020	0.002	0.015	A
174	0.01	0.05	0.003	0.002	0.008	0.003	0.020	0.015	0.015	A
187	0.01	0.05	0.003	0.002	0.008	0.003	0.020	0.002	0.015	A
86	0.01	0.007	0.017	0.002	0.008	0.003	0.002	0.002	0.015	B
256	0.01	0.007	0.017	0.002	0.008	0.003	0.002	0.002	0.015	B
166	0.01	0.007	0.017	0.002	0.008	0.003	0.002	0.002	0.015	B
233	0.01	0.007	0.003	0.002	0.008	0.003	0.020	0.002	0.015	C
265	0.01	0.007	0.003	0.002	0.008	0.003	0.002	0.002	0.015	C
183	0.01	0.05	0.003	0.002	0.008	0.003	0.002	0.002	0.015	C

Untuk normalisasi data *testing* yang telah dikalikan dengan bobot sub kriteria. Berikut contoh pengitungan untuk normalisasi data *testing* dikalikan dengan bobot sub kriteria terpilih untuk nomor peserta 135 dengan sub kriteria P1.

$$X_{1,1} = 0.1 \times 0.095$$

$$= 0.01.$$

Dimana 0.1 adalah hasil normalisasi untuk data *testing* 135 pada sub kriteria P1, sedangkan 0.095 merupakan hasil perhitungan bobot AHP untuk sub kriteria P1. Hasil perhitungan normalisasi data *testing* yang telah dikalikan dengan bobot sub kriteria terpilih secara lengkap dapat dilihat pada Lampiran M.

Tabel 3.51 Perkalian Bobot Sub Kriteria dengan Normalisasi Data Testing

NP	P1	P2	P3	P4	P6	P34	P37	P41	P46	KELAS
135	0.1	0.74	0.1	0.1	0.1	0.1	0.9	0.1	0.9	A
226	0.1	0.1	0.58	0.1	0.1	0.1	0.1	0.1	0.9	B
185	0.1	0.1	0.58	0.1	0.1	0.1	0.1	0.1	0.9	C

3.3.2.3 Perhitungan SVM

Pada proses *training* akan digunakan metode *Sequential Training*. *Hyperplane* dibutuhkan sebagai fungsi pemisah dengan nilai margin maksimal. Untuk mendapatkan margin maksimal inilah dirumuskan dengan metode

Sequential Training. Pada proses *training* dilakukan inisialisasi nilai d (*degree*), $\alpha_i, \lambda, \gamma, C$.

Pada penghitungan *kernel* menggunakan *kernel Polynomial Degreee d*, dengan menentukan nilai $d=2$ dan melakukan perhitungan seperti pada Persamaan 2.22. Berikut merupakan contoh perhitungan dengan *kernel Polynomial Degree d* untuk baris pertama kolom pertama pada Tabel 3.39.

Pada Tabel 3.52 merupakan hasil perhitungan data *training* dengan *Kernel Polynomial*. Untuk hasil lengkap penghitungan *kernel Polynomial Degree d* level 1 dapat dilihat pada Lampiran N. Berikut merupakan contoh penghitungan *kernel Polynomial Degree d* :

$$\begin{aligned}
 K_{(1,1)} = & [(0.01 \times 0.01) + (0.05 \times 0.05) + (0.003 \times 0.003) \\
 & + (0.002 \times 0.002) + (0.008 \times 0.008) + (0.004 \times 0.004) \\
 & + (0.008 \times 0.008) + (0.063 \times 0.063) + (0.003 \times 0.003) \\
 & + (0.002 \times 0.002) + (0.003 \times 0.003) + (0.003 \times 0.003) \\
 & + (0.016 \times 0.016) + (0.003 \times 0.003) + (0.015 \times 0.015) \\
 & + (0.006 \times 0.006) + (0.011 \times 0.011) + (0.023 \times 0.023) \\
 & + (0.003 \times 0.003) + (0.003 \times 0.003) + (0.020 \times 0.020) \\
 & + (0.002 \times 0.002) + (0.015 \times 0.015)]^2 = 0.000073
 \end{aligned}$$

Tabel 3.52 Hasil Perhitungan Kernel Polynomial Degree d

NP	211	174	233	265	183
211	0.000073	0.000022	0.000005	0.000006	0.000016
174	0.000022	0.000034	0.000006	0.000003	0.000014
187	0.000065	0.000021	0.000005	0.000004	0.000016
86	0.000005	0.000005	0.000003	0.000002	0.000002
256	0.000004	0.000004	0.000002	0.000002	0.000002
166	0.000004	0.000004	0.000002	0.000002	0.000002
233	0.000005	0.000006	0.000004	0.000002	0.000002
265	0.000006	0.000003	0.000002	0.000004	0.000002
183	0.000016	0.000014	0.000002	0.000002	0.000013

Setelah seluruh penghitungan *kernel Polynomial Degree d* dilakukan, maka selanjutnya dilakukan penghitungan matriks *Hessian*. Hasil dari penghitungan matriks *Hessian* dapat dilihat pada Tabel 3.53. Berikut contoh penghitungan matriks *Hessian* untuk $D_{1,1}$ seperti pada Persamaan 2.25 :

Untuk baris ke-1 kolom ke-1 :

$$D_{1,1} = 1 \times 1 \times (0.000073 + 0.5^2) = 0.25007295$$

Untuk baris ke-4 kolom ke-1 :

$$D_{4,1} = (-1) \times 1 \times (0.000005 + 0.5^2) = -0.25000468$$

Dimana pada baris ke-1 kolom ke-1, nilai 1 adalah nilai *class* pada atribut pertama, 0.000073 merupakan hasil perhitungan kernel *Polynomial Degree d* dan 0,5 merupakan inisialisasi nilai *lamda*.

Tabel 3.53 Hasil Perhitungan Matriks Hessian

NP	211	174	187	86	256	166	233	265	183
211	0.25007	0.25002	0.25006	-0.25000	-0.25000	-0.25000	-0.25000	-0.25000	-0.25001
174	0.25002	0.25003	0.25002	-0.25000	-0.25000	-0.25000	-0.25000	-0.25000	-0.25001
187	0.25006	0.25002	0.25006	-0.25000	-0.25000	-0.25000	-0.25000	-0.25000	-0.25001
86	-0.25000	-0.25000	-0.25000	0.25000	0.25000	0.25000	0.25000	0.25000	0.25000
256	-0.25000	-0.25000	-0.25000	0.25000	0.25000	0.25000	0.25000	0.25000	0.25000
166	-0.25000	-0.25000	-0.25000	0.25000	0.25000	0.25000	0.25000	0.25000	0.25000
233	-0.25000	-0.25000	-0.25000	0.25000	0.25000	0.25000	0.25000	0.25000	0.25000
265	-0.25000	-0.25000	-0.25000	0.25000	0.25000	0.25000	0.25000	0.25000	0.25000
183	-0.25001	-0.25001	-0.25001	0.25000	0.25000	0.25000	0.25000	0.25000	0.25001

Selanjutnya dilakukan iterasi untuk setiap $i = 1, 2, \dots, n$. Penghitungan ini dilakukan iterasi (itermax)=2, C=1 dan CLR adalah nilai konstanta. Nilai CLR =0.01. Untuk mendapatkan nilai γ seperti pada Persamaan 2.29 dengan nilai konstanta *learning rate* (CLR) dibagi dengan nilai maksimum dari diagonal matriks *hessian*. Berikut contoh penghitungan nilai γ (*gamma*) :

$$\gamma = \frac{0,01}{0,25007} = 0.039988331$$

Hasil penghitungan pada iterasi ke-1 level 1 dapat dilihat pada Tabel 3.54. pada iterasi ke 0 dilakukan perhitungan E_i seperti pada Persamaan 2.26. Berikut contoh penghitungan nilai E_i :

$$E_1 = ((0 \times 0.25007) + (0 \times 0.25002) + (0 \times 0.25006) + (0 \times -0.25000) + (0 \times -0.25000) + (0 \times -0.25000) + (0 \times -0.25000) + (0 \times -0.25001)) = 0$$

Tabel 3.54 Hasil Perhitungan E_i 1 Level 1

NP	211	174	187	86	256	166	233	265	183	E_i
214	0	0	0	0	0	0	0	0	0	0
218	0	0	0	0	0	0	0	0	0	0
246	0	0	0	0	0	0	0	0	0	0
139	0	0	0	0	0	0	0	0	0	0
200	0	0	0	0	0	0	0	0	0	0



226	0	0	0	0	0	0	0	0	0	0
65	0	0	0	0	0	0	0	0	0	0
400	0	0	0	0	0	0	0	0	0	0
401	0	0	0	0	0	0	0	0	0	0

Selanjutnya dilakukan perhitungan nilai $\delta\alpha_i$ seperti pada Persamaan 2.27. Hasil perhitungan nilai $\delta\alpha_i$ dapat dilihat pada Tabel 3.55. Berikut contoh penghitungan nilai $\delta\alpha_i$:

$$\delta\alpha_1 = \min\{\max[0.03998831(1 - 0), -0], 1 - 0\}$$

$$\delta\alpha_1 = \min\{0.03998831, 1\}$$

$$\delta\alpha_1 = 0.03998831$$

Tabel 3.55 Hasil Perhitungan Nilai Delta Alpha

	211	174	187	86	256	166	233	265	183
$\delta\alpha$	0,0399 88331	0,0399 88331	0,0399 88331	0,0399 88331	0,0399 88331	0,0399 88331	0,0399 88331	0,0399 88331	0,0399 88331

Setelah didapat nilai $\delta\alpha_i$ maka langkah selanjutnya yakni memperbaharui nilai α_i seperti pada Persamaan 2.28. Berikut contoh perhitungan nilai α_1 adalah $\alpha_1 = 0 + 0.3998831 = 0.3998831$. Hasil dari perhitungan nilai α_i secara keseluruhan dapat dilihat pada Tabel 3.56.

Tabel 3.56 Hasil Perhitungan Nilai Alpha

	211	174	187	86	256	166	233	265	183
α_i	0,0399 88331	0,0399 88331	0,0399 88331	0,0399 88331	0,0399 88331	0,0399 88331	0,0399 88331	0,0399 88331	0,0399 88331

Setelah nilai α_i diperbaharui, selanjutnya melakukan pengecekan iterasi. Apabila nilai iterasi telah mencapai nilai konvergen dimana $\max(|\delta\alpha_i| < \varepsilon)$, maka iterasi akan dihentikan. Selain pada kondisi tersebut iterasi akan dihentikan jika telah mencapai maksimum iterasi yang ditentukan. Nilai α_i baru pada iterasi sebelumnya akan digunakan untuk perhitungan pada proses iterasi selanjutnya. Diketahui nilai α_i pada iterasi ke-1 adalah 0,039988331. Hasil dari perhitungan iterasi 2 level 1 dapat dilihat pada Tabel 3.57

Tabel 3.57 Hasil Perhitungan E_i 2 Level 1

NP	211	174	187	86	256	166	233	265	183	E_i
211	0.01	0.01	0.01	-0.01	-0.01	-0.01	-0.01	-0.01	-0.01	-0.02999



174	0.01	0.01	0.01	-0.01	-0.01	-0.01	-0.01	-0.01	-0.01	-0.02999
187	0.01	0.01	0.01	-0.01	-0.01	-0.01	-0.01	-0.01	-0.01	-0.02999
86	-0.01	-0.01	-0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.02999
256	-0.01	-0.01	-0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.02999
166	-0.01	-0.01	-0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.02999
233	-0.01	-0.01	-0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.02999
265	-0.01	-0.01	-0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.02999
183	-0.01	-0.01	-0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.02999

Selanjutnya dilakukan penghitungan nilai $\delta\alpha_i$ seperti pada Persamaan 2.27. Hasil perhitungan nilai $\delta\alpha_i$ dapat dilihat pada Tabel 3.58. Berikut contoh perhitungan nilai $\delta\alpha_i$:

$$\delta\alpha_1 = \min\{\max[0.039988331(1-(-0.02999)), -0.039988331], 1-0.039988331\}$$

$$\delta\alpha_1 = 0.041187439$$

Tabel 3.58 Hasil Perhitungan Nilai Delta Alpha Iterasi 2

	211	174	187	86	256	166	233	265	183
$\delta\alpha_i$	0.041187439	0.041187566	0.041187448	0.038789023	0.038789024	0.038789024	0.038789032	0.038789033	0.038789069

Setelah didapat nilai $\delta\alpha_i$ maka langkah selanjutnya yakni memperbaharui nilai α_i seperti pada Persamaan 2.28. Berikut contoh perhitungan nilai α_1 adalah $\alpha_1 = 0.039988331 + 0.41187439 = 0.08117576974$. Hasil dari perhitungan nilai α_i secara keseluruhan dapat dilihat pada Tabel 3.59.

Tabel 3.59 Hasil Perhitungan Nilai Alpha Iterasi 2

	211	174	187	86	256	166	233	265	183
α_i	0.08117576974	0.08117589668	0.08117577850	0.07877735407	0.07877735451	0.07877735451	0.07877736332	0.07877736380	0.07877739960

Hasil dari perhitungan nilai alpha pada Tabel 3.46 akan digunakan untuk menentukan *support vector* pada perhitungan selanjutnya. *Support Vector* adalah perhitungan yang ditentukan dari nilai α_j pada iterasi terakhir (dalam perhitungan ini yakni pada iterasi 2). Nilai α_j dari data *training* dianggap sebagai *support vector* apabila nilainya lebih dari nol. Pada Tabel 3.46 terlihat bahwa nilai α_j yang dihasilkan lebih dari nol sehingga semuanya merupakan *support vector*.

Selanjutnya dihitung nilai bobot data *training*. Dalam penghitungan bobot data *training* terdapat 2 bagian, pertama bobot data training di *dot-product*

dengan menggunakan nilai data *training* yang memiliki nilai α tertinggi kelas positif ($w.x^+$), kedua bobot data *training* di *dot-product* dengan nilai data *training* yang memiliki nilai α tertinggi pada kelas negatif ($w.x^-$). Perhitungan pada setiap bobot menggunakan perhitungan kernel dari data *training* yang memiliki kelas negatif maupun positif untuk masing-masing bobot positif dan negatif. Diketahui nilai x^+ adalah nilai dari salah satu *support vector* dari *class positive* pada nomor peserta (NP) 174 dan x^- adalah nilai dari salah satu *support vector* dari *class negative*. pada nomor peserta 183. Data awal setelah dilakukan normalisasi yang selanjutnya akan dikalikan dengan data yang memiliki nilai *support vector* positif yakni nomor peserta 174 dan data yang memiliki nilai *support vector* negatif yakni nomor peserta 183. Adapun perhitungan untuk menentukan kelas positif dan negatif dilakukan pada data training NP 211 dengan nilai $K(x^+, x_i)$ sebagai berikut :

$$\begin{aligned}
 K(x_1, x^+) = & ((0.01 \times 0.01) + (0.05 \times 0.05) + (0.003 \times 0.003) \\
 & + (0.002 \times 0.002) + (0.008 \times 0.008) + (0.004 \times 0.004) \\
 & + (0.008 \times 0.008) + (0.063 \times 0.063) + (0.003 \times 0.003) \\
 & + (0.002 \times 0.002) + (0.003 \times 0.003) + (0.003 \times 0.003) \\
 & + (0.016 \times 0.016) + (0.003 \times 0.003) + (0.015 \times 0.015) \\
 & + (0.006 \times 0.006) + (0.011 \times 0.011) + (0.023 \times 0.023) \\
 & + (0.003 \times 0.003) + (0.003 \times 0.003) + (0.020 \times 0.020) \\
 & + (0.002 \times 0.002) + (0.015 \times 0.015))^2 = 0.000022
 \end{aligned}$$

Berikut contoh perhitungan untuk $K(x^-, x_i)$:

$$\begin{aligned}
 K(x_1, x^-) = & ((0.01 \times 0.01) + (0.05 \times 0.05) + (0.003 \times 0.003) \\
 & + (0.002 \times 0.002) + (0.008 \times 0.008) + (0.004 \times 0.004) \\
 & + (0.008 \times 0.008) + (0.063 \times 0.063) + (0.003 \times 0.003) \\
 & + (0.002 \times 0.002) + (0.003 \times 0.003) + (0.003 \times 0.003) \\
 & + (0.016 \times 0.016) + (0.003 \times 0.003) + (0.015 \times 0.015) \\
 & + (0.006 \times 0.006) + (0.011 \times 0.011) + (0.023 \times 0.023) \\
 & + (0.003 \times 0.003) + (0.003 \times 0.003) + (0.020 \times 0.002) \\
 & + (0.002 \times 0.002) + (0.015 \times 0.015))^2 = 0.000017
 \end{aligned}$$

Hasil perhitungan dari $K(x_i, x^+)$ dan $K(x_i, x^-)$ secara keseluruhan dapat dilihat pada Tabel 3.60.

Tabel 3.60 Mencari Nilai Kernel x^+ dan x^-

NP	$K(x_i, x^+)$	$K(x_i, x^-)$
211	0.000022	0.000017
174	0.000034	0.000015
187	0.000021	0.000017
86	0.000005	0.000003
256	0.000004	0.000002
166	0.000004	0.000002
233	0.000006	0.000002

265	0.000003	0.000003
183	0.000015	0.000014

Selanjutnya dihitung nilai bobotnya untuk mencari nilai $\alpha_i y_i K(x_i, x)$, jika diketahui nilai $\alpha_1 = 0.08117577139$ dan nilai $y_1 = 1$, maka hasil perhitungan untuk bobot $w.x^+$ dan $w.x^-$ dapat dilihat pada Tabel 3.61. Berikut contoh penghitungan untuk $w.x^+$ dan $w.x^-$ untuk nomor peserta 211 :

$$w.x^+ = \left(\sum_{i=1}^n \alpha_i y_i x_i \right) \cdot x^+ = \sum_{i=1}^n \alpha_i y_i K(x_i, x^+)$$

$$\begin{aligned} \sum w.x^+ &= ((0.081117576974 \times 1 \times 0.000022) \\ &+ (0.08117589668 \times 1 \times 0.000034) \\ &+ (0.08117577850 \times 1 \times 0.000021) \\ &+ (0.07877735407 \times -1 \times 0.000005) \\ &+ (0.07877735451 \times -1 \times 0.000004) \\ &+ (0.07877735451 \times -1 \times 0.000004) \\ &+ (0.07877736332 \times -1 \times 0.000006) \\ &+ (0.07877736380 \times -1 \times 0.000003) \\ &+ (0.07877739960 \times -1 \times 0.000015)) = 0.000003316 \end{aligned}$$

$$\begin{aligned} \sum w.x^- &= ((0.081117576974 \times 1 \times 0.000017) \\ &+ (0.08117589668 \times 1 \times 0.000015) \\ &+ (0.08117577850 \times 1 \times 0.000017) \\ &+ (0.07877735407 \times -1 \times 0.000003) \\ &+ (0.07877735451 \times -1 \times 0.000002) \\ &+ (0.07877735451 \times -1 \times 0.000002) \\ &+ (0.07877736332 \times -1 \times 0.000002) \\ &+ (0.07877736380 \times -1 \times 0.000003) \\ &+ (0.07877739960 \times -1 \times 0.000014)) = 0.000001999 \end{aligned}$$

Tabel 3.61 Nilai Bobot Kelas Positif dan Negatif Data Training

NP	$w.x^+$	$w.x^-$
211	0.000001769	0.000001413
174	0.000002795	0.000001215
187	0.000001731	0.000001379
86	-0.000000413	-0.000000222
256	-0.000000334	-0.000000165
166	-0.000000334	-0.000000165
233	-0.000000455	-0.000000172
265	-0.000000262	-0.000000212



183	-0.000001179	-0.000001070
Σ	0.000003316	0.000001999

Selanjutnya melakukan penghitungan nilai bias yang merupakan posisi bidang relatif terhadap pusat koordinat, dimana membutuhkan jumlah dari kelas bobot positif dan jumlah dari kelas bobot negatif. Nilai bias yang didapatkan dengan melakukan penjumlahan bobot kelas positif dan jumlah dari bobot kelas negatif kemudian dikalikan dengan $-\frac{1}{2}$ seperti pada Persamaan 2.14.

Berikut contoh penghitungan nilai bias dimana b adalah bias.

$$b = -\frac{1}{2}(0.000003316 + 0.000001999) = -0.000002657$$

Selanjutnya menghitung nilai Fungsi $f(x)$ yang dilakukan kepada data *testing* dengan menjumlahkan nilai dari total bobot data *testing* dengan nilai bias yang sudah dihitung pada proses sebelumnya yang menghasilkan tanda dari nilai yang dihasilkan oleh fungsi $f(x)$. Beberapa hal yang harus dipenuhi adalah jika level satu bernilai positif maka akan masuk kelas +1 dan masuk kelas A (sangat layak), apabila bernilai negatif maka akan masuk kelas -1. Apabila level satu bernilai negatif akan dihitung lagi pada level selanjutnya, jika pada level dua bernilai positif maka akan masuk kelas +1 dan masuk kelas B (layak), jika pada level dua bernilai negatif maka akan masuk kelas -1 dan masuk dalam kelas C (tidak layak). Data uji yang digunakan untuk menghitung fungsi $f(x)$ seperti pada Tabel 3.51.

Hasil perhitungan proses *testing* SVM terhadap masing-masing data *training* seperti pada tabel 3.51. Berikut contoh perhitungan kernel untuk data uji ke 1 :

$$K(x_1, x) = ((0.01 \times 0.009) + (0.05 \times 0.049) + (0.003 \times 0.003) + (0.002 \times 0.002) + (0.008 \times 0.008) + (0.004 \times 0.004) + (0.008 \times 0.008) + (0.0063 \times 0.0063) + (0.003 \times 0.003) + (0.002 \times 0.002) + (0.003 \times 0.003) + (0.003 \times 0.003) + (0.016 \times 0.016) + (0.003 \times 0.003) + (0.015 \times 0.015) + (0.006 \times 0.006) + (0.011 \times 0.011) + (0.023 \times 0.003) + (0.003 \times 0.003) + (0.003 \times 0.003) + (0.020 \times 0.020) + (0.002 \times 0.002) + (0.015 \times 0.015))^2 = 0.000065444.$$

Selanjutnya dilakukan penghitungan untuk $\alpha_i y_i K(x_i, x)$ untuk mendapatkan jumlah bobot pada masing-masing data *training*. Berikut contoh perhitungan untuk bobot masing-masing data *training* :

$$\alpha_1 y_1 K(x_1, x) = 0.08117577139 \times 1 \times 0.000065444 = 0.000005313$$

Dilakukan perhitungan $\alpha_i y_i K(x_i, x)$ sebanyak seluruh data *training*. Seluruh hasil perhitungan bobot pada data uji 1 untuk $\alpha_i y_i K(x_i, x)$ akan dijumlahkan.

Tabel 3.62 Hasil Perhitungan Data Uji 1 Level 1

NP	$K(x_i, x)$	$\alpha_i y_i K(x_i, x)$
211	0.000065444	0.000005313
174	0.000021322	0.000001731
187	0.000064635	0.000005247
86	0.000004469	-0.000000352
256	0.000003551	-0.000000280
166	0.000003551	-0.000000280
233	0.000004964	-0.000000391
265	0.000004315	-0.000000340
183	0.000016986	-0.000001338
	Σ	0.000009310

Setelah dilakukan perhitungan kernel dan bobot untuk data uji 1, maka dilanjutkan untuk perhitungan kernel dan bobot pada data uji 2 seperti pada Tabel 3.63 berikut :

Tabel 3.63 Hasil Perhitungan Data Uji 2 Level 1

NP	$K(x_i, x)$	$\alpha_i y_i K(x_i, x)$
211	0.000003742	0.000000304
174	0.000004244	0.000000345
187	0.000003551	0.000000288
86	0.000003592	-0.000000283
256	0.000003496	-0.000000275
166	0.000003496	-0.000000275
233	0.000002249	-0.000000177
265	0.000001523	-0.000000120
183	0.000002099	-0.000000165
	Σ	-0.000000360

Selanjutnya, dilaukan perhitungan kernel dan bobot untuk data uji ke 3 seperti pada Tabel 3.64.

Tabel 3.64 Hasil Perhitungan Data Uji 3 Level 1

NP	$K(x_i, x)$	$\alpha_i y_i K(x_i, x)$
----	-------------	--------------------------

211	0.000003016	0.000000245
174	0.000003469	0.000000282
187	0.000002845	0.000000231
86	0.000002882	-0.000000227
256	0.000002796	-0.000000220
166	0.000002796	-0.000000220
233	0.000002184	-0.000000172
265	0.000001074	-0.000000085
183	0.000001566	-0.000000123
Σ		-0.000000290

Apabila sudah mendapatkan hasilnya, maka selanjutnya dilakukan penghitungan untuk mendapatkan nilai $f(x)$ seperti pada Persamaan 2.20. Hasil dari perhitungan nilai $f(x)$ dapat dilihat pada Tabel 3.65. Berikut contoh perhitungan untuk mendapatkan nilai $f(x)$ pada data uji 1 level 1.

$$f(x) = \text{sign}(0.000009310 + (-0.000002657)) = 0.00000665$$

$$f(x) = 1$$

Tabel 3.65 Hasil Perhitungan Nilai $f(x)$

Nilai $f(x)$ data uji 1 level 1	0.00000665
Fungsi Klasifikasi	1
Nilai $f(x)$ data uji 1 level 1	-0.00000266
Fungsi Klasifikasi	-1
Nilai $f(x)$ data uji 1 level 1	-0.00000295
Fungsi Klasifikasi	-1

Selanjutnya menentukan matriks penilaian alternatif untuk level 2 yang didapatkan dari normalisasi data *training* dan data *testing* dari perhitungan pada level 1 sebelumnya. Pada matriks penilaian alternatif untuk level 2, kelas yang digunakan untuk data *training* adalah kelas B untuk kategori layak dan kelas C untuk kategori tidak layak. Sedangkan untuk data *testing* menggunakan data dimana pada hasil perhitungan $f(x)$ level 1 bernilai -1. Data *training* dan data *testing* yang terpilih untuk dilanjutkan perhitungan ke level 2 seperti yang ditunjukkan pada Tabel 3.66 dan 3.67 yang secara keseluruhan dapat dilihat pada Lampiran O.

Tabel 3.66 Normalisasi Data Training Level 2

NP	P1	P2	P3	P4	P6	P34	P37	P41	P46	KELAS
86	0.1	0.1	0.6	0.1	0.1	0.1	0.1	0.1	0.9	B

256	0.1	0.1	0.6	0.1	0.1	0.1	0.1	0.1	0.9	B
166	0.1	0.1	0.6	0.1	0.1	0.1	0.1	0.1	0.9	B
233	0.1	0.1	0.1	0.1	0.1	0.1	0.9	0.1	0.9	C
265	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.9	C
183	0.1	0.7	0.1	0.1	0.1	0.1	0.1	0.1	0.9	C

Pada Tabel 3.66 merupakan hasil normalisasi data *training* untuk level 2. Karena pada hasil klasifikasi dari setiap nomor peserta yang bernilai -1 pada level 1 adalah nomor peserta 226 dan 185, maka untuk data *testing* yang dilanjutkan perhitungannya pada level 2 adalah data *testing* dengan nomor peserta 226 dan 185. Hasil normalisasi data *testing* untuk level 2 seperti yang ditunjukkan pada Tabel 3.67.

Tabel 3.67 Normalisasi Data Testing Level 2

NP	P1	P2	P3	P4	P6	P34	P37	P41	P46	KELAS
226	0.1	0.1	0.58	0.1	0.1	0.1	0.1	0.1	0.9	B
185	0.1	0.1	0.58	0.1	0.1	0.1	0.1	0.1	0.9	C

Selanjutnya dilakukan terhadap perhitungan perkalian sub bobot kriteria terpilih dengan hasil normalisasi data *training* dan data *testing* terpilih untuk dilanjutkan perhitungan pada level 2 SVM. Data hasil perkalian sub bobot kriteria dengan data *training* seperti pada Tabel 3.68. Secara keseluruhan, hasil perkalian normalisasi data *training* dengan sub bobot kriteria terpilih pada level 2 dapat dilihat pada Lampiran P.

Tabel 3.68 Hasil Perkalian Normalisasi Data Training dengan Sub Bobot Kriteria Terpilih Level 2

NP	P1	P2	P3	P4	P6	P34	P37	P41	P46	KELAS
86	0.009	0.007	0.017	0.002	0.008	0.003	0.002	0.002	0.015	B
256	0.009	0.007	0.017	0.002	0.008	0.003	0.002	0.002	0.015	B
166	0.009	0.007	0.017	0.002	0.008	0.003	0.002	0.002	0.015	B
233	0.009	0.007	0.003	0.002	0.008	0.003	0.020	0.002	0.015	C
265	0.009	0.007	0.003	0.002	0.008	0.003	0.002	0.002	0.015	C
183	0.009	0.049	0.003	0.002	0.008	0.003	0.002	0.002	0.015	C

Sedangkan untuk hasil perkalian normalisasi data *testing* dengan sub bobot kriteria terpilih pada level 2 dapat dilihat pada Tabel 3.69.



Tabel 3.69 Hasil Perkalian Nomalisasi Data *Testing* dengan Sub Bobot Kriteria Terpilih Level 2

NP	P1	P2	P3	P4	P6	P34	P37	P41	P46	CLASS
226	0.009	0.007	0.017	0.002	0.008	0.003	0.002	0.002	0.015	B
185	0.009	0.007	0.017	0.002	0.008		0.003	0.002	0.002	0.015	C

Selanjutnya dilakukan penghitungan matriks *kernel* level 2. Pada proses penghitungan matriks *kernel* level 2 ini menggunakan *kernel Polynomial Degree d*, seperti pada level 1 menggunakan Persamaan 2.22. Berikut contoh perhitungan *Kernel Polynomial* pada Level 2 untuk $x=86$ $y=256$:

$$\begin{aligned}
 [x.y]^d = & [(0.009 \times 0.009) + (0.007 \times 0.007) + (0.017 \times 0.017) \\
 & + (0.002 \times 0.002) + (0.008 \times 0.008) + (0.004 \times 0.004) \\
 & + (0.008 \times 0.008) + (0.007 \times 0.007) + (0.024 \times 0.024) \\
 & + (0.002 \times 0.002) + (0.003 \times 0.003) + (0.003 \times 0.003) \\
 & + (0.016 \times 0.002) + (0.003 \times 0.003) + (0.015 \times 0.015) \\
 & + (0.006 \times 0.006) + (0.011 \times 0.011) + (0.003 \times 0.003) \\
 & + (0.003 \times 0.003) + (0.003 \times 0.003) + (0.002 \times 0.002) \\
 & + (0.002 \times 0.002) + (0.015 \times 0.015)]^2 = 0.000004
 \end{aligned}$$

Hasil dari perhitungan *Kernel Polynomial* dapat dilihat pada Tabel 3.70 berikut:

Tabel 3.70 Hasil Perhitungan *Kernel Polynomial Degree* Level 2

NP	86	256	166	233	265	183
86	0.000005	0.000004	0.000004	0.000003	0.000002	0.000003
256	0.000004	0.000003	0.000003	0.000002	0.000002	0.000002
166	0.000004	0.000003	0.000003	0.000002	0.000002	0.000002
233	0.000003	0.000002	0.000002	0.000004	0.000002	0.000002
265	0.000002	0.000002	0.000002	0.000002	0.000004	0.000003
183	0.000003	0.000002	0.000002	0.000002	0.000003	0.000014

Selanjutnya melakukan penghitungan matriks Hessian level 2 seperti pada Persamaan 2.25. Pada perhitungan matriks *hessian* ini ada beberapa nilai parameter yang diinputkan seperti parameter $\lambda = 0.5$ (variabel skalar), variabel y_i yang merupakan kelas data ke- i , y_j yang merupakan kelas data ke- j . Berikut contoh perhitungan matriks *hessian* untuk baris ke-1 kolom ke-1 :

$$D_{ij} = y_i y_j (K(x_i, x_j) + \lambda^2) = 1 \times 1 \times (0.000005 + 0.5^2) = 0.250004516$$

Hasil dari perhitungan matriks *Hessian* level 2 dapat dilihat pada Tabel 3.71 berikut :

Tabel 3.71 Hasil Perhitungan Matriks *Hessian* Level 2

NP	86	256	166	233	265	183
86	0.250004516	0.250003592	0.250003592	-0.25000299	-0.25000214	-0.25000282
256	0.250003592	0.250003496	0.250003496	-0.25000225	-0.25000152	-0.25000210
166	0.250003592	0.250003496	0.250003496	-0.25000225	-0.25000152	-0.25000210
233	-0.25000299	-0.25000225	-0.25000225	0.250004077	0.250001599	0.250002188
265	-0.25000214	-0.25000152	-0.25000152	0.250001599	0.250003718	0.250002696
183	-0.25000282	-0.25000210	-0.25000210	0.250002188	0.250002696	0.250013587

Selanjutnya melakukan iterasi level 2. Iterasi dilakukan sebanyak $i = 1, 2, \dots, n$. Nilai n berlanjut sampai dengan iterasi maksimum yang telah ditentukan atau iterasi akan berhenti jika nilai dari $\max(|\delta\alpha_i| < \epsilon)$. Pada perhitungan ini dilakukan inisialisasi untuk nilai ϵ (*epsilon*)=0.00001, C (*Complexity*)=1, α (*alpha*)=0, konstanta *Learning Rate* =0.01 dan iterasi maksimum=2. Proses perhitungan E_i diawali dengan menjumlahkan hasil perkalian matriks *hessian* dengan *alpha* ke- i . Proses ini dilakukan sebanyak iterasi yang ditentukan atau jika telah memenuhi kondisi konvergen dimana nilai dari $\max(|\delta\alpha_i| < \epsilon)$. Berikut contoh perhitungan γ (*gamma*) level 2 :

$$\gamma = \frac{0.01}{0.25001359} = 0.03999783$$

Hasil dari perhitungan nilai E_i untuk iterasi 1 level 2 dapat dilihat pada Tabel 3.72.

Tabel 3.72 Hasil Perhitungan E_i 1 Level 2

NP	86	256	166	233	265	183	E_i
86	0	0	0	0	0	0	0
256	0	0	0	0	0	0	0
166	0	0	0	0	0	0	0
233	0	0	0	0	0	0	0
265	0	0	0	0	0	0	0
183	0	0	0	0	0	0	0

Selanjutnya dilakukan perhitungan $\delta\alpha_i$ seperti pada Persamaan 2.27. Hasil dari perhitungan nilai $\delta\alpha_i$ dapat dilihat pada Tabel 3.73. Berikut contoh perhitungan nilai $\delta\alpha_1$ untuk nomor peserta 86 :

$$\begin{aligned} \delta\alpha_j &= \min\{\max[\gamma(1 - E_i), -\alpha_j], C - \alpha_j\} \\ \delta\alpha_1 &= \min\{\max[0.03999783(1 - 0), -0], 1 - 0\} \\ \delta\alpha_1 &= 0.03999783 \end{aligned}$$



Tabel 3.73 Hasil Perhitungan Nilai Delta Alpha Level 2

NP	86	256	166	233	265	183
$\delta\alpha_i$	0.03999783	0.03999783	0.03999783	0.03999783	0.03999783	0.03999783

Setelah didapatkan nilai $\delta\alpha_i$ maka selanjutnya memperbaharui nilai α_i seperti pada Persamaan 2.26. Berikut contoh perhitungan nilai α_i adalah $\alpha_1 = 0 + 0.03999783 = 0.03999783$. Hasil perhitungan nilai α_i secara keseluruhan dapat dilihat pada Tabel 3.74.

Tabel 3.74 Hasil Perhitungan Nilai Alpha Level 2

NP	86	256	166	233	265	183
α_i	0.03999783	0.03999783	0.03999783	0.03999783	0.03999783	0.03999783

Setelah nilai α_i diperbaharui, selanjutnya melakukan pengecekan iterasi. Apabila nilai iterasi telah mencapai nilai konvergen dimanan $\max(|\delta\alpha_i| < \varepsilon)$, maka iterasi akan dihentikan. Selain pada kondisi tersebut iterasi akan dihentikan jika telah mencapai maksimum iterasi yang ditentukan. Nilai α_i bau pada iterasi sebelumnya akan digunakan untuk perhitungan pada proses iterasi selanjutnya. Diketahui nilai α_1 pada iterasi 1 adalah 0.03999783. hasil dari perhitungan iterasi 2 level 2 dapat dilihat pada Tabel 3.75.

Tabel 3.75 Hasil Perhitungan E_i 2 Level 2

NP	86	256	166	233	265	183	Ei
86	0.00999964	0.00999996	0.00999996	-0.00999996	- 0.00999995	- 0.00999996	0.00000015
256	0.00999996	0.00999996	0.009999596	-0.00999996	- 0.00999995	- 0.00999995	0.00000019
166	0.00999996	0.00999996	0.009999596	-0.00999996	- 0.00999995	- 0.00999995	0.00000019
233	-0.00999996	- 0.00999955	-0.00999955	0.00999962	0.00999995	0.00999995	0.00000001
265	-0.00999995	- 0.00999952	-0.00999952	0.00999952	0.00999996	0.00999996	0.00000011
183	-0.00999996	-0.00999995	-0.00999954	0.00999954	0.00999996	0.01	0.00000046

Selanjutnya dilakukan perhitungan $\delta\alpha_i$ seperti pada Persamaan 2.27. Hasil perhitungan nilai $\delta\alpha_i$ dapat dilihat pada Tabel 3.76. berikut contoh perhitungan nilai $\delta\alpha_i$:

$$\delta\alpha_i = \min\{\max[\gamma(1 - E_i), -\alpha_i], C - \alpha_i\}$$

$$\delta\alpha_1 = \min\{\max[0.03999783(1 - 0.00000015), -0.03999783], 1 - 0.03999783\}$$

$$\delta\alpha_1 = 0.03999783$$



Tabel 3.76 Hasil Perhitungan Nilai Delta Alpha Iterasi 2 Level 2

NP	86	256	166	233	265	183
$\delta\alpha_i$	0.039995646	0.039995645	0.039995645	0.039995652	0.039995648	0.039995634

Setelah didapat nilai $\delta\alpha_i$ maka langkah selanjutnya yakni memperbaharui nilai α_i seperti pada Persamaan 2.28. Berikut contoh perhitungan nilai α_i adalah $\alpha_1 = 0.03999783 + 0.03999783 = 0.07999051485172610$. Hasil perhitungan nilai α_i dapat dilihat pada Tabel 3.77.

Tabel 3.77 Hasil Perhitungan Nilai Alpha Iterasi 2 Level 2

NP	86	256	166	233	265	183
α_i	0.0799905148	0.0799904769	0.0799905189	0.079990528	0.079990527	0.079990526

Hasil dari perhitungan nilai alpha pada Tabel 3.77 akan digunakan untuk menentukan *support vector* pada perhitungan selanjutnya. *Support Vector* adalah perhitungan yang ditentukan dari nilai α_i pada iterasi terakhir (dalam perhitungan ini yakni pada iterasi ke-2). Nilai α_i dari data *training* dianggap sebagai *support vector* apabila nilainya lebih dari nol sehingga semuanya merupakan *support vector*.

Selanjutnya dihitung nilai bobot data *training*. dalam perhitungan bobot data *training* terdapat 2 bagian, pertama bobot data *training* di *dot-product* dengan menggunakan nilai data *training* yang memiliki nilai alpha tertinggi kelas positif ($w.x^+$), kedua bobot data *training* di *dot-product* dengan nilai data *training* yang memiliki nilai alpha tertinggi pada kelas negatif ($w.x^-$). Perhitungan pada setiap bobot menggunakan perhitungan kernel dari data *training* yang memiliki kelas negatif maupun kelas positif untuk masing-masing bobot positif dan negatif. Diketahui nilai x^+ pada nomor peserta 86, dan nilai x^- pada nomor peserta 233. Data awal setelah dilakukan normalisasi dikalikan dengan data yang memiliki nilai *support vector* positif yakni nomor peserta 86 dan data data yang memiliki nilai *support vector* negatif yakni nomor peserta 233. Adapun perhitungan untuk menentukan kelas positif dan negatif dilakukan pada data *training* pada nomor peserta 86 dengan nilai $K(x^+, x_i)$ dimana $x=86$ $y=86$ berikut :

$$\begin{aligned}
 [x.y]^d &= [(0.009 \times 0.009) + (0.007 \times 0.007) + (0.017 \times 0.017) \\
 &\quad + (0.002 \times 0.002) + (0.008 \times 0.008) + (0.004 \times 0.004) \\
 &\quad + (0.008 \times 0.008) + (0.007 \times 0.007) + (0.024 \times 0.024) \\
 &\quad + (0.002 \times 0.002) + (0.003 \times 0.003) + (0.003 \times 0.003) \\
 &\quad + (0.016 \times 0.016) + (0.003 \times 0.003) + (0.015 \times 0.015) \\
 &\quad + (0.006 \times 0.006) + (0.011 \times 0.011) + (0.003 \times 0.003) \\
 &\quad + (0.003 \times 0.003) + (0.003 \times 0.003) + (0.020 \times 0.002) \\
 &\quad + (0.002 \times 0.002) + (0.015 \times 0.015)]^2 = 0.00000452
 \end{aligned}$$



Berikut contoh perhitungan untuk $K(x^-, x_i)$ untuk $x=86$ $y=233$:

$$\begin{aligned}
 [x \cdot y]^d = & [(0.009 \times 0.009) + (0.007 \times 0.007) + (0.017 \times 0.003) \\
 & + (0.002 \times 0.002) + (0.008 \times 0.008) + (0.004 \times 0.004) \\
 & + (0.008 \times 0.008) + (0.007 \times 0.007) + (0.024 \times 0.024) \\
 & + (0.002 \times 0.002) + (0.003 \times 0.003) + (0.003 \times 0.003) \\
 & + (0.016 \times 0.016) + (0.003 \times 0.003) + (0.015 \times 0.002) \\
 & + (0.006 \times 0.006) + (0.011 \times 0.011) + (0.003 \times 0.003) \\
 & + (0.003 \times 0.003) + (0.003 \times 0.003) + (0.020 \times 0.020) \\
 & + (0.002 \times 0.002) + (0.015 \times 0.015)]^2 = 0.00000299
 \end{aligned}$$

Hasil perhitungan dari $K(x^+, x_i)$ dan $K(x^-, x_i)$ secara keseluruhan dapat dilihat pada Tabel 3.78.

Tabel 3.78 Mencari Nilai Kernel x^+ dan x^-

NP	$K(x^+, x_i)$	$K(x^-, x_i)$
86	0.00000452	0.00000299
256	0.00000401	0.00000258
166	0.00000359	0.00000225
233	0.00000299	0.00000408
265	0.00000214	0.00000160
183	0.00000282	0.00000219

Selanjutnya dihitung nilai bobotnya untuk mencari nilai $\alpha_i y_i K(x_i, x)$, jika diketahui nilai $\alpha_1 = 0.079995646$ dan nilai $y_i = 1$, maka hasil perhitungan untuk bobot $w \cdot x^+$ dan $w \cdot x^-$ dapat dilihat pada Tabel 3.79. berikut contoh perhitungan untuk $w \cdot x^+$ dan $w \cdot x^-$ untuk nomor peserta 86 :

$$w \cdot x^+ = \left(\sum_{i=1}^n \alpha_i y_i x_i \right) \cdot x^+ = \sum_{i=1}^n \alpha_i y_i K(x_i, x^+)$$

$$\begin{aligned}
 \sum w \cdot x^+ = & ((0.0799905148 \times 1 \times 0.00000452) \\
 & + (0.0799904769 \times 1 \times 0.00000401) \\
 & + (0.079990518 \times 1 \times 0.00000359) \\
 & + (0.079990528 \times -1 \times 0.00000299) \\
 & + (0.079990527 \times -1 \times 0.00000214) + (0.079990526 \\
 & \times -1 \times 0.00000282)) = 0.000000333
 \end{aligned}$$



$$\begin{aligned} \sum w \cdot x^- = & ((0.0799905148 \times 1 \times 0.00000299) \\ & + (0.0799904769 \times 1 \times 0.00000258) \\ & + (0.079990518 \times 1 \times 0.00000225) \\ & + (0.079990528 \times -1 \times 0.00000408) \\ & + (0.079990527 \times -1 \times 0.00000160) + (0.079990526 \\ & \times -1 \times 0.00000219)) = -0.000000003 \end{aligned}$$

Tabel 3.79 Nilai Bobot Kelas Positif dan Negatif Level 2

NP	$w \cdot x^+$	$w \cdot x^-$
86	0.000000361	0.000000239
256	0.000000321	0.000000207
166	0.000000287	0.000000180
233	-0.000000239	-0.000000326
265	-0.000000171	-0.000000128
183	-0.000000225	-0.000000175
Σ	0.000000333	-0.000000003

Selanjutnya melakukan penghitungan nilai bias yang merupakan posisi bidang relatif terhadap pusat koordinat, dimana pada penghitungannya membutuhkan jumlah dari kelas bobot positif dan jumlah dari kelas bobot negatif. Nilai bias didapatkan dengan melakukan penjumlahan bobot kelas positif dan jumlah dari bobot kelas negatif dan dikalikan dengan $-\frac{1}{2}$ seperti Persamaan 2.14.

Berikut contoh penghitungan nilai bias dimana b adalah bias.

$$b = -\frac{1}{2}(0.000000333 + 0.000000003) = -0.00000017$$

Selanjutnya menghitung nilai fungsi $f(x)$ yang dilakukan terhadap data *testing* dengan menjumlahkan nilai dari total bobot data *testing* dengan nilai bias yang sudah dihitung pada proses sebelumnya dimana akan menghasilkan tanda dari nilai yang dihasilkan oleh fungsi $f(x)$. Beberapa hal yang harus dipenuhi adalah jika level satu bernilai positif maka akan masuk kelas +1 dan masuk kelas A (sangat layak), sedangkan apabila bernilai negatif maka akan masuk kelas -1 dan akan dihitung lagi pada level 2. Jika pada klasifikasi level 2 bernilai +1 maka akan masuk ke dalam kelas B (layak), sedangkan jika bernilai -1 maka akan langsung masuk ke dalam kelas C (tidak layak). Data uji yang digunakan untuk menghitung fungsi $f(x)$ seperti pada Tabel 3.80. Berikut contoh perhitungan untuk data uji ke-1 :



$$\begin{aligned}
 [x.y]^d = & [(0.009 \times 0.009) + (0.007 \times 0.007) + (0.017 \times 0.017) \\
 & + (0.002 \times 0.002) + (0.008 \times 0.008) + (0.004 \times 0.004) \\
 & + (0.008 \times 0.008) + (0.007 \times 0.007) + (0.024 \times 0.024) \\
 & + (0.002 \times 0.002) + (0.003 \times 0.003) + (0.003 \times 0.003) \\
 & + (0.016 \times 0.002) + (0.003 \times 0.003) + (0.015 \times 0.015) \\
 & + (0.006 \times 0.006) + (0.011 \times 0.011) + (0.003 \times 0.003) \\
 & + (0.003 \times 0.003) + (0.003 \times 0.003) + (0.002 \times 0.002) \\
 & + (0.002 \times 0.002) + (0.015 \times 0.015)]^2 = 0.000003592
 \end{aligned}$$

Selanjutnya dilakukan penghitungan untuk $\alpha_i y_i K(x_i, x)$ sebanyak seluruh data *training*. seluruh hasil perhitungan bobot pada data uji ke-1 untuk $\alpha_i y_i K(x_i, x)$ akan dijumlahkan. Berikut contoh perhitungan untuk $\alpha_i y_i K(x_i, x)$ pada nomor peserta 86 :

$$\alpha_1 y_1 K(x_1, x) = 0.079995646 \times 1 \times 0.000003592 = 0.000000287$$

Tabel 3.80 Hasil Pengujian Data Uji 1 Level 2

NP	$K(x_i, x)$	$\alpha_i y_i K(x_i, x)$
86	0.000003592	0.000000287
256	0.000003496	0.000000280
166	0.000003496	0.000000280
233	0.000002249	-0.000000180
265	0.000001523	-0.000000122
183	0.000002099	-0.000000168
Σ		0.000000377

Setelah dilakukan perhitungan kernel dan bobot untuk data uji ke-1 kelas 1, maka dilanjutkan untuk penghitungan kernel dan bobot pada data uji ke-2 seperti pada Tabel 3.81.

Tabel 3.81 Hasil Perhitungan Data Uji 2 Level 2

NP	$K(x_i, x)$	$\alpha_i y_i K(x_i, x)$
86	0.000002882	0.000000231
256	0.000002796	0.000000224
166	0.000002796	0.000000224
233	0.000002184	-0.000000175
265	0.000001074	-0.000000086
183	0.000001566	-0.000000125
Σ		0.000000292

Apabila telah didapatkan hasil nilai pengujian terhadap masing-masing data uji, maka selanjutnya dilakukan perhitungan untuk mendapatkan nilai $f(x)$



seperti pada Persamaan 2.20. Hasil dari perhitungan nilai $f(x)$ dapat dilihat pada Tabel 3.82. berikut contoh perhitungan untuk mendapatkan nilai $f(x)$ pada data uji ke-1 Level 2.

$$f(x) = \text{sign}(0.000000377 + (-0.00000017))$$

$$f(x) = 0.00000021$$

$$f(x) = 1$$

Proses penghitungan untuk mendapatkan hasil klasifikasi dilakukan pada keseluruhan data *testing* yang terpilih. Hasil klasifikasi pada level 2 dengan nilai 1 masuk dalam kategori B (layak), sedangkan hasil klasifikasi -1 pada level 2 masuk dalam kategori C (tidak layak).

Tabel 3.82 Hasil Perhitungan $f(x)$ dan Klasifikasi

NP	$f(x)$ level 1	$f(x)$ level 2	Hasil Klasifikasi	Actual Class
135	0.00000665	-	1 dilevel 1 A	A
226	-0.00000266	0.00000021	1 dilevel 2 B	B
185	-0.00000295	0.000000127	1 dilevel 2 B	C

3.3.2.4 Perhitungan Manual Nilai Akurasi

Perhitungan akurasi bertujuan untuk mengukur tingkat keberhasilan dari suatu sistem atau membandingkan hasil implementasi yang telah dilakukan dengan standar yang dilakukan dalam proses pengujian. Proses penghitungan akurasi dilakukan dengan menggunakan *confusion* matriks seperti pada Tabel 3.83.

Tabel 3.83 Model Evaluasi *Confusion Matriks*

ACTUAL CLASS	PREDICTED CLASS			JUMLAH
	A	B	C	
A	0	0	1	1
B	0	1	0	1
C	0	1	0	1

Untuk menghitung nilai akurasi sistem menggunakan Persamaan 2.30 sebagai berikut :

$$\text{Akurasi} = \frac{TP + TN}{TP + TN + FP + FN}$$

Dari Persamaan di atas, berikut nilai akurasi yang didapatkan :

- $TP = (C_{1,1}) + (C_{2,2}) + (C_{3,3}) = 0 + 1 + 0 = 1$
- $TN = (C_{1,1} + C_{2,2}) + (C_{1,1} + C_{3,3}) + (C_{2,2} + C_{3,3}) = 1 + 0 + 1 = 2$



c. $FP = (C_{2,1} + C_{3,1}) + (C_{1,2} + C_{3,2}) + (C_{1,3} + C_{2,3}) = 0 + 1 + 1 = 2$

d. $FN = (C_{1,2} + C_{1,3}) + (C_{2,1} + C_{2,3}) + (C_{3,1} + C_{3,2}) = 1 + 0 + 1 = 2$

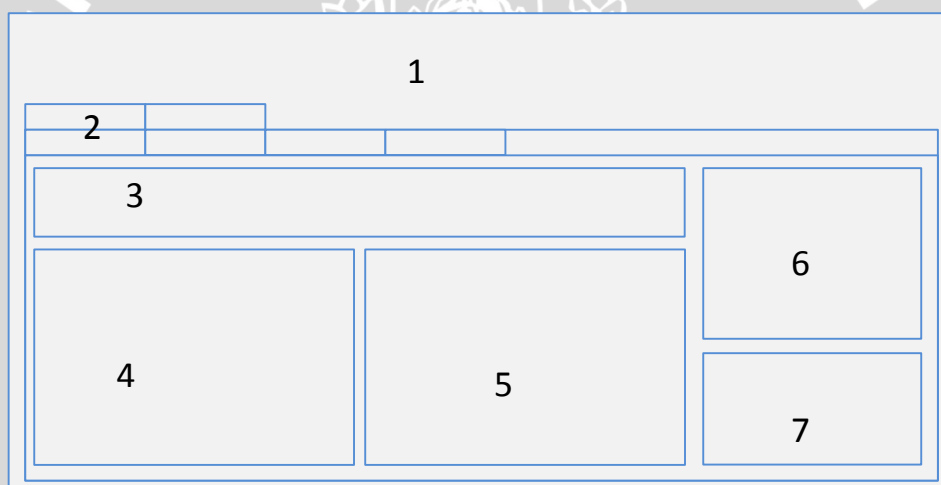
$$Akurasi = \frac{2 + 4}{2 + 4 + 1 + 1} = \frac{6}{8} = 75\%$$

3.4 Implementasi Sistem

Implementasi pada penelitian ini mengacu kepada perancangan sistem yang dilakukan menggunakan bahasa pemrograman Java, manajemen *database* menggunakan MySQL serta *tools* pendukung lainnya.

3.4.1 Sub Sistem Antar Muka

Sub sistem antar muka bertujuan untuk mempermudah setiap pengguna untuk berkomunikasi dan melakukan interaksi dengan sistem. Sub sistem antar muka meliputi perancangan mulai dari letak menu yang akan digunakan beserta antar muka dimana sistem akan menampilkan hasil perhitungannya. Perancangan halaman utama sistem ini ditunjukkan pada Gambar 3.19.



Gambar 3.19 Perancangan Antarmuka Sistem

Gambar 3.19 Merupakan perancangan antarmuka halaman utama sistem klasifikasi penerima beasiswa dengan metode AHP-SVM.

Keterangan :

1. Nomor 1 adalah untuk menampilkan judul dari sistem.
2. Nomor 2 adalah tab yang akan menampung menu-menu yang akan digunakan di dalam sistem.
3. Nomor 3 adalah untuk menampung *field* berupa masukan dari pengguna.

4. Nomor 4,5,6 dan 7 adalah untuk menampilkan hasil perhitungan sistem menggunakan metode AHP dan SVM.

3.5 Perancangan Pengujian

Pada perancangan pengujian menjelaskan mengenai beberapa pengujian yang akan dilakukan terhadap klasifikasi penerima beasiswa GNOTA ini, adapun pengujian terhadap sistem meliputi :

1. Pengujian untuk mengetahui perbandingan jumlah data *training* dan data *testing* terhadap hasil akurasi.
2. Pengujian tingkat akurasi berdasarkan nilai *threshold* AHP.
3. Pengujian tingkat akurasi berdasarkan nilai parameter *Lamda*.
4. Pengujian tingkat akurasi berdsarkan nilai konstanta *gamma*.
5. Pengujian tingkat akurasi berdasarkan nilai *epsilon*.
6. Pengujian tingkat akurasi berdasarkan nilai Iterasi Maksimum.
7. Pengujian tingkat akurasi berdasarkan pengaruh nilai C (*Complexity*).

3.5.1 Pengujian tingkat akurasi terhadap perbandingan jumlah data *training* dan data *testing*

Pengujian terhadap perbandingan jumlah data *training* dan data *testing* terhadap nilai akurasi untuk mengetahui banyaknya data *training* dan data *testing* dalam perbandingan yang mempengaruhi nilai akurasi. Perancangan pengujian untuk mengetahui perbandingan jumlah data *training* dan data *testing* terhadap nilai akurasi seperti yang ditunjukkan pada Tabel 3.84.

Tabel 3.84 Perancangan Pengujian Perbadingan Jumlah Data *Training* dan Data *Testing*

Iterasi	Percobaan Ke- <i>i</i>					Rata-rata
	1	2	3	4	5	
90 : 10						
80 : 20						
70 : 30						
60 : 40						
50 : 50						
40 : 60						
30 : 70						
20 : 80						
10 : 90						

3.5.2 Pengujian tingkat akurasi berdasarkan pengaruh nilai *Threshold* AHP.

Pengujian terhadap pengaruh nilai *threshold* AHP dilakukan setelah mendapatkan akurasi terbaik dari pengujian terhadap perbandingan data *training* dan data *testing*. Berikut skenario pengujian berdasarkan pengaruh nilai *threshold* AHP seperti pada Tabel 3.85.

Tabel 3.85 Skenario Pengujian Berdasarkan Nilai *Threshold* AHP

Iterasi	Skenario Pengujian					Rata-rata
	Banyak fitur	1	2	3	4	
0.001						
0.0035						
0.01						
0.015						
0.02						
0.025						
0.03						
0.035						
0.04						
0.07						

3.5.3 Pengujian Berdasarkan Nilai Parameter *Lamda*

Pengujian berdasarkan nilai parameter *lamda* bertujuan untuk mengetahui pengaruh dari nilai parameter *lamda* yang digunakan terhadap tingkat akurasi yang dihasilkan. Berikut skenario pengujian terhadap pengaruh nilai dari parameter *lamda* seperti pada Tabel 3.86.

Tabel 3.86 Skenario Pengujian Berdasarkan Nilai Parameter *Lamda*

Lamda	Pengujian					Rata-rata
	1	2	3	4	5	
0.5						
1						
1.5						

2						
2.5						
3						
4						
10						
15						
30						

3.5.4 Pengujian tingkat akurasi berdasarakan nilai konstanta *gamma*

Pengujian tingkat akurasi berdasarakan pengaruh nilai konstanta *gamma* untuk mengetahui nilai dari parameter konstanta *gamma* yang terbaik untuk melakukan pengujian selanjutnya. Berikut skenario pengujian terhadap tingkat akurasi berdasarakan nilai konstanta *gamma* seperti pada Tabel 3.87.

Tabel 3.87 Skenario Pengujian Nilai Konstanta Gamma

Konstanta Gamma	Percobaan Ke- <i>i</i>					Rata-rata
	1	2	3	4	5	
0.0001						
0.0005						
0.003						
0.005						
0.01						
0.02						
0.05						
0.1						
0.5						
1						

3.5.5 Pengujian tingkat akurasi terhadap pengaruh nilai *epsilon*

Pengujian tingkat akurasi terhadap pengaruh nilai *epsilon* bertujuan untuk mengetahui tingkat akurasi terbaik yang dihasilkan dari beberapa percobaan nilai *epsilon*. Berikut skenario pengujian terhadap pengaruh nilai *epsilon* seperti pada Tabel 3.88.

Tabel 3.88 Skenario Pengujian Terhadap Pengaruh Nilai *Epsilon*

Epsilon	Percobaan Ke- <i>i</i>	Rata-
---------	------------------------	-------



	1	2	3	4	5	rata
0.000000001						
0.00000001						
0.0000001						
0.000001						
0.00001						
0.0001						
0.001						
0.01						
0.1						
1						

3.5.6 Pengujian tingkat akurasi terhadap pengaruh iterasi maksimum

Pengujian terhadap tingkat akurasi dari pengaruh iterasi maksimum yang terbaik untuk mengetahui tingkat akurasi yang dihasilkan. Nilai iterasi maksimum yang akan diuji seperti pada Tabel 3.89.

Tabel 3.89 Skenario Pengujian Iterasi Maksimum

Iterasi	Percobaan Ke- <i>i</i>					Rata-rata
	1	2	3	4	5	
10						
20						
30						
50						
100						
500						
1000						
2000						
3000						
5000						

3.5.7 Pengujian tingkat akurasi berdasarkan nilai C (*Complexity*)

Pengujian terhadap pengaruh nilai C (*Complexity*) kepada tingkat akurasi yang dihasilkan sehingga akan mendapatkan nilai C yang terbaik yang akan



digunakan. Perancangan pengujian terhadap pengaruh nilai C terhadap akurasi seperti pada Tabel 3.90.

Tabel 3.90 Skenario Pengujian terhadap pengaruh nilai C terhadap akurasi

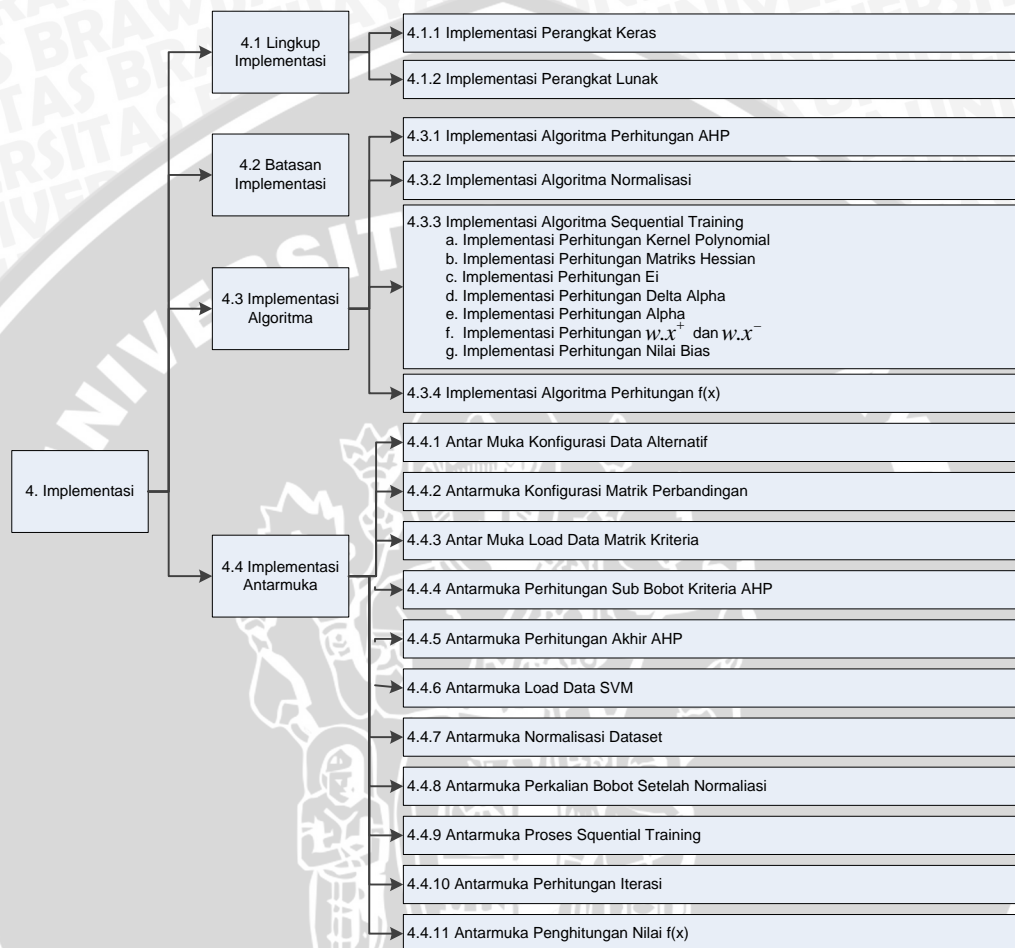
Complexity	Percobaan Ke- <i>i</i>					Rata-rata
	1	2	3	4	5	
1						
10						
20						
30						
40						
50						
60						
70						
80						
90						

3.6 Pengambilan Kesimpulan dan Saran

Kesimpulan didapatkan setelah semua tahapan meliputi analisis, perancangan, implementasi serta pengujian telah selesai dilaksanakan. Kesimpulan didapatkan berdasarkan hasil dari pengujian sistem dan analisa dari penggunaan metode AHP dan SVM dengan tujuan menjawab rumusan masalah yang telah ditetapkan sebelumnya. Selain kesimpulan, penulisan saran akan sangat bermanfaat untuk memberikan masukan dan pertimbangan atas hasil yang didapat untuk memberikan pertimbangan atas pengembangan penelitian selanjutnya.

BAB 4 IMPLEMENTASI

Bab ini mengenai implementasi dari metode AHP-SVM untuk klasifikasi penerima beasiswa pada Lembaga GNOTA Kediri yang meliputi spesifikasi sistem, batasan implementasi, implementasi algoritma dan implementasi antarmuka. Diagram implementasi sistem seperti pada Gambar 4.1.



Gambar 4.1 Diagram Implementasi Sistem

Implementasi dari sistem klasifikasi penerima beasiswa pada lembaga GNOTA Kediri dengan menggunakan metode AHP dan SVM dapat dilihat seperti pada Gambar 4.1.

4.1 Lingkup Implementasi

Sub bab ini membahas mengenai lingkup perangkat keras dan perangkat lunak yang digunakan untuk membangun sistem yang sesuai dengan kebutuhan.

4.1.1 Lingkup Implementasi Perangkat Keras

Lingkup implementasi perangkat keras yang digunakan dalam membangun penelitian ini seperti yang ditunjukkan pada Tabel 4.1.

Tabel 4.1 Implementasi Perangkat Keras

Komponen	Spesifikasi
Processor	AMD A4-5000 APU with Radeon(TM) HD Graphics 1.50 GHz
Memory	4 GB RAM
Harddisk	500 GB

4.1.2 Lingkup Implementasi Perangkat Lunak

Lingkup implementasi perangkat lunak yang digunakan dalam membangun penelitian ini seperti yang ditunjukkan pada Tabel 4.2.

Tabel 4.2 Lingkup Implementasi Perangkat Lunak

Tipe Perangkat Lunak	Spesifikasi
Sistem Operasi	Microsoft Windows 8 Pro 64-bit
Bahasa Pemrograman	JAVA
Database	MySQL
Tools Pemrograman	Netbeans IDE 8.0.2
Tools DBMS	SQLyog Ultimate v.9.0.2

4.2 Batasan Implementasi

Batasan implementasi yang terdapat dalam membangun sistem Penerapan Metode AHP-SVM untuk Klasifikasi Penerima Beasiswa pada Lembaga GNOTA Kediri sebagai berikut :

1. Penelitian ini menggunakan kriteria sejumlah 13 kriteria yang akan dilakukan penghitungan dengan proses AHP.
2. Pada penelitian ini tidak melakukan uji sensitifitas terhadap kriteria AHP.
3. Data dalam pengembangan penelitian ini adalah data hasil seleksi penerimaan beasiswa pada lembaga GNOTA Kediri pada Tahun 2015 sebanyak 111 data.
4. Implementasi SVM menggunakan Kernel *Polynomial degree* dengan nilai $d=2$.
5. Hasil evaluasi dalam bentuk akurasi pada metode SVM dengan menggunakan evaluasi *Confussion Matriks*.

4.3 Implementasi Algoritma

Sistem Klasifikasi Penerima Beasiswa di Lembaga GNOTA Kediri dengan Penerapan Metode AHP dan SVM memiliki 6 algoritma yang ada pada sistem seleksi dan rekomendasi, yang mana total keseluruhan ada 6 algoritma, algoritma tersebut dijelaskan pada Gambar 4.1 pada Diagram Implementasi bagian implementasi algoritma, berikut penjelasan dari masing-masing algoritma yang sudah di jelaskan sebelumnya sehingga nanti pada implementasi dapat dicek satu persatu pada Kode Program.

4.3.1 Implementasi Algoritma Perhitungan AHP

Proses implementasi perhitungan AHP diawali dengan proses normalisasi matrik perbandingan kriteria yang telah di *load* dari database MySQL untuk selanjutnya dilakukan perhitungan terhadap konsistensi dari nilai matrik perbandingan berpasangan tersebut. Pada perhitungan sub bobot kriteria AHP digunakan cara yang sama dengan perhitungan bobot kriteria seperti pada Kode Program 4.1 yang menunjukkan implementasi proses perhitungan bobot dengan AHP.

```
1 public void NormalisasiA(){
2     for(i=0;i<baris;i++){
3         for(j=1;j<kolom;j++){
4
5             matrix_norm[i][j]=matrix_ku[i][j]/bb_k[j];
6             matrix_norm[i][j] = bd.floatValue();
7         }
8     } //end for i
9     for(i=0;i<baris;i++){
10        tempNorm=0;
11        for(j=1;j<kolom;j++){
12            tempNorm=tempNorm+matrix_norm[i][j];
13            jum_norm[i]=tempNorm;
14            jum_norm[i] = bd.floatValue();
15        } //for b
16    } //for jml perbaris
17    for(i=0;i<baris;i++){
18        temp_c[i]=tblKriteriaUtama.getValueAt(i,
19        0).toString();
20    } //akhir normalisasi
21 public void hitungBobotAhp(){
    for(i=0;i<baris;i++){
```

```
22         tempBobot=0;
23         for(j=1;j<kolom;j++){
24             tempBobot+=matrix_norm[i][j];
25         }
26         hitung_bobot[i]=tempBobot/n_kriteria;
27         hitung_bobot[i] = bd.floatValue();
28     }
29     for( i=0;i<baris;i++){
30         tempAx=0;
31         for(j=0;j<kolom-1;j++){
32             tempAx+=matrix_ku[i][j]*hitung_bobot[j];
33             ax[i]=tempAx;
34         }
35     }
36 } //end hitungbobot ahp
37
38 public void lamdaMax(){
39     lamda=0; tempLamda=0;
40     for(i=0;i<baris;i++){
41         tempLamda +=ax[i]/hitung_bobot[i];
42     }
43     lamda=tempLamda/n_kriteria;
44 } //end hitung lamda max
45 public void CI(){
46     CI=0;
47     CI=(lamda-n_kriteria)/(n_kriteria-1);
48 } //pv ci
49 public void CR(){
50     CR=0;IR=0;
51     try{
52         String SQL = "SELECT indek,ri FROM konsistensi
53 WHERE indek="+n_kriteria+"";
54         ResultSet cons = Koneksi.executeQuery(SQL);
55         while(cons.next()){
56             IR=Float.parseFloat(txtRI.getText());
57         }
58         CR=CI/IR;
59         if(CR<0.1){
60             lblKonsistensi.setText("KONSISTEN");
61         }
62     } catch (SQLException ex) {
63         JOptionPane.showMessageDialog(this, ex.getMessage());
64     }
65 }
```

```

60     }
61     else{lblKonsistensi.setText("TIDAK
KONSISTEN");}
62     cons.close();
63     }catch(Exception kr){
64         JOptionPane.showMessageDialog(null,"GAGAL");
65     } }//pv cr

```

Kode Program 4.1 Proses Perhitungan AHP

Berikut adalah penjelasan Kode Program 4.1.

1. Baris 1-19 merupakan proses perhitungan normalisasi matrik perbandingan berpasangan AHP
2. Baris 20-36 merupakan proses perhitungan bobot AHP.
3. Baris 38-44 merupakan proses pencarian nilai lamda maksimum.
4. Baris 45-48 merupakan proses perhitungan nilai *CI*
5. Baris 49-65 merupakan proses perhitungan nilai *CR*

4.3.2 Implementasi Algoritma Normalisasi

Berikut merupakan implementasi algoritma normalisasi dataset seperti pada Kode Program 4.2.

```

1  Public void NormalisasiLatih
2
3  ba=Float.parseFloat(tf_ba.getText());bb=Float.parseFloat(tf
_bb .getText());
4
5  matrik_norm_latih=new float[baris_lt][kolom_lt];
6  col_name_lt=new String[kolom_lt];
7  for(j=1;j<kolom_lt;j++){
8  col_name_lt[j]=tblDataLatih.getModel().getColumnName(j);
9  }
10 for(i=0;i<baris_lt;i++){
11     for(j=1;j<kolom_lt;j++){
12         if(col_name_lt[j].equals("P1") ||
13             col_name_lt[j].equals("P2") ||
14             col_name_lt[j].equals("P3") ||
15             col_name_lt[j].equals("P4") ||
16             col_name_lt[j].equals("P5")){
17             matrik_norm_latih[i][j]=((matrik_latih[i][j]-
minC1) / (maxC1-minC1)) * (ba-bb)+bb;
18         }
19     }
20 }

```

```

18 else if(col_name_lt[j].equals("P6") ||
19 col_name_lt[j].equals("P7") ||
20 col_name_lt[j].equals("P8") ){
21 matrik_norm_latih[i][j]=((matrik_latih[i][j]-
minC2)/(maxC2-minC2))*(ba-bb)+bb;
22 }else if(col_name_lt[j].equals("P9") ||
23 col_name_lt[j].equals("P10") ||
24 col_name_lt[j].equals("P11") ||
25 col_name_lt[j].equals("P12") ||
26 col_name_lt[j].equals("P13")){
27 matrik_norm_latih[i][j]=((matrik_latih[i][j]-
minC3)/(maxC3-minC3))*(ba-bb)+bb;
28 }else if(col_name_lt[j].equals("P14") ||
29 col_name_lt[j].equals("P15") ||
30 col_name_lt[j].equals("P16") ||
31 col_name_lt[j].equals("P17") ||
32 col_name_lt[j].equals("P18") ||
33 col_name_lt[j].equals("P19")){
34 matrik_norm_latih[i][j]=((matrik_latih[i][j]-
minC4)/(maxC4-minC4))*(ba-bb)+bb;
35 }else if(col_name_lt[j].equals("P20") ||
36 col_name_lt[j].equals("P21") ||
37 col_name_lt[j].equals("P22") ||
38 col_name_lt[j].equals("P23") ||
39 col_name_lt[j].equals("P24") ){
40 matrik_norm_latih[i][j]=((matrik_latih[i][j]-
minC5)/(maxC5-minC5))*(ba-bb)+bb;
41 }else if(col_name_lt[j].equals("P25") ||
42 col_name_lt[j].equals("P26") ||
43 col_name_lt[j].equals("P27")){
44 matrik_norm_latih[i][j]=((matrik_latih[i][j]-
minC7)/(maxC7-minC7))*(ba-bb)+bb;
45 }else if(col_name_lt[j].equals("P31") ||
46 col_name_lt[j].equals("P32") ||
47 col_name_lt[j].equals("P33")){
48 matrik_norm_latih[i][j]=((matrik_latih[i][j]-
minC8)/(maxC8-minC8))*(ba-bb)+bb;
49 }else if(col_name_lt[j].equals("P34") ||
50 col_name_lt[j].equals("P35") ||
51 col_name_lt[j].equals("P36")){
52 matrik_norm_latih[i][j]=((matrik_latih[i][j]-
minC9)/(maxC9-minC9))*(ba-bb)+bb;
53 }else if(col_name_lt[j].equals("P37") ||
54 col_name_lt[j].equals("P38") ||
55 col_name_lt[j].equals("P39") ||
56 col_name_lt[j].equals("P40")){

```

```

55 matrik_norm_latih[i][j]=(matrik_latih[i][j]-
56 minC10)/(maxC10-minC10)*(ba-bb)+bb;
57         }else if(col_name_lt[j].equals("P41") ||
58 col_name_lt[j].equals("P42") ||
59 col_name_lt[j].equals("P43") ||
60 col_name_lt[j].equals("P44") ||
61 col_name_lt[j].equals("P45")){
62 matrik_norm_latih[i][j]=(matrik_latih[i][j]-
63 minC11)/(maxC11-minC11)*(ba-bb)+bb;
64         }else if(col_name_lt[j].equals("P46") ||
65 col_name_lt[j].equals("P47") ||
66 col_name_lt[j].equals("P48")){
67 matrik_norm_latih[i][j]=(matrik_latih[i][j]-
68 minC12)/(maxC12-minC12)*(ba-bb)+bb;
69         }else if(col_name_lt[j].equals("P49") ||
70 col_name_lt[j].equals("P50")){
71 matrik_norm_latih[i][j]=(matrik_latih[i][j]-
72 minC13)/(maxC13-minC13)*(ba-bb)+bb;
73     }
74     }
75     }
76     }
77     }
78     }
79     }
80     }
81     }
82     }
83     }
84     }
85     }
86     }
87     }
88     }
89     }
90     }
91     }
92     }
93     }
94     }
95     }
96     }
97     }
98     }
99     }
100    }
101    }
102    }
103    }
104    }
105    }
106    }
107    }
108    }
109    }
110    }
111    }
112    }
113    }
114    }
115    }
116    }
117    }
118    }
119    }
120    }
121    }
122    }
123    }
124    }
125    }
126    }
127    }
128    }
129    }
130    }
131    }
132    }
133    }
134    }
135    }
136    }
137    }
138    }
139    }
140    }
141    }
142    }
143    }
144    }
145    }
146    }
147    }
148    }
149    }
150    }
151    }
152    }
153    }
154    }
155    }
156    }
157    }
158    }
159    }
160    }
161    }
162    }
163    }
164    }
165    }
166    }
167    }
168    }
169    }
170    }
171    }
172    }
173    }
174    }
175    }
176    }
177    }
178    }
179    }
180    }
181    }
182    }
183    }
184    }
185    }
186    }
187    }
188    }
189    }
190    }
191    }
192    }
193    }
194    }
195    }
196    }
197    }
198    }
199    }
200    }
201    }
202    }
203    }
204    }
205    }
206    }
207    }
208    }
209    }
210    }
211    }
212    }
213    }
214    }
215    }
216    }
217    }
218    }
219    }
220    }
221    }
222    }
223    }
224    }
225    }
226    }
227    }
228    }
229    }
230    }
231    }
232    }
233    }
234    }
235    }
236    }
237    }
238    }
239    }
240    }
241    }
242    }
243    }
244    }
245    }
246    }
247    }
248    }
249    }
250    }
251    }
252    }
253    }
254    }
255    }
256    }
257    }
258    }
259    }
260    }
261    }
262    }
263    }
264    }
265    }
266    }
267    }
268    }
269    }
270    }
271    }
272    }
273    }
274    }
275    }
276    }
277    }
278    }
279    }
280    }
281    }
282    }
283    }
284    }
285    }
286    }
287    }
288    }
289    }
290    }
291    }
292    }
293    }
294    }
295    }
296    }
297    }
298    }
299    }
300    }
301    }
302    }
303    }
304    }
305    }
306    }
307    }
308    }
309    }
310    }
311    }
312    }
313    }
314    }
315    }
316    }
317    }
318    }
319    }
320    }
321    }
322    }
323    }
324    }
325    }
326    }
327    }
328    }
329    }
330    }
331    }
332    }
333    }
334    }
335    }
336    }
337    }
338    }
339    }
340    }
341    }
342    }
343    }
344    }
345    }
346    }
347    }
348    }
349    }
350    }
351    }
352    }
353    }
354    }
355    }
356    }
357    }
358    }
359    }
360    }
361    }
362    }
363    }
364    }
365    }
366    }
367    }
368    }
369    }
370    }
371    }
372    }
373    }
374    }
375    }
376    }
377    }
378    }
379    }
380    }
381    }
382    }
383    }
384    }
385    }
386    }
387    }
388    }
389    }
390    }
391    }
392    }
393    }
394    }
395    }
396    }
397    }
398    }
399    }
400    }
401    }
402    }
403    }
404    }
405    }
406    }
407    }
408    }
409    }
410    }
411    }
412    }
413    }
414    }
415    }
416    }
417    }
418    }
419    }
420    }
421    }
422    }
423    }
424    }
425    }
426    }
427    }
428    }
429    }
430    }
431    }
432    }
433    }
434    }
435    }
436    }
437    }
438    }
439    }
440    }
441    }
442    }
443    }
444    }
445    }
446    }
447    }
448    }
449    }
450    }
451    }
452    }
453    }
454    }
455    }
456    }
457    }
458    }
459    }
460    }
461    }
462    }
463    }
464    }
465    }
466    }
467    }
468    }
469    }
470    }
471    }
472    }
473    }
474    }
475    }
476    }
477    }
478    }
479    }
480    }
481    }
482    }
483    }
484    }
485    }
486    }
487    }
488    }
489    }
490    }
491    }
492    }
493    }
494    }
495    }
496    }
497    }
498    }
499    }
500    }
501    }
502    }
503    }
504    }
505    }
506    }
507    }
508    }
509    }
510    }
511    }
512    }
513    }
514    }
515    }
516    }
517    }
518    }
519    }
520    }
521    }
522    }
523    }
524    }
525    }
526    }
527    }
528    }
529    }
530    }
531    }
532    }
533    }
534    }
535    }
536    }
537    }
538    }
539    }
540    }
541    }
542    }
543    }
544    }
545    }
546    }
547    }
548    }
549    }
550    }
551    }
552    }
553    }
554    }
555    }
556    }
557    }
558    }
559    }
560    }
561    }
562    }
563    }
564    }
565    }
566    }
567    }
568    }
569    }
570    }
571    }
572    }
573    }
574    }
575    }
576    }
577    }
578    }
579    }
580    }
581    }
582    }
583    }
584    }
585    }
586    }
587    }
588    }
589    }
590    }
591    }
592    }
593    }
594    }
595    }
596    }
597    }
598    }
599    }
600    }
601    }
602    }
603    }
604    }
605    }
606    }
607    }
608    }
609    }
610    }
611    }
612    }
613    }
614    }
615    }
616    }
617    }
618    }
619    }
620    }
621    }
622    }
623    }
624    }
625    }
626    }
627    }
628    }
629    }
630    }
631    }
632    }
633    }
634    }
635    }
636    }
637    }
638    }
639    }
640    }
641    }
642    }
643    }
644    }
645    }
646    }
647    }
648    }
649    }
650    }
651    }
652    }
653    }
654    }
655    }
656    }
657    }
658    }
659    }
660    }
661    }
662    }
663    }
664    }
665    }
666    }
667    }
668    }
669    }
670    }
671    }
672    }
673    }
674    }
675    }
676    }
677    }
678    }
679    }
680    }
681    }
682    }
683    }
684    }
685    }
686    }
687    }
688    }
689    }
690    }
691    }
692    }
693    }
694    }
695    }
696    }
697    }
698    }
699    }
700    }
701    }
702    }
703    }
704    }
705    }
706    }
707    }
708    }
709    }
710    }
711    }
712    }
713    }
714    }
715    }
716    }
717    }
718    }
719    }
720    }
721    }
722    }
723    }
724    }
725    }
726    }
727    }
728    }
729    }
730    }
731    }
732    }
733    }
734    }
735    }
736    }
737    }
738    }
739    }
740    }
741    }
742    }
743    }
744    }
745    }
746    }
747    }
748    }
749    }
750    }
751    }
752    }
753    }
754    }
755    }
756    }
757    }
758    }
759    }
760    }
761    }
762    }
763    }
764    }
765    }
766    }
767    }
768    }
769    }
770    }
771    }
772    }
773    }
774    }
775    }
776    }
777    }
778    }
779    }
780    }
781    }
782    }
783    }
784    }
785    }
786    }
787    }
788    }
789    }
790    }
791    }
792    }
793    }
794    }
795    }
796    }
797    }
798    }
799    }
800    }
801    }
802    }
803    }
804    }
805    }
806    }
807    }
808    }
809    }
810    }
811    }
812    }
813    }
814    }
815    }
816    }
817    }
818    }
819    }
820    }
821    }
822    }
823    }
824    }
825    }
826    }
827    }
828    }
829    }
830    }
831    }
832    }
833    }
834    }
835    }
836    }
837    }
838    }
839    }
840    }
841    }
842    }
843    }
844    }
845    }
846    }
847    }
848    }
849    }
850    }
851    }
852    }
853    }
854    }
855    }
856    }
857    }
858    }
859    }
860    }
861    }
862    }
863    }
864    }
865    }
866    }
867    }
868    }
869    }
870    }
871    }
872    }
873    }
874    }
875    }
876    }
877    }
878    }
879    }
880    }
881    }
882    }
883    }
884    }
885    }
886    }
887    }
888    }
889    }
890    }
891    }
892    }
893    }
894    }
895    }
896    }
897    }
898    }
899    }
900    }
901    }
902    }
903    }
904    }
905    }
906    }
907    }
908    }
909    }
910    }
911    }
912    }
913    }
914    }
915    }
916    }
917    }
918    }
919    }
920    }
921    }
922    }
923    }
924    }
925    }
926    }
927    }
928    }
929    }
930    }
931    }
932    }
933    }
934    }
935    }
936    }
937    }
938    }
939    }
940    }
941    }
942    }
943    }
944    }
945    }
946    }
947    }
948    }
949    }
950    }
951    }
952    }
953    }
954    }
955    }
956    }
957    }
958    }
959    }
960    }
961    }
962    }
963    }
964    }
965    }
966    }
967    }
968    }
969    }
970    }
971    }
972    }
973    }
974    }
975    }
976    }
977    }
978    }
979    }
980    }
981    }
982    }
983    }
984    }
985    }
986    }
987    }
988    }
989    }
990    }
991    }
992    }
993    }
994    }
995    }
996    }
997    }
998    }
999    }
1000   }

```

Kode Program 4.2 Proses Normalisasi

Berikut penjelasan Kode Program 4.2.

1. Baris 3 merupakan inisialisasi nilai batas atas dan batas bawah dalam proses normalisasi
2. Baris 4-5 merupakan inisialisasi variabel
3. Baris 6-75 merupakan penghitungan normalisasi berdasarkan setiap kriteria.

4.3.3 Implementasi Algoritma Proses *Sequential Training*

Pada proses perhitungan iterasi, hal pertama yang dilakukan adalah inisialisasi variabel, melakukan perhitungan kernel yang digunakan, melakukan perhitungan matrik hessian, mencari nilai E_i , $\delta\alpha_i$, nilai α baru, iterasi dan juga perhitungan untuk mendapatkan nilai bias.

a. Algoritma Perhitungan Kernel *Polynomial Degree d*

Perhitungan kernel *polynomial degree d* dilakukan dengan melakukan *dot product* antar dua buah data training pada ruang vector yang berdimensi. Hasil dari perhitungan kernel *polynomial degree d* berupa matriks kernel

dengan indeks $n \times n$ dimana n mewakili banyaknya jumlah data *training* yang digunakan.

```

1 public void hitungPolynomial1() {
2     degree=Float.parseFloat(tf_d.getText());
3     sumpoly=new float[baris_lt];
4     temp_polynomial1=new float [baris_lt+1][baris_lt+1];
5     matrik_polynomial1=new double[baris_lt][baris_lt];
6     for(i=0;i<baris_lt;i++){
7         for(j=0;j<baris_lt;j++){
8             x=0;
9             for(int k=1;k<kolom_lt;k++){
10                x=x+(normxbobotLatih[i][k]*normxbobotLatih[j][k]);
11            }//for k
12            sumpoly[j]=x;
13        }//for j
14        for(int l=0;l<baris_lt;l++){
15            temp_polynomial1[l][i]=sumpoly[l];
16            w=(double)temp_polynomial1[l][i];
17            temp_powl=Math.pow(w, degree);
18            matrik_polynomial1[i][l]=temp_powl;
19        }//for l
20    }//fori
21 }//end normalisasi

```

Kode Program 4.3 Proses Perhitungan Kernel Polynomial

Berikut penjelasan Kode Program 4.3.

1. Baris 1-5 merupakan inisialisasi variabel
2. Baris 6-21 merupakan proses perhitungan *kernel Polynomial*

b. Algoritma Perhitungan Matriks Hessian

Pada Kode Program 4.4 dijelaskan mengenai implementasi algoritma proses perhitungan *Matriks Hessian* dimana variabel dari perhitungan *matriks hessian* digunakan dalam proses *sequential training SVM*.

```

1 public void hitungMatrikHessian() {
2     double power=2;
3     lambda=Float.parseFloat(tf_lambda.getText());
4     konstanta=Float.parseFloat(tf_konstanta.getText());
5     tempy=new float[baris_lt];
6     int ambilIndex=tblDataLatih.getColumnCount()-1;
7     for(i=0;i<baris_lt;i++){

```



```

7      tempy[i]=Float.parseFloat(tblDataLatih.getValueAt(
8          i, ambilIndex).toString());
9      }//fori
10     templamdasvm=(double) lambda;
11     templamdasvm=Math.pow(templamdasvm,power);
12     templamdabarusvm=(float) templamdasvm;
13     matrik_hessian1=new float[baris_lt][baris_lt];
14     for(i=0;i<baris_lt;i++){
15         for(int k=0;k<baris_lt;k++){
16             matrik_hessian1[i][k]=tempy[k]*tempy[k];
17             tempPolynomial11=(float)matrik_polynomial11[i]
18                 [k];
19             matrik_hessian1[i][k]=(tempy[i]*tempy[k])*(t
20                 empPolynomial11+templamdabarusvm);
21         }//forkey
22     }//fori
23 }//end hessian

```

Kode Program 4. 4 Proses Perhitungan Matriks Hessian

Berikut penjelasan Kode Program 4.4.

1. Baris 2-6 dan 10-12 inialisasi
2. Baris 7-9 proses pengambilan kelas data
3. Baris 14-22 proses perhitungan Matriks Hessian

c. Implementasi Algoritm Perhitungan terhadap nilai E_i

Proses perhitungan E_i didapatkan dengan melakukan penjumlahan hasil perkalian dari matrik hessian dengan nilai alpha ke- j . Untuk kode program perhitungan matrik E_i seperti yg dijelaskan pada Kode Program 4.5.

```

1      public void iterasiLevel1(){
2          iterasimax=Integer.parseInt(tf_iterasi1.getText());
3          Ce=Float.parseFloat(tf_nilaic.getText());
4          alphaawal=Float.parseFloat(tf_alphaawal.getText());
5          epsilon=Float.parseFloat(tf_epsilon.getText());
6          tresholdsv=Float.parseFloat(tf_tresholdsv.getText());
7          alpha_hasil=new float[baris_lt+2][baris_lt+2];
8          nilaialpha=new float[baris_lt];
9          Eil=new float[baris_lt];
10         deltaAlpha=new float[baris_lt];
11         alpha=new float[baris_lt];
12         for(int al=0;al<baris_lt;al++){
13             alpha[al]=alphaawal; }

```

```

13     int iterasi,iterasiStop;
14     for( iterasi=0;iterasi<iterasimax;iterasi++){
15         System.out.println("iterasi ke : "+iterasi);
16         for(i=0;i<baris_lt;i++){
17             float temp_alpha=0;
18             for(int k=0;k<baris_lt;k++){
19                 alpha_hasil[i][k]=matrik_hessian1[i][k]*alpha[k];
20                 temp_alpha+=alpha_hasil[i][k];
21             }
22             Eil[i]=temp_alpha;
23             temp_deltaalpha=Math.min((Math.max(gamma*(1-Eil[i]), -
24             1*alpha[i])), Ce-alpha[i]);
25             deltaAlpha[i]=temp_deltaalpha;
26             maxDeltaAlpha=deltaAlpha[0];
27         }//fori
28         for(int a=0;a<baris_lt;a++){
29             alpha[a]=alpha[a]+deltaAlpha[a];
30         }//fora
31         for(i=0;i<baris_lt;i++){
32             if(deltaAlpha[i]>maxDeltaAlpha){
33                 maxDeltaAlpha=deltaAlpha[i];
34             }//if
35         }//foricek da
36         if(maxDeltaAlpha<epsilon){
37             break;
38         }
39     }//iter
40 }//end iterasi level 1

```

Kode Program 4.5 Perhitungan E_i

Berikut penjelasan Kode Program 4.5.

1. Baris 2-5 proses pengambilan parameter yang diberi oleh pengguna
2. 6-10 merupakan proses inialisasi parameter
3. baris 11-12 merupakan proses inialisasi alpha awal
4. baris 13 merupakan awal dari proses iterasi
5. baris 18-39 merupakan proses perhitungan E_i

d. Implementasi Algoritma Perhitungan Nilai Delta Alpha

Nilai delta alpha didapatkan dari perkalian nilai gamma dengan hasil dari 1 dikurangi dengan nilai E_i yang ke- i lalu selanjutnya dibandingkan dengan nilai

alpha ke- i untuk dicari nilai paling maksimum. Hasil dari nilai maksimum akan dibandingkan dengan hasil dari variabel C yang dikurangi dengan nilai alpha ke- i . Untuk Kode Program Perhitungan Nilai Delta Alpha seperti pada Kode Program 4.6.

```

1   temp_deltaalpha=Math.min((Math.max(gamma*(1-Ei1[i]),
2   -1*alpha[i])), Ce-alpha[i]);
3   deltaAlpha[i]=temp_deltaalpha;
4   maxDeltaAlpha=deltaAlpha[0];

```

Kode Program 4. 6 Perhitungan Nilai Delta Alpha

Berikut penjelasan Kode Program 4.6.

1. Baris 1-2 merupakan proses perhitungan nilai delta alpha

e. Implementasi Algoritma Perhitungan Nilai Alpha Baru

Nilai alpha baru didapatkan dengan menambahkan nilai delta alpha dengan alpha awal yang sudah ditetapkan sebelumnya. Untuk algoritma perhitungan nilai alpha baru seperti yang terlihat pada Kode Program 4.7.

```

1   for(int a=0;a<baris_lt;a++){
2       alpha[a]=alpha[a]+deltaAlpha[a];
3   }//fora

```

Kode Program 4.7 Perhitungan Nilai Alpha Baru

Berikut penjelasan Kode Program 4.7.

1. Baris 1-3 merupakan proses penghitungan nilai alpha baru.

f. Implementasi Algoritma Perhitungan Nilai WX^+ dan WX^-

Proses perhitungan nilai WX^+ dan WX^- diawali dengan mencari *training* yang memiliki alpha tertinggi pada kelas positif akan ditetapkan sebagai (WX^+) dan data *training* yang memiliki nilai alpha tertinggi pada kelas negataif (WX^-). Algoritma perhitungan nilai WX^+ dan WX^- seperti pada Kode Program 4.8.

```

1   public void hitungwxpositif(){
2       wxpositif1=new float[baris_lt];
3       for(i=0;i<baris_lt;i++){
4           wxpositif1[i]=matrik_kernelxpos1[i]*tempy[i]*alpha[i];
5           temp_totwxpos1+=wxpositif1[i];
6       }//fori
7   }//endhtg wxpositif
8   public void hitungwxnegatif(){
9       wxnegatif1=new float[baris_lt];
10      for(i=0;i<baris_lt;i++){
11          wxnegatif1[i]=matrik_kernelxneg1[i]*tempy[i]*alpha[i];
              temp_totwxneg1+=wxnegatif1[i];

```

12	} //end for i
13	} //end hitung wxnegatif

Kode Program 4.8 Proses Perhitungan (WX^+) dan (WX^-)

Berikut penjelasan Kode Program 4.8.

1. Baris 2 merupakan proses inialisasi WX^+ .
2. Baris 3-4 merupakan proses penghitungan WX^+
3. Baris 5 merupakan proses penjumlahan total nilai WX^+
4. baris 9 merupakan inialisasi WX^-
5. Baris 10 merupakan proses penghitungan WX^-
6. Baris 11 merupakan proses penjumlahan total nilai WX^-

g. Implementasi Algoritma Perhitungan Nilai Bias

Nilai bias didapatkan dari menjumlahkan total nilai WX^+ dan WX^- lalu mengalikan dengan $-\frac{1}{2}$ seperti pada Kode Program 4.9.

1	public void hitungBias() {
2	bias = -(temp_totwxpos1 + temp_totwxneg1) / 2;
3	} //end hitung bias

Kode Program 4.9 Proses Perhitungan Nilai Bias

Berikut Penjelasan Kode Program 4.9

1. Baris 2 merupakan proses penghitungan nilai bias.

4.3.4 Implementasi Algoritma Perhitungan Nilai $f(x)$

Nilai $f(x)$ pada tahap pengujian didapatkan dengan menggunakan data *testing* dengan parameter nilai bias. Hasil dari perhitungan nilai $f(x)$ akan bernilai positif dan negatif. Jika bernilai positif maka data yang diuji akan masuk ke dalam kelas A sedangkan yang bernilai negatif akan dilanjutkan untuk proses penghitungan di level 2 untuk menentukan data tersebut masuk ke dalam kelas B atau C. Proses perhitungan nilai $f(x)$ seperti yang terlihat pada Kode Program 4.10.

1	Public void testingLevel() {
2	float abs_nilaifx;
3	temp_aiyik = new float[baris_lt];
4	temp_kxtesting = new float[baris_lt];
5	temp_ndatatesting = new float[baris_lt + baris_tt];
6	matrik_kxtesting = new float[baris_lt];
7	temp_nilfx = new float[baris_lt + 20];
8	for (int ntes = 0; ntes < baris_tt; ntes++) {

```

9      float temp_ntes;
10     for(i=0;i<baris_lt;i++){
11         temp_ntes=0;
12         for(j=1;j<kolom_lt;j++){
13             temp_ndatatesting[ntes]=Float.parseFloat(tblNormxBobotTe
14             sting.getValueAt(ntes, j).toString());
15             temp_ntes=temp_ntes+(temp_ndatatesting[ntes]*normxbobotL
16             atih[i][j]);
17             temp_kxtesting[i]=temp_ntes;
18             }//je
19             dbl_kxtesting=(double)temp_kxtesting[i];
20             dbl_tempkxtesting=Math.pow(dbl_kxtesting,degree);
21             matrik_kxtesting[i]=(float)dbl_tempkxtesting;
22             }//ii
23             float tempaiyi=0;
24             for(i=0;i<baris_lt;i++){
25                 temp_aiyik[i]=alpha[i]*matrik_kxtesting[i]*tempy[i];
26                 tempaiyi+=temp_aiyik[i];
27             }//ii
28             temp_nilafx=tempaiyi+bias;
29             abs_nilafx=Math.signum(temp_nilafx);
30             temp_nilfx[ntes]=abs_nilafx;
31             }//for banyak testing
32         }//end testing level 1

```

Kode Program 4. 10 Proses Perhitungan Nilai $f(x)$

Berikut penjelasan Kode Program 4.10 tentang proses perhitungan nilai $f(x)$:

1. Baris 2 hingga baris 7 merupakan inisialisasi variabel.
2. Baris 8 merupakan awal dari perulangan penghitungan nilai $f(x)$ sebanyak data *testing* yang digunakan.
3. Baris 24 hingga 25 merupakan proses perhitungan $\alpha_i y_i K(x_i, x)$.
4. Baris 27 merupakan proses perhitungan nilai $f(x)$.
5. Baris 27 merupakan proses *signum* dari nilai $f(x)$.

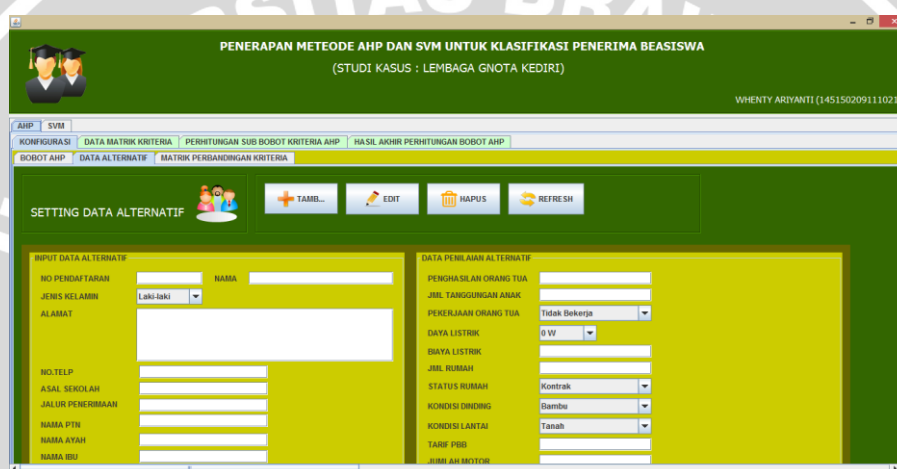
4.4 Implementasi Antarmuka

Terdapat beberapa bagian dalam implementasi antarmuka yakni antarmuka konfigurasi data alternatif, antarmuka konfigurasi matriks perbandingan kriteria, antarmuka *load* matrik kriteria, antarmuka perhitungan sub bobot kriteria, antarmuka perhitungan hasil akhir AHP, antarmuka *load* data SVM, antarmuka normalisasi *dataset*, antarmuka perkalian bobot setelah

normalisasi, antarmuka proses *sequential training*, antarmuka perhitungan iterasi dan antarmuka perhitungan nilai $f(x)$.

4.4.1 Antarmuka Konfigurasi Data Alternatif

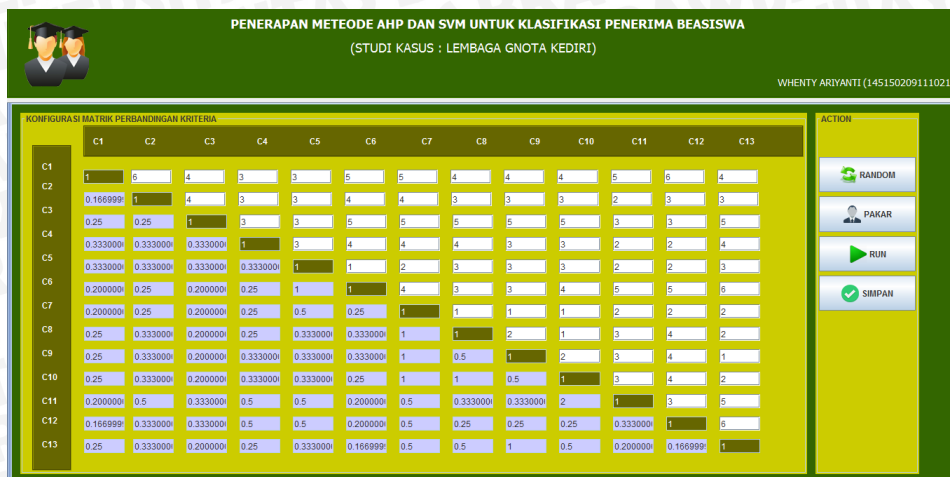
Antarmuka konfigurasi data alternatif merupakan suatu *interface* yang bertujuan untuk melakukan pengisian data alternatif beserta data penilaian alternatif yang nantinya secara otomatis akan berpengaruh pada nilai pada setiap alternatif di dalam tabel dataset. Terdapat beberapa tombol yang bisa digunakan dalam antarmuka konfigurasi data alternatif meliputi tombol tambah untuk menambahkan data alternatif, tombol *edit* untuk melakukan *edit* data alternatif, tombol hapus untuk menghapus data alternatif dan tombol refresh untuk melakukan *refresh* data alternatif. Implementasi antarmuka konfigurasi data alternatif seperti yang terlihat pada Gambar 4.2.



Gambar 4.2 Antarmuka Konfigurasi Data Alternatif

4.4.2 Antarmuka Konfigurasi Matriks Perbandingan

Antarmuka konfigurasi matrik perbandingan seperti yang terlihat pada Gambar 4.3 adalah untuk melakukan pengisian nilai bobot pada matriks perbandingan berpasangan untuk proses AHP. Terdapat empat tombol yakni *random*, *run* dan *simpan* dan *pakar*. Tombol *random* untuk mengisikan nilai matrik perbandingan secara random, tombol *run* untuk memproses perhitungan dan tombol *simpan* untuk menyimpan hasil dari proses perhitungan. Sedangkan tombol *pakar* digunakan untuk melakukan *reset* matriks perbandingan kriteria sesuai dengan nilai yang ditentukan oleh pakar.



Gambar 4.3 Antarmuka Konfigurasi Matriks Perbandingan

4.4.3 Antarmuka Load Data Matriks Kriteria

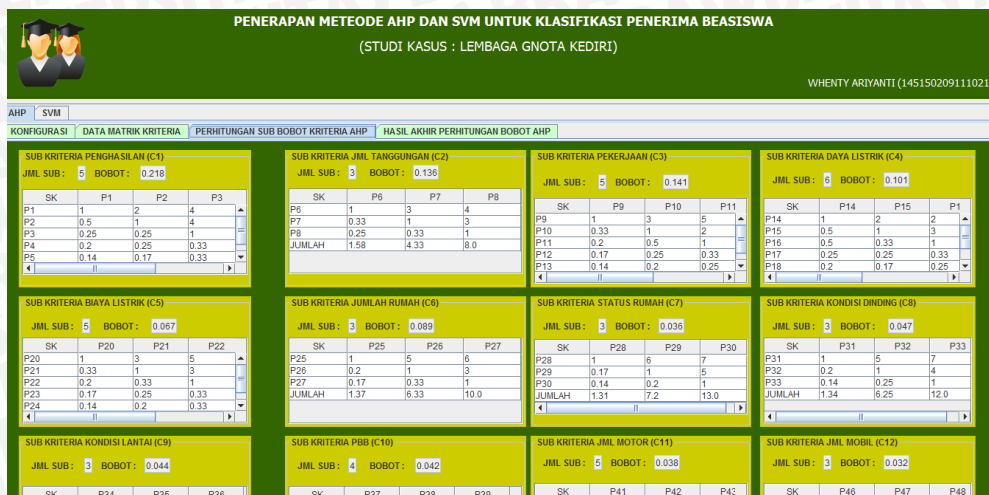
Antarmuka load data matriks kriteria digunakan untuk melakukan *load* matriks kriteria berpasangan beserta melakukan perhitungan untuk konsistensi dan sub bobot kriteria dengan metode AHP. Pada Gambar 4.4 selain menampilkan matriks kriteria berpasangan, juga menampilkan hasil dari normalisasi matriks kriteria berpasangan, nilai lamda maksimum, banyak kriteria yang digunakan, nilai *CI* dan *CR*, nilai *RI* beserta status konsistensinya.



Gambar 4.4 Antarmuka Load Data Matriks Kriteria

4.4.4 Antarmuka Perhitungan Sub Bobot Kriteria AHP

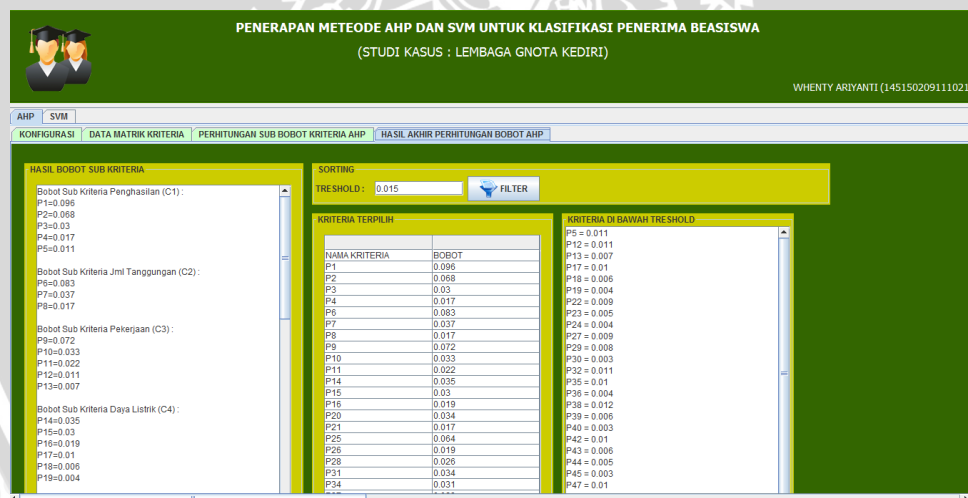
Antarmuka proses perhitungan sub bobot kriteria AHP seperti yang terlihat pada Gambar 4.5 merupakan hasil dari matriks kriteria berpasangan untuk sub kriteria dan jumlah bobot pada kriteria tersebut. Matriks kriteria berpasangan untuk sub kriteria AHP bersifat statis karena data dari sub kriteria diperoleh dari hasil wawancara dengan pakar Lembaga GNOTA Kediri.



Gambar 4.5 Hasil Perhitungan Sub Bobot Kriteria AHP

4.4.5 Antarmuka Hasil Akhir Perhitungan AHP

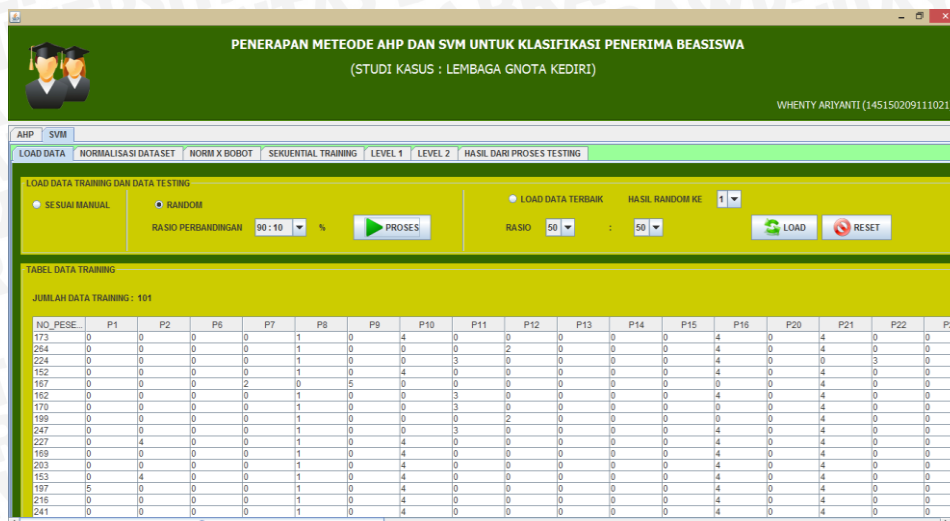
Antarmuka hasil akhir perhitungan AHP dapat dilihat pada Gambar 4.6 dimana pada Gambar 4.6 terdapat tombol *filter* yakni untuk menyaring kriteria yang nilainya berada di atas dari nilai *threshold* yang ditetapkan dimana sub kriteria yang nilainya berada di atas *threshold* akan diikuti ke dalam proses perhitungan dengan metode SVM.



Gambar 4.6 Hasil Akhir Perhitungan AHP

4.4.6 Antarmuka Load Data SVM

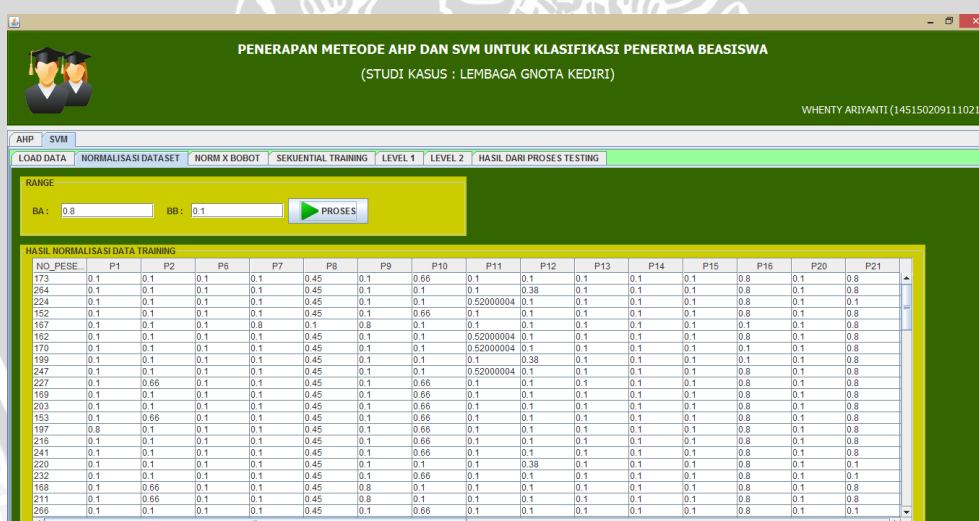
Pada menu load data SVM terdapat tiga pilihan untuk proses *load* data yakni sesuai manual, secara *random* dan *load* data terbaik hasil pengujian. Untuk *load* data secara *random* terdapat rasio perbandingan data untuk menentukan jumlah data latih dan data testing seperti pada Gambar 4.7. Sedangkan untuk *load* data terbaik didapatkan dari hasil *random* yang telah disimpan di dalam *database* untuk melakukan pengujian terhadap parameter SVM.



Gambar 4.7 Load Data SVM

4.4.7 Antarmuka Normalisasi Dataset SVM

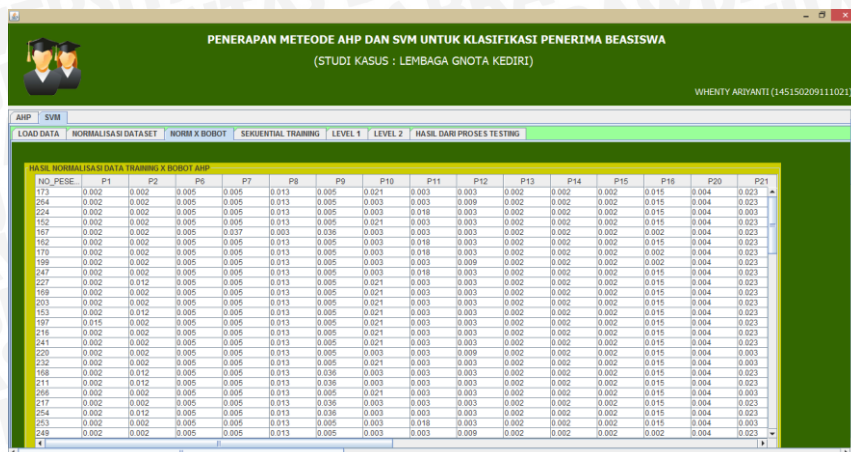
Antarmuka normalisasi dataset SVM seperti pada Gambar 4.8 adalah untuk melakukan normalisasi dari seluruh data *training* maupun data *testing*. Terdapat *field* inputan BA (batas atas) dan BB (batas bawah) untuk menentukan nilai batas atas dan batas bawah yang akan berpengaruh dalam proses normalisasi dataset.



Gambar 4.8 Normalisasi Dataset SVM

4.4.8 Antarmuka Perkalian Bobot Setelah Normalisasi

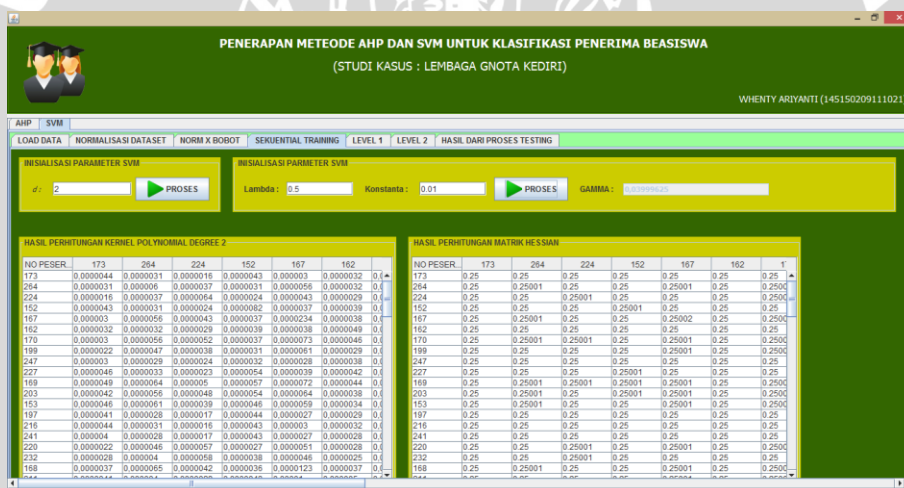
Antarmuka perkalian bobot setelah normalisasi akan dikalikan dengan hasil perhitungan bobot sub kriteria terpilih yang memiliki nilai di atas nilai *threshold* yang ditentukan seperti pada Gambar 4.9.



Gambar 4.9 Antarmuka Perkalian Bobot Setelah Normalisasi

4.4.9 Antarmuka Proses Sequential Training

Antarmuka proses *Sequential Training* seperti pada Gambar 4.10 terdapat beberapa *textfield* untuk mendapatkan parameter *inputan* yang akan digunakan yakni nilai λ (*lambda*), iterasi maksimum, C (*Complexity*), konstanta γ (*gamma*) seperti pada Gambar 4.10.

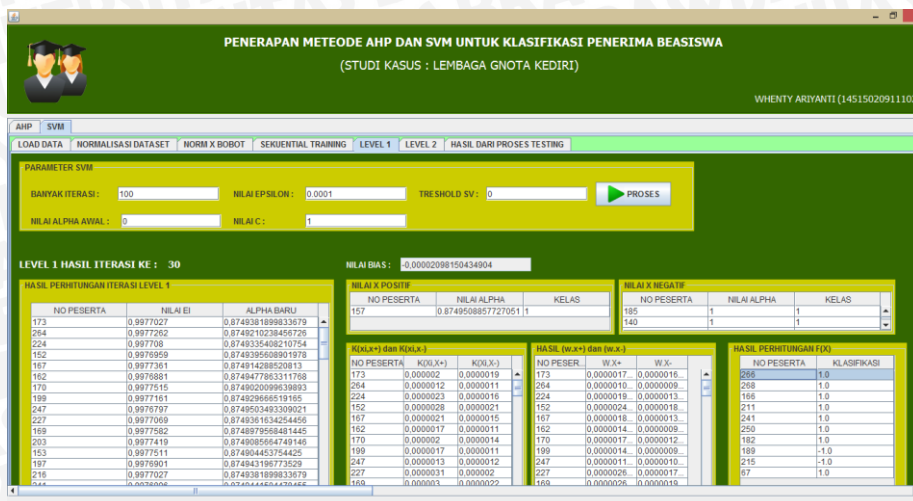


Gambar 4.10 Antarmuka Proses Sequential Training

Pada Gambar 4.10 merupakan proses *Sequential Training* dengan melakukan perhitungan *kernel Polynomial Degree* dan perhitungan *Matriks Hessian*.

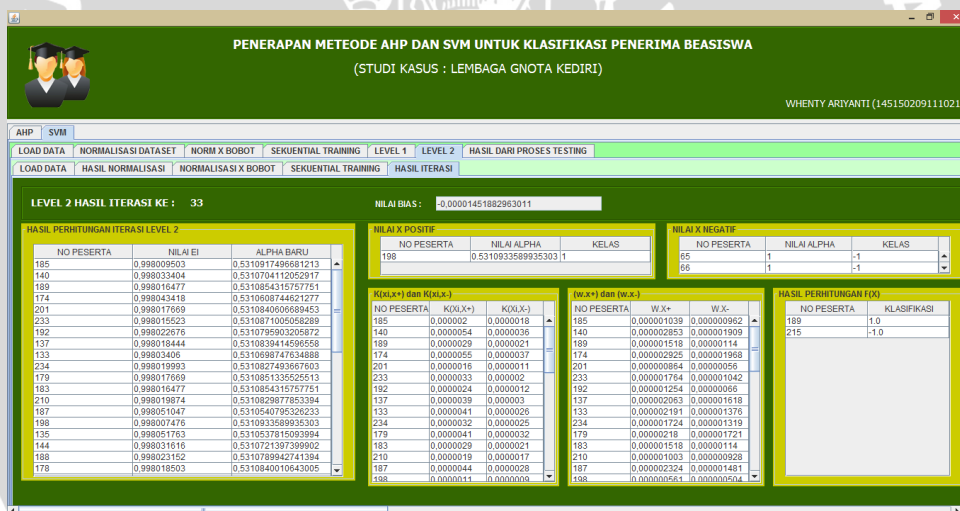
4.4.10 Antarmuka Perhitungan Iterasi

Gambar 4.11 merupakan antarmuka perhitungan hasil iterasi level 1 dimana akan didapatkan nilai E_i dan Alpha baru pada iterasi terakhir. Sedangkan untuk hasil iterasi level 2 seperti yang terlihat pada Gambar 4.12.



Gambar 4.11 Hasil Perhitungan Iterasi Level 1

Gambar 4.11 menjelaskan mengenai hasil perhitungan iterasi level 1 dengan metode SVM. Beberapa *field* tersedia untuk menerima masukan dari parameter SVM seperti berapa banyak iterasi yang akan dilakukan, besar nilai *epsilon*, nilai *Threshold SV*, nilai alpha awal dan nilai C (*Complexity*).



Gambar 4.12 Hasil Perhitungan Iterasi Level 2

Pada Gambar 4.12 merupakan hasil perhitungan SVM pada iterasi level 2. Pada menu tersebut akan menampilkan hasil klasifikasi sistem pada level 2 beserta nilai alpha baru yang didapatkan dari banyak iterasi yang dilakukan.

4.4.11 Antarmuka Perhitungan Nilai $f(x)$

Pada Gambar 4.12 merupakan hasil perhitungan nilai $f(x)$ untuk menentukan kelas SVM ke dalam bentuk kelas A jika sangat layak, kelas B jika layak dan kelas C jika tidak layak. Pada Gambar 4.12 terdapat *field* id penyimpanan untuk menyimpan hasil perhitungan sistem ke dalam *database*.

PENERAPAN METODE AHP DAN SVM UNTUK KLASIFIKASI PENERIMA BEASISWA
(STUDI KASUS : LEMBAGA GNOTA KEDIRI)

WHENTY ARIYANTI (145150209111021)

AHP | SVM

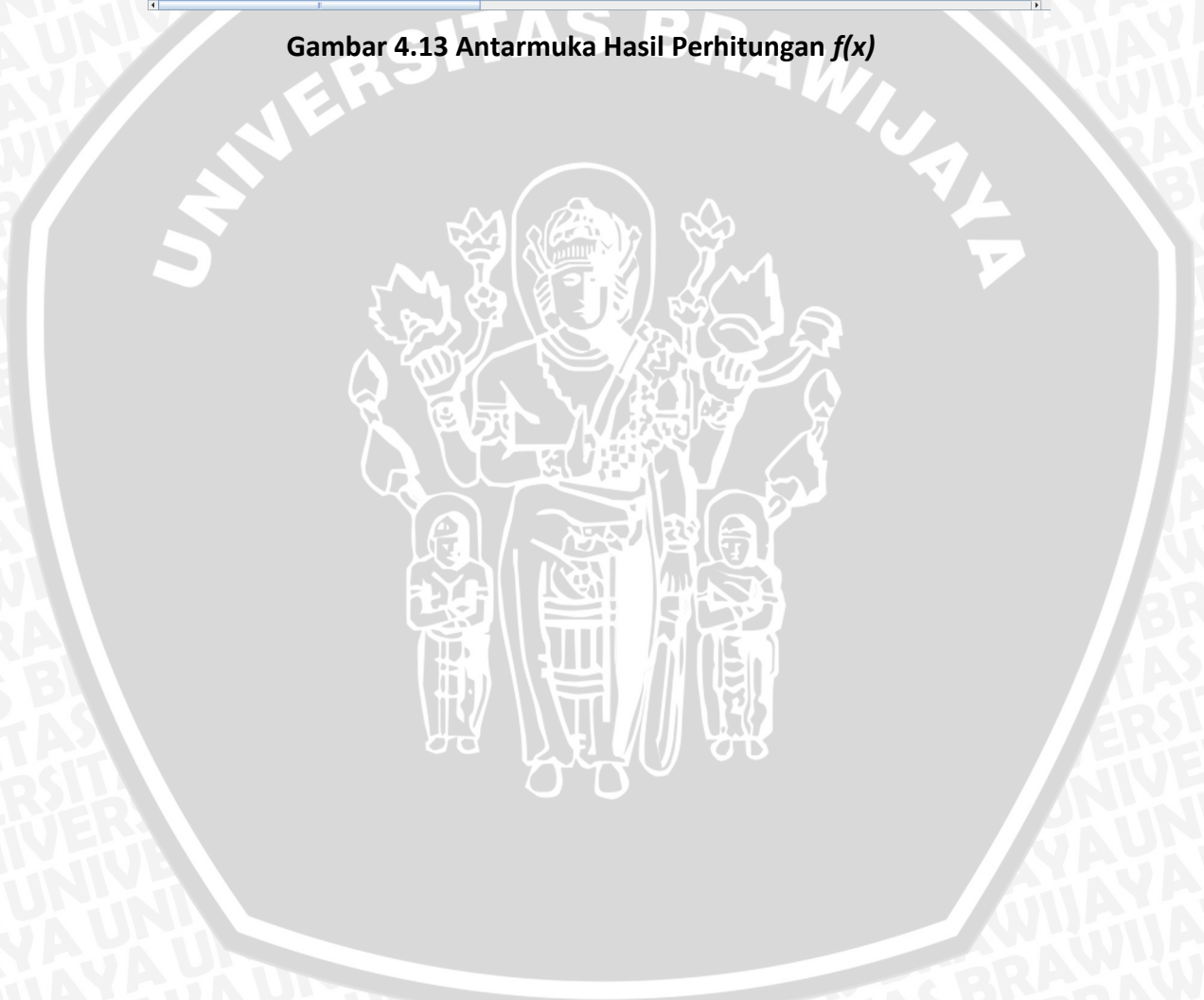
LOAD DATA | NORMALISASI DATASET | NORM X BOBOT | SEKUNJIAL TRAINING | LEVEL 1 | LEVEL 2 | HASIL DARI PROSES TESTING

AKURASI SISTEM: **AKURASI : 93.10345**
EKSEKUSI : 43,958706035 detik

SAVE HASIL TESTING
ID PENYIMPANAN: SAVE

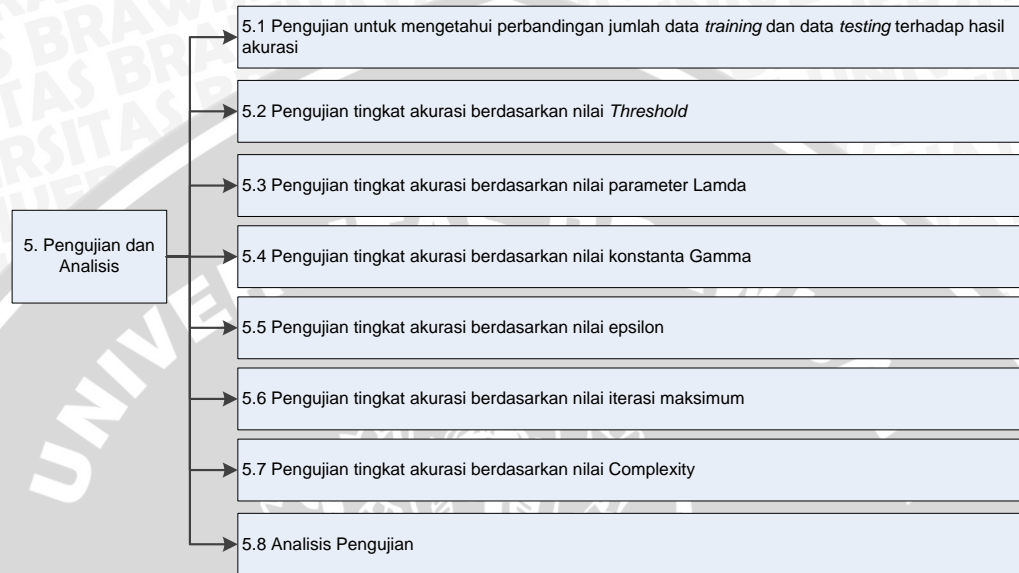
HASIL EVALUASI CONFUSION CLASS					MODEL EVALUASI CONFUSION CLASS				
NO PESERTA	PREDICTED CLASS	ACTUAL CLASS	CLASS ACTUAL	CLASS PREDICTED	ACTUAL	A	B	C	JUMLAH
211	1	1	A	A	A	8.0	0.0	0.0	8.0
187	1	1	A	A	B	0.0	0.0	1.0	1.0
135	1	1	A	A	C	0.0	0.0	1.0	1.0
271	1	1	A	A					
217	1	1	A	A					
232	1	1	A	A					
221	1	1	A	A					
196	1	1	A	A					
199	-1	-1	B	C					
195	-1	-1	C	C					

Gambar 4.13 Antarmuka Hasil Perhitungan $f(x)$



BAB 5 PENGUJIAN DAN ANALISIS

Bab ini membahas mengenai pengujian dan analisis terhadap sistem yang telah dibuat pada tahap implementasi. Pengujian dilakukan berdasarkan skenario yang telah ditentukan sebelumnya. Diagram proses pengujian seperti pada Gambar 5.1.



Gambar 5.1 Diagram Pengujian dan Analisis

Pada Gambar 5.1 menunjukkan diagram proses pengujian dan analisis yang terdiri dari delapan tipe pengujian yakni pengujian perbandingan jumlah data training dan data testing, pengujian tingkat akurasi berdasarkan nilai *threshold*, parameter lamda, konstanta gamma, epsilon, iterasi maksimum dan C (*Complexity*).

Berdasarkan skenario pengujian yang terdapat pada Bab 3, maka berikut adalah hasil pengujian yang didapatkan :

5.1 Pengujian Terhadap Rasio Perbandingan Data *Training* dan Data *Testing* Terhadap Akurasi

Pengujian terhadap rasio perbandingan data training dan data *testing* bertujuan untuk mendapatkan hasil akurasi terbaik yang akan dijelaskan pada skenario pengujian dan analisa skenario pengujian.

5.1.1 Skenario Pengujian Rasio Perbandingan

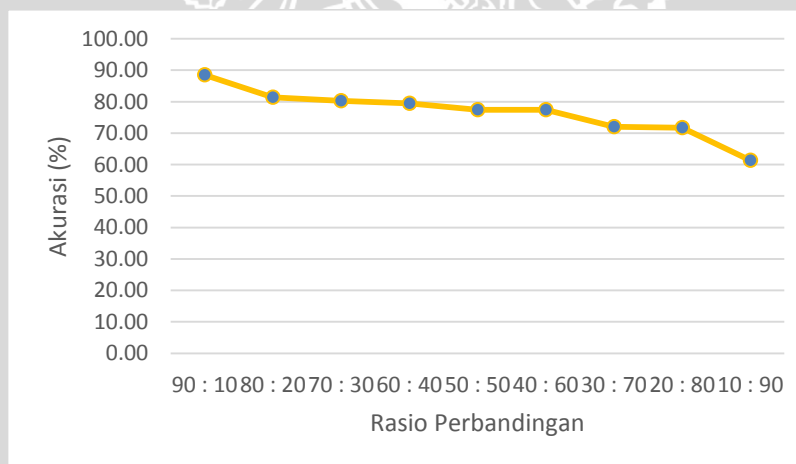
Pengujian berdasarkan rasio data *training* dan data *testing* terbaik terhadap akurasi diambil secara *random* dari dataset sejumlah 111 data dengan rasio perbandingan 90%:10%, 80%:20%, 70%:30%, 60%:40%, 50%:50%, 40%:60%, 30%:70%, 20%:80% dan 10%:90%. Nilai dari parameter *Sequential Training* pada pengujian ini adalah $\lambda = 0.5$, $\gamma = 0.01$, $\epsilon = 0.0001$,

iterasi maksimum=100, C=1 dan nilai $d = 2$. Hasil dari pengujian terhadap rasio perbandingan data *training* dan data *testing* seperti pada Tabel 5.1.

Tabel 5.1 Hasil Pengujian Rasio Data *Training* dan Data *Testing*

Rasio	Percobaan ke- <i>i</i>					Rata-rata
	1	2	3	4	5	
90 : 10	93.10	77.77	93.10	93.10	85.71	88.56
80 : 20	80.00	80.00	76.27	87.09	83.60	81.39
70 : 30	81.81	74.11	76.74	84.26	84.26	80.24
60 : 40	78.15	74.35	90.47	72.41	81.81	79.44
50 : 50	76.55	79.59	73.42	75.00	82.55	77.42
40 : 60	72.88	77.90	76.66	79.12	80.32	77.38
30 : 70	67.83	65.48	80.00	74.63	72.41	72.07
20 : 80	69.82	69.82	67.82	79.33	71.79	71.72
10 : 90	60.48	68.87	63.34	60.48	53.52	61.34

5.1.2 Analisis Pengujian Rasio Perbandingan



Gambar 5.2 Grafik Tingkat Akurasi Hasil Pengujian Rasio Perbandingan

Pada Gambar 5.2 merupakan grafik akurasi rasio perbandingan dimana diketahui bahwa rata-rata tingkat akurasi tertinggi sebesar 88.56% dan akurasi terbaik pada 93.10% yang menggunakan rasio perbandingan 90%:10% dimana pada pengujian selanjutnya akan menggunakan rasio 90%:10%. Pemilihan data *training* dan data *testing* dilakukan secara *random*, dimana hasil *random* dengan akurasi terbaik akan disimpan untuk dibandingkan dengan hasil *random* lainnya yang akan digunakan untuk proses pengujian selanjutnya. Pada Gambar 5.2 menunjukkan bahwa grafik hasil pengujian terhadap rasio perbandingan meningkat ketika nilai perbandingan antara data *training* dan data *testing* semakin tinggi karena dengan banyaknya data

training yang digunakan maka sistem akan semakin cerdas dimana sistem dapat mengenali lebih banyak pola data yang terbentuk sehingga dapat menghasilkan akurasi yang tinggi. Begitupula sebaliknya, jika perbandingan nilai data *training* dan data *testing* semakin rendah maka akurasi yang dihasilkan juga menurun.

5.2 Pengujian Nilai *Threshold* AHP

Pengujian nilai *threshold* AHP dilakukan untuk mendapatkan nilai akurasi terbaik berdasarkan sub kriteria tertentu yang digunakan.

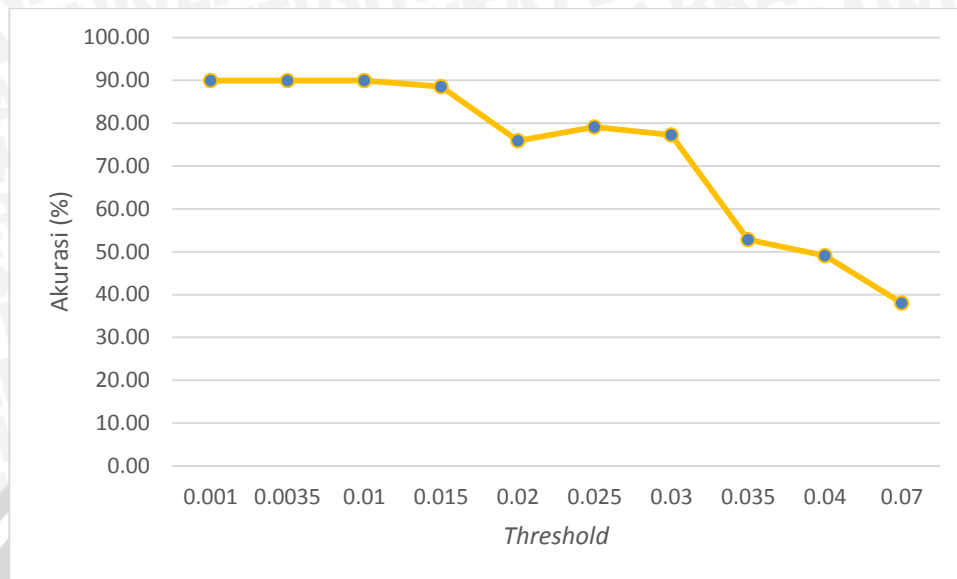
5.2.1 Skenario Pengujian Nilai *Threshold* AHP

Tabel 5.3 menunjukkan pengujian terhadap nilai *threshold* AHP yang dihasilkan dengan menggunakan parameter *Sequential Training* pada pengujian ini adalah $\lambda = 0.5$, $\gamma = 0.01$, $\varepsilon = 0.0001$, iterasi maksimum=100, $C=1$ dan nilai $d = 2$. Beberapa nilai *threshold* yang digunakan meliputi 0.001, 0.0035, 0.01, 0.015, 0.02, 0.025, 0.03, 0.035, 0.04 dan 0.07. Hasil dari pengujian terhadap nilai *threshold* AHP ditunjukkan pada Tabel 5.2.

Tabel 5.2 Hasil Pengujian Nilai *Threshold* AHP

Threhsold	Banyak Fitur	Percobaan ke- <i>i</i>					Akurasi	Waktu
		1	2	3	4	5		
0.001	50	93.10	77.77	93.10	100.00	85.71	89.94	23.89
0.0035	47	93.10	77.77	93.10	100.00	85.71	89.94	23.84
0.01	33	93.10	77.77	93.10	100.00	85.71	89.94	23.55
0.015	23	93.10	77.77	93.10	93.10	85.71	88.56	23.33
0.02	16	85.71	69.23	77.77	77.78	69.23	75.94	23.18
0.025	14	93.10	77.77	77.77	69.23	77.77	79.13	23.16
0.03	13	93.10	77.77	77.77	60.00	77.77	77.28	23.19
0.035	7	77.77	39.13	60.00	27.27	60.00	52.83	23.16
0.04	5	69.23	39.13	60.00	27.27	50.00	49.13	23.19
0.07	3	69.23	27.27	27.27	27.27	39.13	38.03	22.35

5.2.2 Analisis Pengujian Nilai Threshold AHP



Gambar 5.3 Grafik Pengujian Nilai Threshold

Gambar 5.3 menunjukkan grafik hasil pengujian terhadap nilai *threshold* dimana rata-rata akurasi terbaik sebesar 89.94% dengan menggunakan nilai *threshold* 0.001, 0.0035, dan 0.01. Pada proses pengujian selanjutnya akan digunakan nilai *threshold* 0.01 dengan 33 fitur terpilih karena terlihat bahwa pada nilai *threshold* 0.01 memiliki rata-rata waktu eksekusi tercepat sebesar 23.55 detik, sedangkan nilai *threshold* 0.001 membutuhkan waktu 23.89 detik dan nilai *threshold* 0.0035 membutuhkan waktu 23.84 detik. Semakin besar nilai *threshold* yang digunakan maka akan semakin sedikit fitur yang terpilih sehingga mengakibatkan menurunnya tingkat akurasi yang dihasilkan oleh sistem. Nilai *threshold* AHP adalah batasan nilai bobot kriteria yang akan digunakan dalam proses pengujian. Hasil dari pengujian berdasarkan nilai *threshold* secara keseluruhan dapat dilihat pada Lampiran R.

5.3 Pengujian Terhadap Tingkat Akurasi dari Pengaruh Parameter Lamda

Pengujian terhadap tingkat akurasi dari pengaruh parameter nilai lamda berupa skenario dan analisa.

5.3.1 Skenario Pengujian Lamda

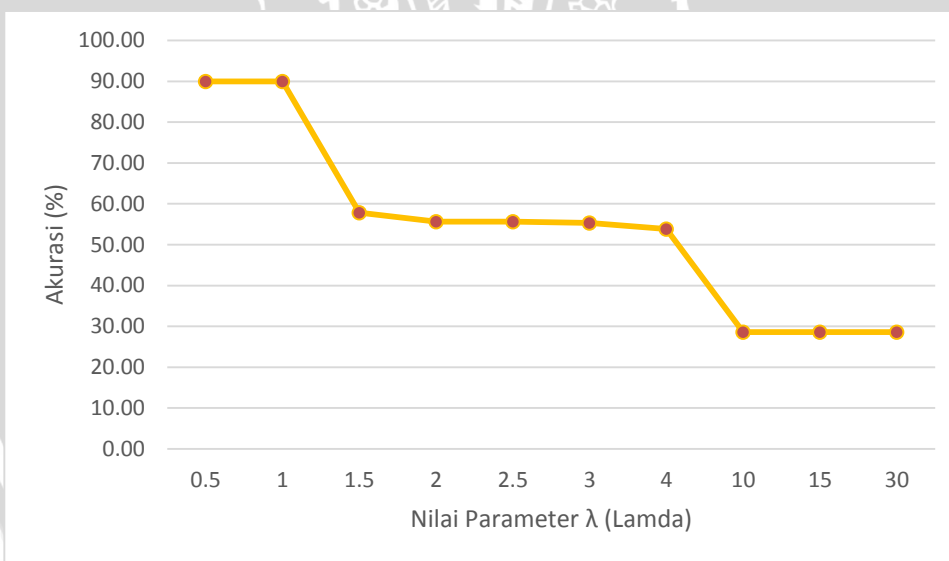
Pengujian nilai lamda dilakukan untuk mendapatkan nilai terbaik untuk lamda seperti pada Tabel 5.3 dengan nilai parameter *Sequential Training SVM* yang digunakan adalah $\gamma = 0.01$, $\epsilon = 0.0001$, iterasi Maksimum = 100, $C = 1$, rasio perbandingan 90%:10% dan nilai $d=2$. Beberapa nilai λ (*Lamda*) yang digunakan meliputi 0.5, 1, 1.5, 2, 2.5, 3, 4, 10, 15 dan 30.



Tabel 5.3 Hasil Pengujian Nilai Lamda

Lamda	Percobaan ke- <i>i</i>					Rata-rata
	1	2	3	4	5	
0.5	93.10	77.77	93.10	100.00	85.71	89.94
1	93.10	77.77	100.00	93.10	85.71	89.94
1.5	60.00	50.00	69.23	50.00	60.00	57.85
2	60.00	39.13	69.23	50.00	60.00	55.67
2.5	60.00	39.13	69.23	50.00	60.00	55.67
3	60.00	39.13	69.23	39.13	69.23	55.34
4	60.00	39.13	60.00	50.00	60.00	53.83
10	60.00	14.28	27.27	14.28	27.27	28.62
15	60.00	14.28	27.27	14.28	27.27	28.62
30	60.00	14.28	27.27	14.28	27.27	28.62

5.3.2 Analisis Pengujian Nilai Lamda



Gambar 5.4 Grafik Akurasi Terhadap Nilai Lamda

Gambar 5.4 merupakan grafik hasil dari pengujian nilai λ (Lamda) dimana diketahui akurasi tertinggi sebesar 89.94% pada nilai lamda 0.5 dan 1, dimana semakin tinggi nilai dari parameter λ (Lamda) maka cenderung semakin menurun nilai akurasi yang dihasilkan karena nilai dari parameter λ (Lamda) berpengaruh terhadap perhitungan komputasi pada matriks *Hessian* dimana hasil dari perhitungan matriks *Hessian* akan mempengaruhi nilai dari perhitungan $E_i, \delta\alpha$ dan nilai α (*alpha*) baru yang akan mempengaruhi iterasi selanjutnya. Kecenderungan proses komputasi yang lama disebabkan oleh *Augmented factor* pada nilai λ (lamda) sehingga proses perhitungan pada



sistem sangat lambat untuk memaksimalkan nilai margin dan akan terjadi ketidakstabilan proses *learning*.

5.4 Pengujian Terhadap Konstanta Gamma

Pengujian terhadap konstanta gamma untuk mengetahui nilai gamma terbaik yang akan berpengaruh terhadap akurasi. Pengujian terhadap konstanta gamma akan dijelaskan pada skenario pengujian dan analisa pengujian.

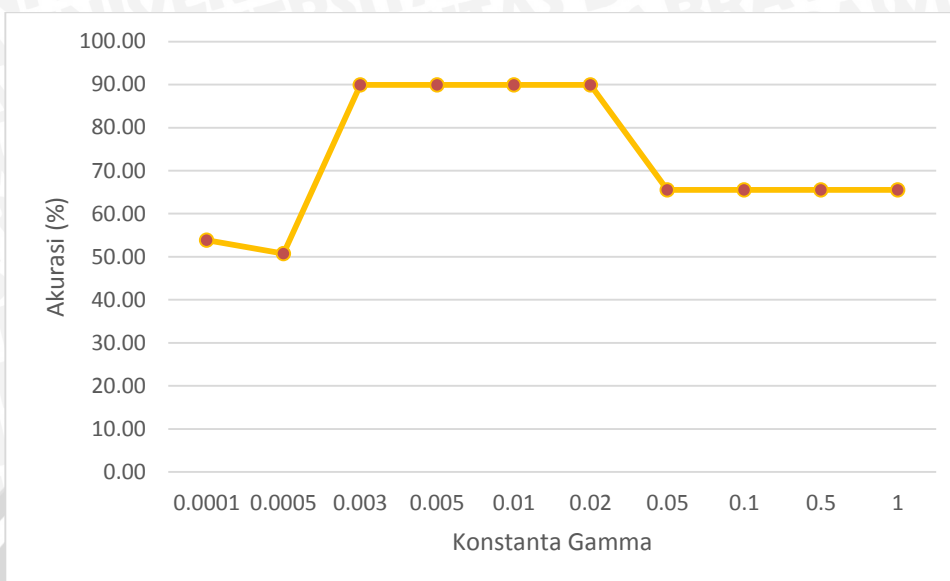
5.4.1 Skenario Pengujian Konstanta Gamma

Tabel 5.4 merupakan hasil pengujian terhadap nilai konstanta gamma dimana *gamma* merupakan parameter dari *Sequential Training SVM*. Nilai parameter yang digunakan adalah rasio perbandingan 90%:10%, nilai *threshold* 0.01, $\lambda = 0.5$, $\varepsilon = 0.0001$, iterasi maksimum=100, C=1 dan nilai $d = 2$.

Tabel 5.4 Hasil Pengujian Nilai Konstanta Gamma

Konstanta Gamma	Percobaan ke- <i>i</i>					Rata-rata
	1	2	3	4	5	
0.0001	60.00	39.13	60.00	50.00	60.00	53.83
0.0005	60.00	14.28	60.00	50.00	69.23	50.70
0.003	93.10	77.77	93.10	100.00	85.71	89.94
0.005	93.10	77.77	93.10	100.00	85.71	89.94
0.01	93.10	77.77	93.10	100.00	85.71	89.94
0.02	93.10	77.77	93.10	100.00	85.71	89.94
0.05	60.00	60.00	69.23	69.23	69.23	65.54
0.1	60.00	60.00	69.23	69.23	69.23	65.54
0.5	60.00	60.00	69.23	69.23	69.23	65.54
1	60.00	60.00	69.23	69.23	69.23	65.54

5.4.2 Analisis Pengujian Konstanta Gamma



Gambar 5.5 Grafik Akurasi Terhadap Nilai Konstanta Gamma

Gambar 5.5 merupakan grafik hasil pengujian dari konstanta *gamma* dimana diketahui akurasi tertinggi untuk pengujian konstanta gamma adalah sebesar 89.94% dengan nilai gamma 0.003, 0.005 dan 0.01. nilai dari konstanta gamma berpengaruh terhadap iterasi yang terjadi. Sehingga hasil pengujian pada nilai konstanta gamma 0.0001 dan 0.0005 cenderung tidak stabil karena semakin kecil nilai konstanta gamma maka iterasi berhenti mendekati iterasi maksimum (itermax). Jika semakin besar konstanta gamma maka semakin besar nilai *learning rate*. *Learning rate* adalah laju pembelajaran dimana semakin besar nilai *learning rate* maka proses pembelajaran akan semakin cepat. Namun apabila nilai *learning rate* relatif terlalu besar maka proses *training* dapat melampaui keadaan optimal yaitu pada nilai *error* paling minimal. Dengan kata lain, *learning rate* mempengaruhi ketelitian suatu sistem. Semakin besar nilai *learning rate*, maka ketelitian sistem akan semakin berkurang. Sebaliknya jika *learning rate* semakin kecil maka ketelitian suatu sistem akan semakin besar tetapi proses pembelajaran membutuhkan waktu yang lama.

5.5 Pengujian Terhadap Nilai Epsilon

Pengujian terhadap nilai epsilon yang merupakan parameter *Sequential Training SVM* akan dijelaskan pada skenario pengujian dan analisa pengujian.

5.5.1 Skenario Pengujian Nilai Epsilon

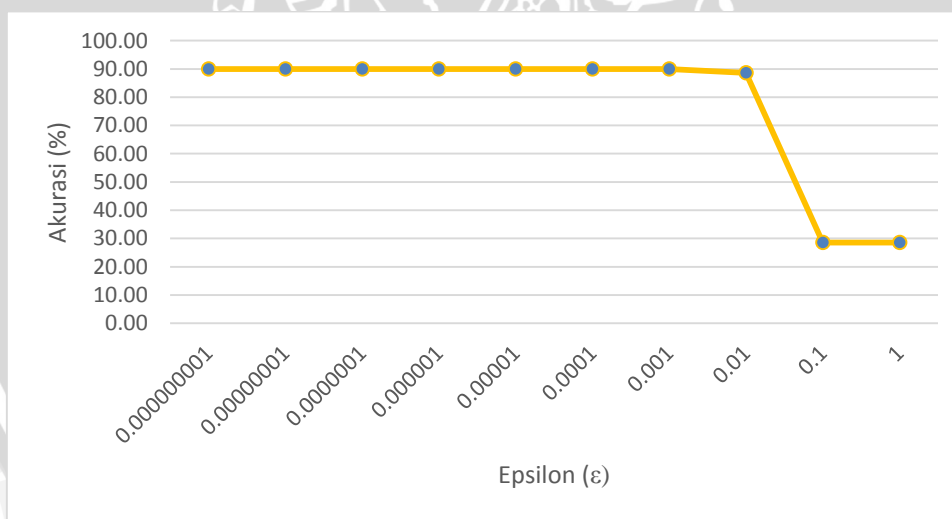
Tabel 5.5 menunjukkan hasil pengujian terhadap nilai epsilon dimana parameter yang digunakan adalah rasio 90%10%, nilai *threshold* 0.01, nilai dari $\lambda = 0.5$, konstanta γ (*gamma*) = 0.003, iterasi maksimum = 100, dan $C = 1$. Beberapa nilai ϵ (*epsilon*) yang digunakan dalam pengujian ini yaitu

0.000000001, 0.00000001, 0.0000001, 0.000001, 0.00001, 0.0001, 0.001, 0.01, 0.1 dan 1.

Tabel 5.5 Hasil Pengujian Nilai Epsilon

Nilai Epsilon	Percobaan ke- <i>i</i>					Rata-rata
	1	2	3	4	5	
0.000000001	93.10	77.77	93.10	100.00	85.71	89.94
0.00000001	93.10	77.77	93.10	100.00	85.71	89.94
0.0000001	93.10	77.77	93.10	100.00	85.71	89.94
0.000001	93.10	77.77	93.10	100.00	85.71	89.94
0.00001	93.10	77.77	93.10	100.00	85.71	89.94
0.0001	93.10	77.77	93.10	100.00	85.71	89.94
0.001	93.10	77.77	93.10	100.00	85.71	89.94
0.01	93.10	77.77	93.10	93.10	85.71	88.56
0.1	60.00	14.28	27.27	14.28	27.27	28.62
1	60.00	14.28	27.27	14.28	27.27	28.62

5.5.2 Analisis Pengujian Nilai Epsilon



Gambar 5.6 Grafik Akurasi Terhadap Nilai Epsilon

Gambar 5.6 menunjukkan grafik pengujian nilai ϵ (*epsilon*) dimana diketahui akurasi tertinggi sebesar 89.94% pada nilai ϵ (*epsilon*) 0.000000001, 0.00000001, 0.0000001, 0.000001, 0.0001, 0.0001, dan 0.001. Semakin besar nilai ϵ (*epsilon*) maka iterasi yang dihasilkan akan semakin sedikit. Apabila semakin kecil nilai ϵ (*epsilon*) maka jumlah iterasi akan semakin banyak sehingga proses *learning* akan berlangsung lama, maka nilai α dan nilai *support vector* akan semakin optimal. Iterasi berhenti pada nilai ϵ 0.0001, dan 0.001 karena telah mencapai konvergen dimana nilai $\max(|\delta\alpha_i|) < \epsilon$. Nilai α



(α) dan *support vector* yang tidak optimal akan mempengaruhi nilai akurasi yang dihasilkan semakin rendah.

5.6 Pengujian Tingkat Akurasi Terhadap Iterasi Maksimum

Pengujian tingkat akurasi terhadap iterasi maksimum akan dijelaskan pada skenario pengujian dan hasil pengujian.

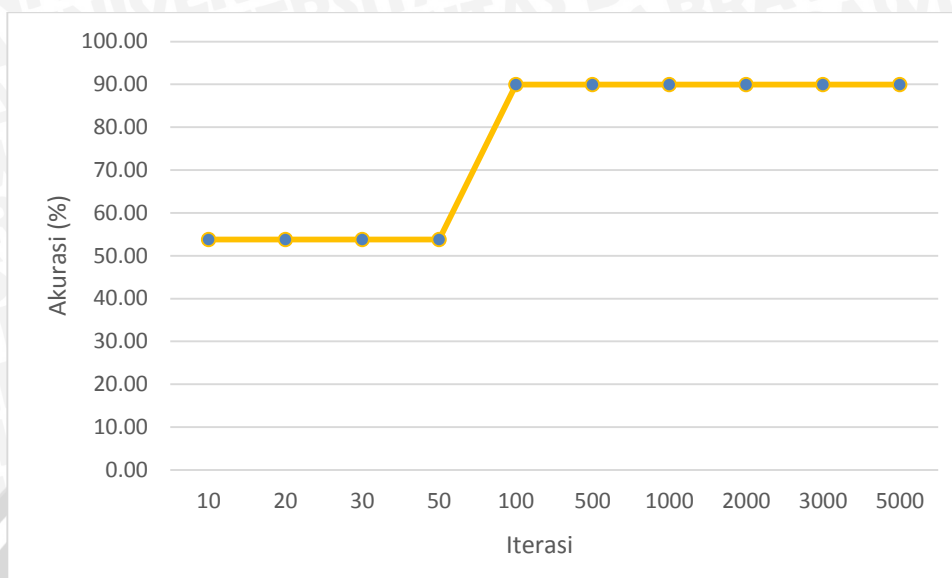
5.6.1 Skenario Pengujian Iterasi Maksimum

Tabel 5.7 menunjukkan hasil skenario pengujian terhadap iterasi maksimum dengan menggunakan parameter rasio perbandingan 90%:10%, %, nilai *threshold* 0.01, nilai dari $\lambda = 0.5$, konstanta γ (*gamma*) = 0.003, ε (*epsilon*) = 0.001, dan $C = 1$. Beberapa iterasi yang dilakukan mulai dari iterasi 10, 20, 30, 50, 100, 500, 1000, 2000, 3000 dan 5000.

Tabel 5.7 Hasil Pengujian Iterasi Maksimum

Iterasi	Percobaan ke- <i>i</i>					Rata-rata
	1	2	3	4	5	
10	60.00	39.13	60.00	50.00	60.00	53.83
20	60.00	39.13	60.00	50.00	60.00	53.83
30	60.00	39.13	60.00	50.00	60.00	53.83
50	60.00	39.13	60.00	50.00	60.00	53.83
100	93.10	77.77	93.10	100.00	85.71	89.94
500	93.10	77.77	93.10	100.00	85.71	89.94
1000	93.10	77.77	93.10	100.00	85.71	89.94
2000	93.10	77.77	93.10	100.00	85.71	89.94
3000	93.10	77.77	93.10	100.00	85.71	89.94
5000	93.10	77.77	93.10	100.00	85.71	89.94

5.6.2 Analisa Pengujian Iterasi Maksimum



Gambar 5.7 Grafik Tingkat Akurasi Hasil Pengujian Iterasi Maksimum

Gambar 5.8 merupakan grafik hasil pengujian iterasi maksimum dimana diketahui rata-rata akurasi terbaik yakni sebesar 89.94% pada iterasi ke 100, 500, 1000, 2000, 3000 dan 5000 cenderung stabil karena pada iterasi tersebut telah mencapai konvergen. Nilai iterasi maksimum dipengaruhi oleh nilai ϵ (*epsilon*) yang digunakan. Pada pengujian ini telah ditentukan nilai ϵ (*epsilon*) = 0.001 dan iterasi maksimum=5000. Pada pengujian dengan iterasi 10, 20, dan 30 iterasi berhenti pada masing-masing batas maksimum iterasi karena belum memenuhi syarat konvergen dimana nilai dari $\max(|\delta\alpha_i|) < \epsilon$. Konvergen didefinisikan sebagai tingkat perubahan nilai α (*alpha*) dimana nilai berhentinya suatu iterasi akan berpengaruh pada perubahan nilai α (*alpha*) dan b (*bias*) pada proses pelatihan .

5.7 Pengujian Terhadap Pengaruh Nilai C

Pengujian untuk mengetahui pengaruh nilai C pada tingkat akurasi akan dijelaskan pada skenario pengujian dan analisa pengujian.

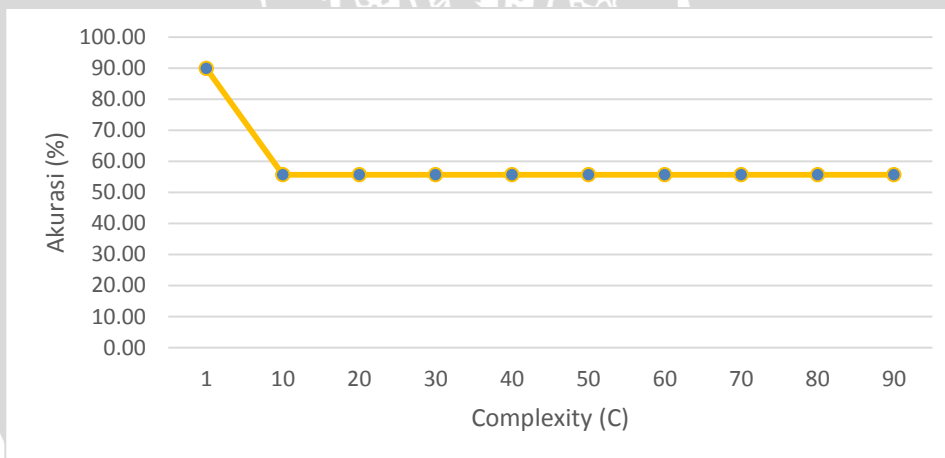
5.7.1 Skenario Pengujian Terhadap Pengaruh Nilai C

Tabel 5.7 menunjukkan hasil pengujian pengaruh nilai C terhadap tingkat akurasi yang dihasilkan. Parameter yang digunakan yakni rasio 90%:10%, nilai *threshold* 0.01, nilai dari $\lambda = 0.5$, konstanta γ (*gamma*) = 0.003, ϵ (*epsilon*) =0.001, dan iterasi maksimum=100. Beberapa nilai C (*Complexity*) yang digunakan yakni 1, 10, 20, 30, 40, 50, 60, 70, 80 dan 90.

Tabel 5.6 Hasil Pengujian Terhadap Pengaruh Nilai C

C	Percobaan ke- <i>i</i>					Rata-rata
	1	2	3	4	5	
1	93.10	77.77	93.10	100.00	85.71	89.94
10	60.00	39.13	69.23	50.00	60.00	55.67
20	60.00	39.13	69.23	50.00	60.00	55.67
30	60.00	39.13	69.23	50.00	60.00	55.67
40	60.00	39.13	69.23	50.00	60.00	55.67
50	60.00	39.13	69.23	50.00	60.00	55.67
60	60.00	39.13	69.23	50.00	60.00	55.67
70	60.00	39.13	69.23	50.00	60.00	55.67
80	60.00	39.13	69.23	50.00	60.00	55.67
90	60.00	39.13	69.23	50.00	60.00	55.67

5.7.2 Analisa Pengujian Pengaruh Nilai C



Gambar 5.8 Grafik Akurasi Terhadap Pengaruh Nilai C

Gambar 5.8 merupakan grafik hasil pengujian pengaruh nilai C (*Complexity*) terhadap tingkat akurasi yang dihasilkan dimana diketahui tingkat akurasi terbaik sebesar 89.94% dengan nilai C=1. Tujuan dari nilai C (*Complexity*) adalah untuk meminimalkan *error*. Semakin dekat nilai C (*Complexity*) dengan 0 maka lebar margin pada bidang pemisah (*hyperplane*) menjadi maksimum. Sebaliknya jika semakin besar nilai C (*Complexity*) maka tingkat akurasi yang dihasilkan akan semakin menurun karena jumlah data yang dilatih berada dalam *margin* yang salah sehingga data *testing* tidak dapat diklasifikasikan dengan benar.

5.8 Analisis Tingkat Akurasi

Analisis hasil pengujian keseluruhan terhadap waktu eksekusi perhitungan sistem dengan metode AHP-SVM didapatkan waktu terlama yakni 25.96 detik sedangkan waktu tercepat 22.52 detik. Rata-rata keseluruhan waktu eksekusi selama 24.69 detik. Hasil akurasi terbaik pada rasio 90%:10% sebesar 88.56%. Rasio tersebut digunakan kembali untuk melakukan pengujian terhadap parameter lainnya, dimana nilai akurasi tersebut meningkat menjadi 89.94% pada pengujian terhadap nilai *threshold*, *lamda*, *gamma*, *epsilon*, iterasi maksimum dan *C (Complexity)*. Nilai rata-rata akurasi secara keseluruhan yang dihasilkan oleh metode AHP dan SVM untuk Klasifikasi Penerima Beasiswa (Studi Kasus : Lembaga GNOTA Kediri) sebesar 89.74% dengan parameter Rasio Perbandingan 90%:10%, Nilai *Threshold* AHP=0.01, nilai parameter *lamda* = 0.5, nilai konstanta *gamma* $\gamma = 0.003$, nilai *epsilon* (ϵ) = 0.001, iterasi maksimum = 100 dan nilai parameter *C* = 1.



BAB 6 PENUTUP

Bab ini membahas kesimpulan dan saran dari seluruh proses penelitian Penerapan Metode AHP dan SVM untuk Klasifikasi Penerima Beasiswa pada Lembaga GNOTA Kediri.

6.1 Kesimpulan

Berdasarkan hasil penelitian dan pengujian yang telah dilakukan, dapat diambil kesimpulan sebagai berikut :

1. Klasifikasi penerima beasiswa dengan metode AHP dan SVM dengan beberapa langkah, yakni pertama penghitungan bobot kriteria dan sub kriteria data matrik perbandingan berpasangan. Selanjutnya dilakukan penyaringan terhadap bobot kriteria terpilih dimana nilai bobot pada sub kriteria tersebut di atas nilai *threshold* yang ditetapkan. Nilai *threshold* adalah nilai batasan bobot kriteria yang digunakan untuk pengujian pada proses selanjutnya. Pada penelitian ini menggunakan data yang diperoleh dari hasil *interview* dengan Lembaga GNOTA Kediri. Untuk proses klasifikasi dengan metode SVM diawali dengan load data *training* dan data *testing* yang dipilih secara acak berdasarkan rasio data tertentu. Selanjutnya dilakukan normalisasi dataset. Hasil dari normalisasi dataset akan dikalikan dengan sub bobot kriteria terpilih AHP. Selanjutnya dilakukan proses perhitungan *kernel* dan *Sequential Training SVM*. Setelah proses training selesai akan dilakukan proses *testing* yang akan menghasilkan klasifikasi berupa *predicted class* hasil pengujian.
2. Tingkat akurasi rata-rata yang dihasilkan oleh sistem adalah sebesar 89.74% dengan pemilihan rasio 90%:10%, nilai *threshold* AHP = 0.1, nilai lamda = 0.5, nilai konstanta gamma=0.003, nilai epsilon=0.001, iterasi maksimum=100 dan nilai C=1.

6.2 Saran

Saran yang dapat diberikan untuk penelitian selanjutnya antara lain :

1. Penelitian selanjutnya diharapkan melakukan seleksi fitur secara otomatis menggunakan algoritma evolusi atau PCA karena dalam penelitian ini masih menggunakan *threshold*.
2. Pada penelitian selanjutnya, dalam pemilihan rasio perbandingan antara data *training* dan data *testing* diharapkan menggunakan metode *K-Fold*, sehingga setiap data memiliki kesempatan untuk menjadi data *training* dan data *testing*.

DAFTAR PUSTAKA

- Alonso, Jose Antonio., dan Lamata, M Teresa. 2006. *Consistency In The AHP : A New Approach*. Vo. 14, No.4, Hal 445-459.
- C. Cortes dan V. Vapnik, 1995, *Machine Learning, Support Vector Network*, Vol.20, Hal 273-297.
- Chih, Wei Hsu., Chih, Jen Lin. 2002. A Comparison of Methods for Multiclass Support Vector Machines. Vol.13, No.2.
- Damyanti Fitri, Arifin Agus Zainal, Soelaiman Rully. 2010. Jurnal Ilmiah Cursor, Pengenalan Citra Wajah Menggunakan Metode *Two-Dimensional Linier Discriminant Analysis* dan *Support Vector Machine*, Vol. 5, No 3, Hal 147-156.
- GNOTA : Apa itu GNOTA. Diakses pada 20 September 2015 dari <http://www.gnoto.or.id/index.php?controller=page&action=view&page=tentangnota&subpage=tentang-kami&lang=id>.
- Gafur, Abdul. 2008. Cara Mudah Mendapatkan Beasiswa. Jakarta : Penebar Plus Daihani, Dadan Umar. 2001. Komputerisasi Pengambilan Keputusan. Jakarta : Elex Media Komputindo.
- Hermaduanti, N., n,d. 2008. Sistem Pendukung Keputusan Berbasis SMS untuk Menentukan Status Gizi Dengan Metode K-Nearest Neighbor. Universitas Islam Indonesia.
- Jiawei Han *et all*. 2012. *Data Mining Concepts and Techniques Third Edition*. Elsevier Inc. USA.
- Kartal Hasan Basri, Cebi Ferhan. 2013, *International journal of Machine Learning and Computing, Support Vectore Machine for Multi-Attribute ABC Analysis*, Vol 3, No 1, Hal. 154-157
- Kumar Jain, Yogendra., Kumar Bhandare, Santosh. 2011, Min Max Normalization Based Data Perturbation Method for Privacy Protection. Samrat Ashok Technological Institute.
- Mohanty, S. Dan Bebartta, H. N. D., 2011. *Performace Comparison of SVM and K-NN for Oriya Character Recognition*. (IJACSA) International Journal of Advanched Computer Science and Applications, Special Issue on Image Processing and Analysis, Hal 112-116.
- Mujiasih Subekti. 2011, Jurnal Meteorologi dan Geofisika, Pemanfaatan *Data Mining untuk Perkiaraan Cuaca*, Vol 12, No 2, Hal 189-195.
- Novianti Fourina Ayu, Purnami Santi Wulan. 2012, Jurnal Sains dan Seni ITS, Analisis Diagnosis Pasien Kanker Payudara Menggunakan Regresi Logistik dan Support Vector Machine (SVM) Berdasarkan Hasil Mamografi, Vol 1, No1, Hal D-147 – D-152

- Peng, Jianliang. 2012, *Selection of Logistics Outsourcing Service Suppliers Based on AHP*. Huangzhou, China.
- Purnami, S. W., dan Embong, A. 2008. *Smooth Support Vector Machine For Breast Cancer Classification*. ICMSA08, Banda Aceh, Indonesia.
- Gaspar Paulo, Carbonell Jaime, Jose Luis Oliveira. 2012. "On The Parameter Optimization of Support Vector Machines for Binary Clasification", Vol 3, Hal 201
- Rachman Fairizi, Purnami Santi Wulan. 2102, Jurnal Sains dan Seni ITS, Perbandingan Klasifikasi Tingkat Keganasan *Breast Cancer* dengan Menggunakan Regresi Logistik Ordinal dan *Support Vector Machine*, Vol 1, No 1, Hal D-130 – D135.
- Saaty, T. L. 1980. *The Analytic Hierarchy Process :Planning, Priority Allocation*.
- Saaty, T. L. 1991. Pengambilan Keputusan Bagi Para Pemimpin. PT. Pustaka Binaman Pressindo.
- Sembiring, 2007. Penerapan Teknik Support Vector Machine untuk Pendeteksian Intrusi pada Jaringan. Institute Teknologi Bandung.
- Sparague, R.H dan Watson H. J., 1993. *Decision Support Systems : Putting Theory Into Practice*. Englewood Clifts, N. J., Prentice Hall.
- Turban, Efraim., Aronson, Jay E. & Liang, Ting-Peng., 2005. *Decision Support Systems and Intelligent Systems*, 7th ed. Yogyakarta: ANDI.
- Vijayakumar S, Wu S. Proc. International Conference on Soft Computing (SOCO'99), *Sequential Support Vector Classifier and Regression*, Genoa, Italy, pp.610-619, 1999.
- Vijayarani S, Dhayanand S. 2015. *Liver Disease Prediction using SVM and Naive Bayes Algorithm*. Vol 4, Hal 2278-7798.
- Yang Yi, Du Quishi, Zhao Jinying. 2010, *Logistic System and Intelligent Management, The Application of site selection based on AHP-SVM in 500KV substation*, Vol 2, Hal 1225 – 1229.



LAMPIRAN

Lampiran A Surat Rekomendasi Penelitian



PEMERINTAH KABUPATEN KEDIRI
BADAN KESATUAN BANGSA DAN POLITIK
JALAN SOEKARNO HATTA NOMOR 1 TELEPON 689969
KEDIRI

Website : www.Kedirikab.go.id Email : bakesbangpol@kedirikab.go.id

Kediri, 23 September 2015

Nomor : 070/ 537 /418.62/2015
 Sifat : Biasa
 Lampiran :
 Perihal : Rekomendasi

KEPADA
 YTH. SDR. KA DINAS PENDIDIKAN PEMUDA
 DAN OLAH RAGA KAB. KEDIRI
 DI
KEDIRI

REKOMENDASI

Menunjuk Surat Saudara Wakil Ketua I Bidang Akademik Universitas Brawijaya Malang tanggal 21 September 2015 Nomor : 3295/UN10.3/AK/2015 perihal Permohonan Data Skripsi.

Berdasarkan :

1. Peraturan Menteri Dalam Negeri Nomor 7 Tahun 2014 tentang Perubahan Atas Permendagri Nomor 64 Tahun 2011 tentang Pedoman Penerbitan Rekomendasi Penelitian;
2. Peraturan Daerah Kabupaten Kediri Nomor 1 Tahun 2015 tentang Tata Kerja Badan Kesatuan Bangsa dan Politik;
3. Peraturan Bupati Kediri Nomor 4 tahun 2015 tentang Penjabaran Tugas dan Fungsi Badan Kesatuan Bangsa dan Politik;
4. Surat Edaran Bupati Kediri tanggal 6 Agustus 2012 Nomor : 070/1541/418.62/2012 Perihal Perubahan Proses Penerbitan Rekomendasi Penelitian

Bersama ini diberitahukan bahwa :

Nama : 1. GILVY LANGGAWAN PUTRA
 2. WHENTY ARIYANTI
 Pekerjaan : Mahasiswa Universitas Brawijaya Malang
 Alamat : Jl. Veteran Malang
 Kebangsaan : Indonesia

Diberikan Rekomendasi untuk mengadakan kegiatan dimaksud di Unit / Wilayah Kerja Saudara dengan :

Judul : 1. Sistem Pendukung Keputusan Rekomendasi Penerima Beasiswa Menggunakan Metode SVM dan Tosis di Lembaga GN-OTA Kediri
 2. Penerapan metode SVM dan AHP pada system Pendukung Keputusan Pemilihan Penerima Beasiswa (Studi Kasus : Lembaga GN-OTA Kediri)
 Penanggung jawab : Ir. Heru Nurwarsito, M.Kom.
 Waktu : 3 (tiga) Bulan sejak tanggal rekomendasi diterbitkan
 Lokasi : Dinas Dikpora Kab. Kediri. (GN-OTA).
 Peserta :

Penerima rekomendasi wajib mentaati peraturan dan tata tertib yang berlaku dan apabila selesai melaksanakan kegiatannya diwajibkan memberikan laporan.

Demikian Rekomendasi ini dikeluarkan untuk dilaksanakan dengan sebaik-baiknya

a.n. KEPALA BAKESBANGPOL
 KABUPATEN KEDIRI
 Kepala Bidang Kewaspadaan



Iwan Agus Wijaya
IWAN AGUS WIJAYA, S.Sos
 Penata Tingkat I
 NIP. 19710808 199101 1 001

TEMBUSAN : Kepada YTH.

1. Ibu Bupati Kediri (sebagai laporan);
2. Sdr. Kepala Bappeda Kab. Kediri
3. Sdr. Wakil Ketua I Bidang Akademik UNBRA Malang
4. Sdr. Yang Berangkutan



Lampiran B Surat Persetujuan Menjadi Pakar

SURAT PERSETUJUAN MENJADI PAKAR

Saya yang bertanda tangan di bawah ini :

Nama : *AJI INDRA LAKSANA*
Pekerjaan :

denyatakan bersedia untuk berpartisipasi menjadi pakar dalam mendukung penyelesaian skripsi mahasiswa berikut :

Nama : Whenty Ariyanti
NIM : 145150209111021
Jurusan : Teknik Informatika
Fakultas : Fakultas Ilmu Komputer
Universitas : Universitas Brawijaya Malang
Judul Skripsi : Penerapan Metode AHP dan SVM untuk Klasifikasi Penerima Beasiswa (Studi Kasus : Lembaga GNOTA Kediri)

dan menyatakan bahwa data yang telah diklasifikasikan dan digunakan dalam skripsi mahasiswa tersebut telah dikonsultasikan dan disetujui.

Kediri 23 September 2015
Menyetujui,
Pakar,



AJI INDRA LAKSANA
NIP.

Lampiran C Formulir Pendaftaran Beasiswa

Istrumen Penerima Beasiswa L-GNOTA Kab. Kediri

NO :

1. Biodata Penerima Beasiswa

Nama Siswa	:	
Jenis Kelamin	:	<input type="checkbox"/> Laki-laki <input type="checkbox"/> Perempuan
Alamat Lengkap	:	
Nomor Ponsel	:	
Sekolah Asal	:	

2. Informasi Pendaftaran

Jalur Penerimaan	:	
Diterima di PTN	:	
Program Beasiswa Yang Diikuti	:	Bidik Misi/Lainnya*
Sebutkan	:	
Sekolah Asal	:	

3. Data Orang Tua/Wali

Nama Ayah	:	
Pendidikan Terakhir	:	
Pekerjaan	:	
Status Pekerjaan	:	
Penghasilan/Gaji	Bulan Sekarang	: Rp.
	Bulan Lalu	: Rp.
Nama Ibu	:	
Pendidikan Terakhir	:	
Pekerjaan	:	
Status Pekerjaan	:	
Penghasilan/Gaji	Bulan Sekarang	: Rp.
	Bulan Lalu	: Rp.



Jumlah Anak	:	Jiwa
Anak yang masih menjadi tanggungan	:	Jiwa
Tagihan PBB Sekarang	Tahun	: Rp.
Daya Listrik	:	Watt
Tagihan Listrik		
Bulan Sekarang	:	Rp.
Bulan Lalu	:	Rp.
2 Bulan Lalu	:	Rp.
Kepemilikan Kendaraan Bermotor		
Motor Roda 2	:	Unit (Tahun.....)
Mobil	:	Unit (Jenis.....Tahun.....)
Rumah		
a. Jumlah Rumah	:	Unit
b. Status	:	Rumah Sendiri/Sewa/Numpang
c. Dinding	:	Tembok/Papan Kayu/Bambu
d. Lantai	:	Keramik/Tegel/Semen/Tanah
Biaya Daftar Ulang	:	Rp.

4. Catatan Lain-lain :

A

B

C

*coret yang tidak perlu

Responden _____ Kediri, _____, 20..
 Surveyor _____



Lampiran D Inisialisasi Data

INISIALISASI DATA

PENGHASILAN (C1)	BOBOT
Rp. 0	5
Rp. 1 sd Rp. 500.000	4
Rp. 500.001 sd Rp. 1.000.000	3
Rp. 1.000.001 sd Rp. 2.000.000	2
> Rp. 2.000.000	1
JML TANGGUNGAN (C2)	BOBOT
> 5	3
3 sd 5	2
0 sd 2	1
PEKERJAAN (C3)	BOBOT
TIDAK BEKERJA	5
BURUH	4
KARYAWAN	3
WIRASWASTA	2
PENGUSAHA	1
DAYA LISTRIK (C5)	BOBOT
0 W	6
220 W	5
450 W	4
900 W	3
1300 W	2
2200 W	1
JUMLAH RUMAH (C6)	BOBOT
0 RUMAH	3
1 RUMAH	2
> 1 RUMAH	1
STATUS RUMAH (C7)	BOBOT
KONTRAK	3
NUMPANG	2
MILIK SENDIRI	1

KONDISI DINDING (C8)	BOBOT
BAMBU	3
KAYU	2
TEMBOK	1
KONDISI LANTAI (C9)	BOBOT
TANAH	3
TEGEL	2
KERAMIK	1
BIAYA PBB (C10)	BOBOT
Rp. 0 sd Rp. 50.000	4
Rp.50.001 sd Rp.100.000	3
Rp.100.001 sd Rp.200.000	2
> Rp.200.000	1
JUMLAH MOTOR (C11)	BOBOT
0 motor	5
1 motor	4
2 motor	3
3 motor	2
> 3 motor	1
JUMLAH MOBIL (C12)	BOBOT
0	3
1	2
> 1	1
PROGRAM BEASISWA (C13)	BOBOT
BIDIK MISI	2
LAINNYA	1

Kediri, 23 September 2015
Mengetahui,
Ketua Harian Lembaga GNOTA
Kab. Kediri



DR. Mokh. Muhsin, M.Pd
NIP.

Lampiran E Nilai Perbandingan Kriteria Utama

NILAI PERBANDINGAN KRITERIA UTAMA

NILAI PERBANDINGAN BERPASANGAN			
Angket	No	Nama Kriteria	1-9
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	1	• Penghasilan • Jumlah Tanggungan	6
	2	• Penghasilan • Pekerjaan	4
	3	• Penghasilan • Daya Listrik	3
	4	• Penghasilan • Biaya Listrik	3
	5	• Penghasilan • Jumlah Rumah	5
	6	• Penghasilan • Status Rumah	5
	7	• Penghasilan • Kondisi Dinding	4
	8	• Penghasilan • Kondisi Lantai	4
	9	• Penghasilan • Biaya PBB	4
	10	• Penghasilan • Jumlah Motor	5
	11	• Penghasilan • Jumlah Mobil	6
	12	• Penghasilan • Program Beasiswa	4
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	13	• Jumlah Tanggungan • Pekerjaan	4
	14	• Jumlah Tanggungan • Daya Listrik	3
	15	• Jumlah Tanggungan • Biaya Listrik	3
	16	• Jumlah Tanggungan • Jumlah Rumah	4
	17	• Jumlah Tanggungan • Status Rumah	4
	18	• Jumlah Tanggungan • Kondisi Dinding	3
	19	• Jumlah Tanggungan • Kondisi Lantai	3
	20	• Jumlah Tanggungan • Biaya PBB	3

	21	• Jumlah Tanggungan • Jumlah Motor	2
	22	• Jumlah Tanggungan • Jumlah Mobil	3
	23	• Jumlah Tanggungan • Program Beasiswa	3
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	24	• Pekerjaan • Daya Listrik	3
	25	• Pekerjaan • Biaya Listrik	3
	26	• Pekerjaan • Jumlah Rumah	5
	27	• Pekerjaan • Status Rumah	5
	28	• Pekerjaan • Kondisi Dinding	5
	29	• Pekerjaan • Kondisi Lantai	5
	30	• Pekerjaan • Biaya PBB	5
	31	• Pekerjaan • Jumlah Motor	3
	32	• Pekerjaan • Jumlah Mobil	3
	33	• Pekerjaan • Program Beasiswa	5
	Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	34	• Daya Listrik • Biaya Listrik
35		• Daya Listrik • Jumlah Rumah	4
36		• Daya Listrik • Status Rumah	4
37		• Daya Listrik • Kondisi Dinding	4
38		• Daya Listrik • Kondisi Lantai	3
39		• Daya Listrik • Biaya PBB	3
40		• Daya Listrik • Jumlah Motor	2
41		• Daya Listrik • Jumlah Mobil	2
42		• Daya Listrik • Program Beasiswa	4

	43	• Biaya Listrik • Jumlah Rumah	1
	44	• Biaya Listrik • Status Rumah	2
	45	• Biaya Listrik • Kondisi Dinding	3
	46	• Biaya Listrik • Kondisi Lantai	3
	47	• Biaya Listrik • Biaya PBB	3
	48	• Biaya Listrik • Jumlah Motor	2
	49	• Biaya Listrik • Jumlah Mobil	2
	50	• Biaya Listrik • Program Beasiswa	3
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	51	• Jumlah Rumah • Status Rumah	4
	52	• Jumlah Rumah • Kondisi Dinding	3
	53	• Jumlah Rumah • Kondisi Lantai	3
	54	• Jumlah Rumah • Biaya PBB	4
	55	• Jumlah Rumah • Jumlah Motor	5
	56	• Jumlah Rumah • Jumlah Mobil	5
	57	• Jumlah Rumah • Program Beasiswa	6
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	58	• Status Rumah • Kondisi Dinding	1
	59	• Status Rumah • Kondisi Lantai	1
	60	• Status Rumah • Biaya PBB	1
	61	• Status Rumah • Jumlah Motor	2
	62	• Status Rumah • Jumlah Mobil	2
	63	• Status Rumah • Program Beasiswa	2
	64	• Kondisi Dinding • Kondisi Lantai	2

Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	65	• Kondisi Dinding • Biaya PBB	1
	66	• Kondisi Dinding • Jumlah Motor	3
	67	• Kondisi Dinding • Jumlah Mobil	4
	68	• Kondisi Dinding • Program Beasiswa	2
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	69	• Kondisi Lantai • Biaya PBB	2
	70	• Kondisi Lantai • Jumlah Motor	3
	71	• Kondisi Lantai • Jumlah Mobil	4
	72	• Kondisi Lantai • Program Beasiswa	1
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	73	• Biaya PBB • Jumlah Motor	3
	74	• Biaya PBB • Jumlah Mobil	4
	75	• Biaya PBB • Program Beasiswa	2
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	76	• Jumlah Motor • Jumlah Mobil	3
	77	• Jumlah Motor • Program Beasiswa	5
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	78	• Jumlah Mobil • Program Beasiswa	6

Kediri, 23 September 2015
Mengetahui,
Ketua Harian Lembaga GNOTA
Kab. Kediri



DR. Mokh. Muhsin, M.Pd
NIP.

Lampiran F Matrik Perbandingan Berpasangan Kriteria Utama

MATRIKS PERBANDINGAN BERPASANGAN

KRITERIA	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13
C1	1	6	4	3	3	5	5	4	4	4	5	6	4
C2	0.167	1	4	3	3	4	4	3	3	3	2	3	3
C3	0.25	0.25	1	3	3	5	5	5	5	5	3	3	5
C4	0.33	0.33	0.33	1	3	4	4	4	3	3	2	2	4
C5	0.33	0.33	0.33	0.33	1	1	2	3	3	3	2	2	3
C6	0.2	0.25	0.20	0.25	1	1	4	3	3	4	5	5	6
C7	0.2	0.25	0.20	0.25	0.5	0.25	1	1	1	1	2	2	2
C8	0.25	0.33	0.20	0.25	0.33	0.33	1	1	2	1	3	4	2
C9	0.25	0.33	0.20	0.33	0.33	0.33	1	0.5	1	2	3	4	1
C10	0.25	0.33	0.20	0.33	0.33	0.25	1	1	0.5	1	3	4	2
C11	0.2	0.5	0.33	0.5	0.5	0.2	0.5	0.33	0.33	0.33	1	3	5
C12	0.167	0.33	0.33	0.5	0.5	0.2	0.5	0.25	0.25	0.25	0.33	1	6
C13	0.25	0.33	0.20	0.25	0.33	0.167	0.5	0.5	1	0.5	0.2	0.167	1

KETERANGAN :

- C1 : Penghasilan
- C2: Jumlah Tanggungan
- C3: Pekerjaan
- C4: Daya Listrik
- C5: Biaya Listrik
- C6: Jumlah Rumah
- C7: Status Rumah
- C8: Status Dinding
- C9: Status Lantai
- C10: PBB
- C11: Jumlah Motor
- C12: Jumlah Mobil
- C13: Program Beasiswa

Kediri, 23 September 2015
 Mengetahui,
 Ketua Harian Lembaga GNOTA
 Kab. Kediri



DR. Mokh. Muhsin, M.Pd
NIP.



Lampiran G Nilai Perbandingan Sub Kriteria

NILAI PERBANDINGAN BERPASANGAN			
SUB KRITERIA PENGHASILAN (C1)			
Angket	No	Nama Sub Kriteria	1-9
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	1	<ul style="list-style-type: none"> Rp.0 Rp. 1 sd Rp.500.000 	2
	2	<ul style="list-style-type: none"> Rp.0 Rp. 500.001 sd Rp.1.000.000 	4
	3	<ul style="list-style-type: none"> Rp.0 Rp. 1.000.001 sd Rp.2.000.000 	5
	4	<ul style="list-style-type: none"> Rp.0 > Rp.2.000.000 	7
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	5	<ul style="list-style-type: none"> Rp. 1 sd Rp.500.000 500.001 sd Rp.1.000.000 	4
	6	<ul style="list-style-type: none"> Rp. 1 sd Rp.500.000 Rp.1.000.001 sd Rp.2.000.000 	4
	7	<ul style="list-style-type: none"> Rp. 1 sd Rp.500.000 > Rp.2.000.000 	6
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	8	<ul style="list-style-type: none"> Rp.500.001 sd Rp.1.000.000 Rp.1.000.001 sd Rp.2.000.000 	3
	9	<ul style="list-style-type: none"> Rp.500.001 sd Rp.1.000.000 > Rp.2.000.000 	3
	10	<ul style="list-style-type: none"> Rp.1.000.001 sd Rp.2.000.000 > Rp.2.000.000 	2
SUB KRITERIA JUMLAH TANGGUNGAN (C2)			
Angket	No	Nama Sub Kriteria	1-9
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	1	<ul style="list-style-type: none"> > 5 anak 3 sd 5 anak 	3
	2	<ul style="list-style-type: none"> > 5 anak 0 sd 3 	4
	3	<ul style="list-style-type: none"> 3 sd 5 anak 	3



• 0 sd 3 anak			
SUB KRITERIA PEKERJAAN (C3)			
Angket	No	Nama Sub Kriteria	1-9
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	1	• Tidak Bekerja • Buruh	3
	2	• Tidak Bekerja • Karyawan	5
	3	• Tidak Bekerja • Wiraswasta	6
	4	• Tidak Bekerja • Pengusaha	7
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	5	• Buruh • Karyawan	2
	6	• Buruh • Wiraswasta	4
	7	• Buruh • Pengusaha	5
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	8	• Karyawan • Wiraswasta	3
	9	• Karyawan • Pengusaha	4
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	10	• Wiraswasta • Pengusaha	2
SUB KRITERIA DAYA LISTRIK (C4)			
Angket	No	Nama Sub Kriteria	1-9
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	1	• 0 W • 120 W	-
	2	• 0 W • 220 W	2
	3	• 0 W • 450 W	2
	4	• 0 W • 900 W	4
	5	• 0 W • 1300 W	5
	6	• 0 W • 2200 W	7
	7	• 220 W • 450 W	3

Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	8	<ul style="list-style-type: none"> • 220 W • 900 W 	4
	9	<ul style="list-style-type: none"> • 220 W • 1300 W 	6
	10	<ul style="list-style-type: none"> • 220 W • 2200 W 	6
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	16	<ul style="list-style-type: none"> • 450 W • 900 W 	3
	11	<ul style="list-style-type: none"> • 450 W • 1300 W 	4
	12	<ul style="list-style-type: none"> • 450 W • 2200 W 	5
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	13	<ul style="list-style-type: none"> • 900 W • 1300 W 	3
	14	<ul style="list-style-type: none"> • 900 W • 2200 W 	3
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	15	<ul style="list-style-type: none"> • 1300 W • 2200 W 	2
SUB KRITERIA BIAYA LISTRIK (C5)			
Angket	No	Nama Sub Kriteria	1-9
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	1	<ul style="list-style-type: none"> • Rp. 0 • Rp. 1 sd Rp. 50.000 	3
	2	<ul style="list-style-type: none"> • Rp. 0 • Rp. 50.001 sd Rp.100.000 	5
	3	<ul style="list-style-type: none"> • Rp. 0 • Rp. 100.001 sd Rp. 150.000 	6
	4	<ul style="list-style-type: none"> • Rp. 0 • > Rp. 50.000 	7
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	5	<ul style="list-style-type: none"> • Rp. 1 sd Rp. 50.000 • Rp. 50.001 sd Rp.100.000 	3
	6	<ul style="list-style-type: none"> • Rp. 1 sd Rp. 50.000 • Rp. 100.001 sd Rp. 150.000 	4
	7	<ul style="list-style-type: none"> • Rp. 1 sd Rp. 50.000 • > Rp. 150.000 	5
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	8	<ul style="list-style-type: none"> • Rp.50.001 sd Rp. 100.000 • Rp. 100.001 sd Rp. 150.000 	3
	9	<ul style="list-style-type: none"> • Rp.50.001 sd Rp. 100.000 • > Rp. 150.000 	3
Berapa nilai perbandingan berpasangan antar kriteria berikut?	10	<ul style="list-style-type: none"> • Rp. 100.001 sd Rp. 150.000 • > Rp. 150.000 	2

Manakah yang lebih penting?			
SUB KRITERIA JUMLAH RUMAH (C6)			
Angket	No	Nama Sub Kriteria	1-9
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	1	<ul style="list-style-type: none"> • 0 Rumah • 1 Rumah 	5
	2	<ul style="list-style-type: none"> • 0 Rumah • > 1 Rumah 	6
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	3	<ul style="list-style-type: none"> • 1 Rumah • >1 Rumah 	3
SUB KRITERIA STATUS RUMAH (C7)			
Angket	No	Nama Sub Kriteria	1-9
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	1	<ul style="list-style-type: none"> • Kontrak • Numpang 	6
	2	<ul style="list-style-type: none"> • Kontrak • Milik Sendiri 	7
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	3	<ul style="list-style-type: none"> • Numpang • Milik Sendiri 	5
SUB KRITERIA KONDISI DINDING (C8)			
Angket	No	Nama Sub Kriteria	1-9
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	1	<ul style="list-style-type: none"> • Bambu • Kayu 	5
	2	<ul style="list-style-type: none"> • Bambu • Tembok 	7
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	3	<ul style="list-style-type: none"> • Kayu • Tembok 	4
SUB KRITERIA KONDISI LANTAI (C9)			
Angket	No	Nama Sub Kriteria	1-9
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	1	<ul style="list-style-type: none"> • Tanah • Tegel 	5
	2	<ul style="list-style-type: none"> • Tanah • Keramik 	6

Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	3	<ul style="list-style-type: none"> • Tegel • Keramik 	4
SUB KRITERIA BIAYA PBB (C10)			
Angket	No	Nama Sub Kriteria	1-9
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	1	<ul style="list-style-type: none"> • Rp. 0 sd Rp. 50.000 • Rp. 50.001 sd Rp. 100.000 	3
	2	<ul style="list-style-type: none"> • Rp. 0 sd Rp. 50.000 • Rp. 100.001 sd Rp. 200.000 	4
	3	<ul style="list-style-type: none"> • Rp. 0 sd Rp. 50.000 • > Rp. 200.000 	6
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	4	<ul style="list-style-type: none"> • Rp. 50.001 sd Rp. 100.000 • Rp. 100.001 sd Rp. 200.000 	3
	5	<ul style="list-style-type: none"> • Rp. 50.001 sd Rp. 100.000 • > Rp. 200.000 	4
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	6	<ul style="list-style-type: none"> • Rp. 100.001 sd Rp. 200.000 • > Rp. 200.000 	3
SUB KRITERIA JUMLAH MOTOR (C11)			
Angket	No	Nama Sub Kriteria	1-9
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	1	<ul style="list-style-type: none"> • 0 motor • 1 motor 	3
	2	<ul style="list-style-type: none"> • 0 motor • 2 motor 	3
	3	<ul style="list-style-type: none"> • 0 motor • 3 motor 	4
	4	<ul style="list-style-type: none"> • 0 motor • > 3 motor 	5
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	5	<ul style="list-style-type: none"> • 1 motor • 2 motor 	2
	6	<ul style="list-style-type: none"> • 1 motor • 3 motor 	3
	7	<ul style="list-style-type: none"> • 1 motor • > 3 motor 	4
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	8	<ul style="list-style-type: none"> • 2 motor • 3 motor 	1
	9	<ul style="list-style-type: none"> • 2 motor • > 3 motor 	3



Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	10	<ul style="list-style-type: none"> • 3 motor • > 3 motor 	3
SUB KRITERIA JUMLAH MOBIL (C12)			
Angket	No	Nama Sub Kriteria	1-9
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	1	<ul style="list-style-type: none"> • 0 mobil • 1 mobil 	2
	2	<ul style="list-style-type: none"> • 0 mobil • > 1 mobil 	3
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	3	<ul style="list-style-type: none"> • 1 mobil • > 1 mobil 	2
SUB KRITERIA PROGRAM BEASISWA (C13)			
Angket	No	Nama Sub Kriteria	1-9
Berapa nilai perbandingan berpasangan antar kriteria berikut? Manakah yang lebih penting?	1	<ul style="list-style-type: none"> • BEASISWA BIDIK MISI • BEASISWA LAINNYA 	1

Kediri, 23 September 2015
Mengetahui,
Ketua Harian Lembaga GNOTA
Kab. Kediri



DR. Mokh. Muhsin, M.Pd
NIP.



Lampiran H Matrik Perbandingan Sub Kriteria

MATRIK PERBANDINGAN BERPASANGAN SUB KRITERIA

SUB KRITERIA PENGHASILAN (C1)					
	Rp.0	Rp.1 sd Rp.500.000	Rp.500.001 sd Rp.1.000.000	Rp.1.000.001 sd Rp.2.000.00	> Rp.2.000.000
Rp. 0	1	2	4	5	7
Rp.1 sd Rp.500.000	0.50	1	4	4	6
Rp. 500.001 sd Rp.1.000.000	0.25	0.25	1	3	3
Rp.1.000.001 sd Rp.2.000.000	0.20	0.25	0.33	1	2
> Rp.2.000.000	0.14	0.17	0.33	0.5	1
JUMLAH	2.09	3.67	9.67	13.50	19

SUB KRITERIA JUMLAH TANGGUNGAN (C2)			
	> 5 anak	3 sd 5 anak	0 sd 3 anak
> 5 anak	1	3	4
3 sd 5 anak	0.33	1	3
0 sd 3 anak	0.25	0.33	1
JUMLAH	1.58	4.33	8

SUB KRITERIA PEKERJAAN (C3)					
	TIDAK BEKERJA	BURUH	KARYAWAN	WIRASWASTA	PENGUSAHA
TIDAK BEKERJA	1	3	5	6	7
BURUH	0.33	1	2	4	5
KARYAWAN	0.20	0.50	1	3	4
WIRASWASTA	0.17	0.25	0.33	1	2
PENGUSAHA	0.14	0.20	0.25	0.50	1
JUMLAH	1.84	4.95	8.58	14.50	19

SUB KRITERIA DAYA LISTRIK (C4)						
	0 W	220 W	450 W	900 W	1300 W	2200 W
0 W	1	2	2	4	5	7
220 W	0.50	1	3	4	6	6
450 W	0.50	0.33	1	3	4	5
900 W	0.25	0.25	0.33	1	3	3
1300 W	0.20	0.17	0.25	0.33	1	2
2200 W	0.14	0.17	0.25	0.33	0.50	1
JUMLAH	2.59	3.92	6.78	12.67	19.50	24



SUB KRITERIA BIAYA LISTRIK (C5)

	Rp.0	Rp.1 sd Rp.50.000	Rp.50.001 sd Rp.100.001	Rp.100.001 sd Rp.150.000	> Rp.150.000
Rp.0	1	3	5	6	7
Rp.1 sd Rp.50.000	0-33	1	3	4	5
Rp.50.001 sd Rp.100.000	0-20	0-33	1	3	3
Rp.100.001 sd Rp.150.000	0-17	0-25	0-33	1	2
> Rp.150.000	0-14	0-20	0-33	0-50	1
JUMLAH	1-84	4-78	9-67	14-50	18

SUB KRITERIA JUMLAH RUMAH (C6)

	0 rumah	1 rumah	> 1 rumah
0 rumah	1	5	6
1 rumah	0-20	1	3
>1 rumah	0-17	0-33	1
JUMLAH	1-37	6-33	10

SUB KRITERIA STATUS RUMAH (C7)

	KONTRAK	NUMPANG	MILIK SENDIRI
KONTRAK	1	6	7
NUMPANG	0-17	1	5
MILIK SENDIRI	0-14	0-20	1
JUMLAH	1-31	7-20	13

SUB KRITERIA KONDISI DINDING (C7)

	BAMBU	KAYU	TEMBOK
BAMBU	1	5	7
KAYU	0-20	1	4
TEMBOK	0-14	0-25	1
JUMLAH	1-34	6-25	12

SUB KRITERIA KONDISI LANTAI (C9)

	TANAH	TEGEL	KERAMIK
TANAH	1	5	6
TEGEL	0-20	1	4
KERAMIK	0-17	0-25	1
JUMLAH	1-37	6-25	11

SUB KRITERIA PBB (C10)				
	Rp.0 sd Rp.50.000	Rp.50.001 sd Rp.100.001	Rp.100.001 sd Rp.200.00	> Rp.200.000
Rp.0 sd Rp.50.000	1	3	4	6
Rp.50.001 sd Rp.100.000	0.33	1	3	4
Rp.100.001 sd Rp.200.000	0.25	0.33	1	3
> Rp.200.000	0.17	0.25	0.33	1
JUMLAH	1.75	4.58	8.33	14

SUB KRITERIA JUMLAH MOTOR (C11)					
	0 motor	1 motor	2 motor	3 motor	> 3 motor
0 motor	1	3	3	4	5
1 motor	0.33	1	2	3	4
2 motor	0.33	0.50	1	1	3
3 motor	0.25	0.33	1	1	3
> 3 motor	0.20	0.25	0.33	0.33	1
JUMLAH	2.12	5.08	7.33	9.33	16

SUB KRITERIA JUMLAH MOBIL (C12)			
	0 mobil	1 mobil	> 1 mobil
0 mobil	1	2	3
1 mobil	0.5	1	2
> 1 mobil	0.33	0.5	1
JUMLAH	1.83	3.5	6

SUB KRITERIA PROGRAM BEASISWA (C13)		
	BIDIK MISI	LAINNYA
BIDIK MISI	1	1
LAINNYA	1	1
JUMLAH	2	2

Kediri, 23 September 2015
Mengetahui,
Ketua Harian Lembaga GNOTA
Kab. Kediri



DR. Mokh. Muhsin, M.Pd
NIP.



Lampiran I Struktur Hierarki AHP

KLASIFIKASI PENERIMA BEASISWA					
LEVEL	NAMA KRITERIA			SUB KRITERIA	
LEVEL 1	Penghasilan (C1)			P1 sd P5	
	Jumlah Tanggungan (C2)			P6 sd P8	
	Pekerjaan (C3)			P9 sd P13	
	Daya Listrik (C4)			P14 sd P19	
	Biaya Listrik (C5)			P20 sd P24	
	Jumlah Rumah (C6)			P25 sd P27	
	Status Rumah (C7)			P28 sd P30	
	Kondisi Dinding (C8)			P31 sd P33	
	Kondisi Lantai (C9)			P34 sd P36	
	PBB (C10)			P37 sd P40	
	Jumlah Motor (C11)			P41 sd P45	
	Jumlah Mobil (C12)			P46 sd P48	
	Program Beasiswa (C13)			P49 sd 50	
LEVEL 2					
Rp.0	P1	1300 W	P18	Tegel	P35
Rp.1 sd 500.000	P2	2200 W	P19	Keramik	P36
Rp.500.001 sd 1.000.000	P3	Rp.0	P20	Rp.0 sd 50.00	P37
Rp.1000.001 sd 2.000.000	P4	Rp.1 sd 50.000	P21	Rp.50.001 sd 100.000	P38
> Rp.2.000.000	P5	Rp.50.001 sd Rp.100.000	P22	Rp.100.001 sd 200.000	P39
> 5	P6	Rp.100.001 sd 150.000	P23	> Rp.200.000	P40
3 sd 5	P7	> Rp.150.000	P24	0	P41
0 sd 2	P8	0	P25	1	P42
Tidak Bekerja	P9	1	P26	2	P43
Buruh	P10	> 1	P27	3	P44
Karyawan	P11	Kontrak	P28	> 3	P45
Wiraswasta	P12	Numpang	P29	0	P46
Pengusaha	P13	Milik Sendiri	P30	1	P47
0 W	P14	Bambu	P31	> 1	P48
220 W	P15	Kayu	P32	Bidik Misi	P49
450 W	P16	Tembok	P33	Lainnya	P50
900 W	P17	Tanah	P34		



Lampiran J Data Penerima Beasiswa GNOTA Kediri 2015

NO	NO PESERTA	NAMA	TOTAL PENGHASILAN	JUMLAH TANGGUNGAN	PEKERJAAN	DAYA LISTRIK	BIAYA LISTRIK	JUMLAH RUMAH	STATUS RUMAH	STATUS DINDING	STATUS LANTAI	PBB	JML MOTOR	JML MOBIL	PROGRAM BEASISWA	KLASIFIKASI
1	211	WULAN PUJI UTAMI	1.000.000	1	tidak bekerja	450 W	25.991	1	Numpang	Tembok	Semen	25.536	1	0	LAINNYA	A
2	174	SITI KHIRROTUL A'YUN	0	4	Buruh Tani	450 W	38.792	1	Rumah Sendiri	Tembok	Keramik	27.656	0	0	BIDIK MISI	A
3	187	MUHAMMAD DAMANURI	0	2	tidak bekerja	450 W	40.000	1	Rumah Sendiri	Tembok	Semen	14.471	1	0	LAINNYA	A
4	135	RAHMAD INDRIANTO	400.000	1	tidak bekerja	450 W	21.000	1	Rumah Sendiri	Tembok	Keramik	30.000	1	0	BIDIK MISI	A
5	137	FITRI F	1.000.000	2	Buruh Tani	450 W	50.580	0	Rumah Sendiri	Tembok	Semen	98.000	0	0	LAINNYA	A
6	156	SEKAR PRATIWI S P	0	3	Pedagang	450 W	29.500	1	Numpang	Tembok	Semen	38.586	0	0	BIDIK MISI	A
7	214	RIRIN FAULINA	800.000	1	Tidak Bekerja	450 W	30.000	1	Rumah Sendiri	Tembok	Semen	30.720	1	0	BIDIK MISI	A
8	169	YUSRON AMAN N	2.000.000	2	Jualan grabah	450 W	25.000	1	Rumah Sendiri	Papan Kayu	Semen	35.432	1	0	BIDIK MISI	A
9	273	MIA AGUSTINA	1.000.000	2	Buruh ternak	450 W	38.000	1	Rumah Sendiri	Tembok	Tanah	40.000	1	0	BIDIK MISI	A
10	170	DINDA MAULYDIA I	0	3	KSP	900 W	32.500	1	Kontrak	Tembok	Semen	32.042	0	0	BIDIK MISI	A
11	163	AYUNING DYAH SUSANTI	0	2	tidak bekerja	450 W	33.614	1	Rumah Sendiri	Tembok	Tegel	100.507	0	0	BIDIK MISI	A
12	168	NAILIL HUSNIAH	0	3	tidak bekerja	450 W	45.000	1	Rumah Sendiri	Tembok	Semen	112.000	1	0	BIDIK MISI	A
13	254	ABDUL LATIF	500.000	1	tidak bekerja	450 W	21.000	1	Rumah Sendiri	Tembok	Keramik	92.408	0	0	LAINNYA	A

14	227	ASNA NURHAYATI	500.000	3	Buruh Tani	450 W	28.000	1	Rumah Sendiri	Tembok	Semen	9.720	1	0	BIDIK MISI	A
15	140	ALVIN NAFIAH	1.350.000	3	Buruh Tani	450 W	14.000	1	Rumah Sendiri	Tembok	Keramik	18.000	1	0	BIDIK MISI	A
16	271	HIDAYATURROHMANN	2.000.000	5	Buruh Tani	450 W	30.000	1	Rumah Sendiri	Tembok	Semen	64.089	1	0	LAINNYA	A
17	152	NOVA RINELDA A F	2.000.000	2	Buruh Tani	450 W	39.059	1	Rumah Sendiri	Tembok	Tanah	22.165	1	0	BIDIK MISI	A
18	206	MAULANA MUJTABA	0	3	Buruh Tani	450 W	50.000	1	Rumah Sendiri	Papan Kayu	Semen	37.800	1	0	LAINNYA	A
19	138	SITI KUMALA DEWI	0	1	tidak bekerja	450 W	126.000	1	Rumah Sendiri	Tembok	Keramik	29.000	1	0	LAINNYA	A
20	274	OKKI PUTRA PERDANA	800.000	1	TIDAK BEKERJA	900 W	24.306	1	Numpang	Tembok	Keramik	19.200	1	0	LAINNYA	A
21	203	KHOIRUL LIHAYATI	1.000.000	2	Buruh Tani	450 W	39.000	1	Rumah Sendiri	Tembok	Keramik	46.844	1	0	BIDIK MISI	A
22	162	GEFI INDAH P A	400.000	2	Karyawan Toko	450 W	19.200	1	Rumah Sendiri	Tembok	Semen	41.180	0	0	BIDIK MISI	A
23	228	MUHAMMAD ADIB SULAIMAN	750.000	3	Buruh Tani	450 W	27.000	1	Numpang	Papan Kayu	Semen	153.152	1	0	BIDIK MISI	A
24	246	WINDA FIRDAYANTI	1.300.000	1	Tukang Bangunan	450 W	69.000	1	Rumah Sendiri	Tembok	Semen	25.088	1	0	BIDIK MISI	A
25	253	DEVIANA INDAH PERMATAN	0	3	Karwayan Swasta	450 W	40.000	1	Numpang	Bambu	Keramik	17.920	1	0	BIDIK MISI	A
26	167	AULIA NUR S	0	4	tidak bekerja	900 W	36.337	1	Rumah Sendiri	Tembok	Semen	44.480	0	0	BIDIK MISI	A
27	197	SAHRUL MUNIR	1.000.000	2	Kuli Pikul	450 W	16.363	1	Rumah Sendiri	Tembok	Keramik	56.734	2	0	LAINNYA	A
28	153	YUNIS ROISATUL M	1.000.000	2	Buruh	450 W	53.530	1	Rumah Sendiri	Tembok	Semen	156.737	2	0	LAINNYA	A

29	157	MUMDIYAH	800.000	1	Buruh Tani	900 W	90.000	1	Rumah Sendiri	Tembok	Keramik	34.739	0	0	LAINNYA	A
30	218	MACHICA ROUFUN N	0	4	Makelar	450 W	29.300	1	Numpang	Tembok	Semen	76.176	1	0	BIDIK MISI	A
31	217	PUTRI NUR M	0	3	Tidak Bekerja	450 W	53.000	1	Rumah Sendiri	Tembok	Keramik	57.242	2	0	BIDIK MISI	A
32	245	GUNTUR TRI WASONO	800.000	2	tidak bekerja	900 W	26.000	1	Rumah Sendiri	Tembok	Keramik	32.558	1	0	BIDIK MISI	A
33	216	AHMAD IRFAN FADRI	500.000	1	Buruh Tani	450 W	27.000	1	Rumah Sendiri	Tembok	Tegel	57.808	1	0	BIDIK MISI	A
34	248	AWIATUL MUTHOWIYAH	1.000.000	2	Buruh Tani	450 W	30.000	1	Rumah Sendiri	Tembok	Tegel	126.200	1	0	LAINNYA	A
35	139	NOER ZARINA	0	3	Buruh Tani	450 W	50.000	1	Rumah Sendiri	Tembok	Keramik	40.000	1	0	BIDIK MISI	A
36	133	RATIH DYAH T R	850.000	1	tidak bekerja	450 W	33.248	1	Rumah Sendiri	Tembok	Keramik	50.000	1	0	LAINNYA	A
37	225	EVA DIAH AYUNINGTYAS	1.000.000	1	Buruh Tani	450 W	63.562	1	Rumah Sendiri	Tembok	Semen	100.000	1	0	BIDIK MISI	A
38	232	HERWIN AGUSTYANILLA SARI	900.000	2	Buruh Tani	450 W	58.600	1	Rumah Sendiri	Bambu	Keramik	45.386	2	0	LAINNYA	A
39	269	MOCH KRESNA ALVIAN P	1.000.000	3	Supir	450 W	50.000	1	Kontrak	Tembok	Semen	40.435	2	0	BIDIK MISI	A
40	180	M IQBAL BAIHAQI	1.000.000	2	Konstruksi	450 W	37.159	1	Rumah Sendiri	Tembok	Keramik	118.164	1	0	LAINNYA	A
41	178	NELLY NUR H	0	1	Karwayan Swasta	450 W	83.912	1	Rumah Sendiri	Tembok	Keramik	10.320	2	0	BIDIK MISI	A
42	182	INDIKA AZHARY	0	4	tidak bekerja	900 W	75.488	1	Rumah Sendiri	Tembok	Keramik	121.000	1	0	BIDIK MISI	A
43	200	INTAN PUTRI RAHAYU	1.000.000	1	tidak bekerja	900 W	16.000	1	Rumah Sendiri	Tembok	Tegel	30.336	3	0	LAINNYA	A

44	159	MOCH FAIZIN	1.000.000	2	Buruh	450 W	30.000	1	Rumah Sendiri	Tembok	Semen	256.295	1	0	LAINNYA	A
45	231	INDI DIAS C	700.000	2	Buruh Tani	450 W	23.000	1	Rumah Sendiri	Tembok	Keramik	83.574	1	0	BIDIK MISI	A
46	243	LAILY FITRIA NINGTYAS	900.000	1	Buruh Tani	450 W	50.000	1	Rumah Sendiri	Tembok	Keramik	68.090	2	0	BIDIK MISI	A
47	258	LISA L SHOLIKHAH	700.000	3	Kuli Listrik	900 W	21.500	1	Rumah Sendiri	Tembok	Keramik	31.920	1	0	BIDIK MISI	A
48	247	LINDA PURNAMASARI	1.400.000	1	karyawan BPR	450 W	96.000	1	Rumah Sendiri	Bambu	Keramik	126.106	1	0	LAINNYA	A
49	230	SHEyla CHABIBAH	1.200.000	3	Makelar	900 W	23.456	1	Numpang	Tembok	Keramik	45.000	1	0	BIDIK MISI	A
50	221	IKA VIKNI NAWANG RISMA	1.000.000	1	Kuli Bangunan	450 W	28.000	1	Rumah Sendiri	Tembok	Semen	75.806	1	0	BIDIK MISI	A
51	251	ALFIRA SARI	1.300.000	1	Buruh	450 W	23.000	1	Rumah Sendiri	Tembok	Semen	83.290	2	0	BIDIK MISI	A
52	236	MUHAMMAD ALI M	1.000.000	3	Buruh Tani	450 W	45.593	1	Rumah Sendiri	Tembok	Tegel	190.980	1	0	BIDIK MISI	A
53	87	ZUMROTUS SHOLIKHAH	700.000	2	Buruh Tani	450 W	59.200	1	Rumah Sendiri	Tembok	Keramik	38.475	1	0	LAINNYA	A
54	266	LAILIL AFIF AGUSTINA	750.000	1	Buruh Tani	450 W	23.000	1	Rumah Sendiri	Tembok	Semen	53.050	1	0	BIDIK MISI	A
55	234	ITA DEWI FRANSISKA	1.500.000	1	Supir	450 W	31.300	1	Numpang	Tembok	Keramik	80.733	1	0	BIDIK MISI	A
56	89	MIRA SAGOLA	1.200.000	2	Tukang Las	450 W	35.000	1	Rumah Sendiri	Tembok	Keramik	49.600	1	0	BIDIK MISI	A
57	144	DEWI MARSITOH	0	2	Pedagang Ayam	450 W	82.356	1	Numpang	Tembok	Keramik	70.000	1	0	BIDIK MISI	A
58	196	AISYAH NESTRIA AL-FALAH R	400.000	2	Karyawan Swasta	900 W	80.000	1	Numpang	Tembok	Keramik	254.892	2	0	LAINNYA	A

59	66	M. ILHAM R KURNIAWAN	500.000	2	Buruh Tani	900 W	40.588	1	Rumah Sendiri	Tembok	Keramik	26.100	2	0	LAINNYA	A
60	215	HELLEN FINA R S D	1.000.000	3	Supir	900 W	41.000	1	Rumah Sendiri	Tembok	Semen	121.231	1	0	LAINNYA	A
61	173	SETYA DWI A	1.000.000	1	Kuli Bangunan	450 W	40.000	1	Rumah Sendiri	Tembok	Semen	132.172	2	0	LAINNYA	A
62	86	M. KHOIRUL MUNA	600.000	3	Buruh Tani	450 W	36.000	1	Rumah Sendiri	Tembok	Keramik	99.120	2	0	LAINNYA	B
63	256	DIFA ZUHROTUL FIKIRIYAH	600.000	2	Buruh Tani	900 W	40.000	1	Rumah Sendiri	Tembok	Keramik	54.586	1	0	BIDIK MISI	B
64	166	WAHYA LYN FIZA	1.000.000	1	Buruh	900 W	36.000	1	Rumah Sendiri	Tembok	Keramik	100.708	1	0	LAINNYA	B
65	226	ASNA REDAYANI	1.000.000	2	Kuli Batu	900 W	60.000	1	Rumah Sendiri	Tembok	Keramik	126.800	1	0	LAINNYA	B
66	194	MOHAMMAD KHOIRUZ ZAAD	900.000	2	Buruh	900 W	60.000	1	Rumah Sendiri	Tembok	Keramik	47.312	1	0	LAINNYA	B
67	224	AGUS ZAINAL ARIFIN	1.000.000	2	Supir	450 W	31.614	1	Rumah Sendiri	Tembok	Keramik	46.437	2	0	LAINNYA	B
68	264	MUHAMMAD MUKAINUL H	1.500.000	1	Pracangan	450 W	50.000	1	Rumah Sendiri	Tembok	Semen	118.536	1	0	BIDIK MISI	B
69	165	BAGUS PRIBADI	750.000	4	Jualan Bakso	900 W	29.400	1	Rumah Sendiri	Tembok	Semen	41.202	2	0	LAINNYA	B
70	250	RIKA RAHAYU	1.500.000	1	Buruh Tani	450 W	41.000	1	Rumah Sendiri	Tembok	Tegel	621.224	1	0	LAINNYA	B
71	241	SYAMSUL ARI WIDODO	1.500.000	1	Kuli Bangunan	450 W	65.700	1	Rumah Sendiri	Tembok	Keramik	79.740	1	0	BIDIK MISI	B
72	192	ADELIA DWI SAPUTRI	600.000	2	Supir	450 W	56.700	1	Rumah Sendiri	Tembok	Keramik	48.082	1	0	BIDIK MISI	B
73	220	PRIMA NUR ZAKI	1.000.000	3	Jualan Soto	450 W	27.625	1	Sewa	Tembok	Semen	18.560	3	0	LAINNYA	B

74	189	DEWI EKA MULIANA	1.000.000	2	Pedagang	450 W	67.000	1	Rumah Sendiri	Tembok	Keramik	50.945	2	0	BIDIK MISI	B
75	160	DWI SAKINAH AINUR R	700.000	3	Buruh	900 W	100.000	1	Rumah Sendiri	Tembok	Tegel	195.820	1	0	LAINNYA	B
76	67	ILHAM ARFIANSYAH	1.000.000	1	Pedagang Kecil	900 W	100.000	1	Rumah Sendiri	Tembok	Tegel	114.300	1	0	LAINNYA	B
77	267	YUSUF	1.500.000	1	Makelar	900 W	14.742	1	Numpang	Tembok	Keramik	83.000	1	0	BIDIK MISI	B
78	229	NURFIKKA T F	1.000.000	2	Kuli Bangunan	450 W	30.000	1	Rumah Sendiri	Tembok	Keramik	90.825	2	0	LAINNYA	B
79	263	NERY DWI EVITHELISARI	1.000.000	1	Kuli Bangunan	450 W	47.060	1	Rumah Sendiri	Tembok	Keramik	113.147	1	0	LAINNYA	B
80	242	ERNA RAHAYU	1.000.000	2	Petani	450 W	50.000	1	Rumah Sendiri	Tembok	Keramik	189.712	2	0	LAINNYA	B
81	85	MOCH ZAENAL A	1.500.000	1	Buruh Tani	900 W	25.000	1	Rumah Sendiri	Tembok	Tegel	57.613	2	0	BIDIK MISI	B
82	65	ANI RAHAYU	750.000	1	Kuli Bangunan	900 W	51.000	1	Rumah Sendiri	Tembok	Keramik	111.740	0	0	BIDIK MISI	B
83	272	MOCHMMAD ABDUL GHOFUR	962.000	2	Buruh Pabrik	900 W	49.000	1	Rumah Sendiri	Tembok	Keramik	39.430	2	0	BIDIK MISI	B
84	268	FARADHYBA RIZKY R	1.000.000	1	karyawan BPR	450 W	68.000	1	Rumah Sendiri	Tembok	Keramik	70.319	2	0	LAINNYA	B
85	188	INTAN ARIKA SARI	1.500.000	1	Pedagang	450 W	50.000	1	Rumah Sendiri	Tembok	Keramik	46.264	1	0	BIDIK MISI	B
86	199	LAILATUL HIDAYAH	1.000.000	2	Swasta	900 W	30.000	1	Rumah Sendiri	Tembok	Keramik	5.000	1	0	LAINNYA	B
87	249	RAHMAT HIDAYAH	100.000	1	Jualan sayur	900 W	45.000	1	Rumah Sendiri	Tembok	Keramik	25.250	1	0	BIDIK MISI	B
88	210	MILA GHURROTUL JANNAH	1.000.000	4	Pedagang Ikan	900 W	120.800	1	Rumah Sendiri	Tembok	Keramik	64.400	2	1	BIDIK MISI	B

89	177	INNA FAUZIAH	800.000	3	Buruh	900 W	60.000	1	Rumah Sendiri	Tembok	Keramik	16.487	2	0	BIDIK MISI	B
90	68	LAYLA RAHAYU	1.000.000	1	Karyawan	900 W	60.000	1	Rumah Sendiri	Tembok	Keramik	37.943	2	0	LAINNYA	B
91	132	EMILIA SUSANTI	1.000.000	2	Wiraswasta	450 W	95.000	1	Rumah Sendiri	Tembok	Keramik	82.000	1	0	LAINNYA	B
92	142	MUSTIKA ANGGI P	1.000.000	3	Wiraswasta	900 W	43.693	1	Rumah Sendiri	Tembok	Keramik	25.000	2	0	BIDIK MISI	B
93	179	ZAHRO NUR HELIZA	1.000.000	2	Buruh Tani	450 W	55.000	1	Rumah Sendiri	Tembok	Keramik	155.384	2	0	LAINNYA	B
94	255	PUTRA RAMA TITOR	500.000	1	Kuli Bangunan	450 W	39.000	1	Rumah Sendiri	Tembok	Keramik	92.920	2	0	LAINNYA	B
95	257	BAHDYA RICHANA PUTRI	0	3	Buruh Tani	900 W	130.739	1	Rumah Sendiri	Tembok	Keramik	68.206	2	0	LAINNYA	B
96	204	BAITI KHARISMA SANTI	1.500.000	1	tidak bekerja	900 W	65.000	1	Rumah Sendiri	Tembok	Keramik	73.200	1	0	LAINNYA	B
97	233	EFI JULIANA	900.000	2	Serabutan	450 W	31.000	1	Rumah Sendiri	Tembok	Keramik	27.645	2	0	LAINNYA	B
98	265	NOVA WANDA KARTIKA	1.000.000	2	Supir	450 W	50.000	1	Numpang	Tembok	Keramik	114.400	1	0	LAINNYA	C
99	183	IIK FITRIATUS SANAH	2.000.000	1	Swasta	450 W	67.000	1	Rumah Sendiri	Tembok	Keramik	5.000	1	0	LAINNYA	C
100	185	NILA PUTRI ANGGRAINI	400.000	2	Konstruksi	900 W	125.000	1	Rumah Sendiri	Tembok	Keramik	43.873	3	0	LAINNYA	C
101	270	TOFA KUMALA SARI	1.200.000	3	Becak	900 W	40.000	1	Rumah Sendiri	Tembok	Keramik	233.000	1	0	LAINNYA	C
102	401	RINA RAHAYU N	1.000.000	1	Wiraswasta	450 W	37.426	1	Rumah Sendiri	Tembok	Keramik	173.000	1	0	LAINNYA	C
103	195	DEWI NABILA HIKMAH	2.000.000	3	Pedagang Tempe	450 W	60.000	1	Rumah Sendiri	Tembok	Keramik	230.000	2	0	LAINNYA	C

104	69	M. BAGUS ADE P	1.500.000	3	Pedagang Roti	900 W	130.000	1	Rumah Sendiri	Tembok	Keramik	43.578	1	0	LAINNYA	C
105	70	DESI TRI W	1.200.000	3	Tukang Las	1.300 W	35.000	1	Rumah Sendiri	Tembok	Tegel	43.576	1	0	LAINNYA	C
106	90	FIRDA LAILA S	1.500.000	3	Guru swasta	450 W	151.430	1	Rumah Sendiri	Tembok	Keramik	75.642	2	0	LAINNYA	C
107	201	AMITORUL MA;RIFAH	2.500.000	1	Wiraswasta	900 W	22.000	1	Rumah Sendiri	Tembok	Keramik	29.500	0	0	BIDIK MISI	C
108	400	M MUSYafa YAZIO	0	2	Petani	450 W	114.200	1	Rumah Sendiri	Tembok	Tegel	167.000	2	0	BIDIK MISI	C
109	205	NUR HALIMAH	2.000.000	1	Buruh Tani	450 W	94.257	1	Rumah Sendiri	Tembok	Keramik	182.100	1	0	LAINNYA	C
110	198	WIDYA OCTAVIANI RIDHO	1.500.000	3	Peternak	900 W	110.000	1	Rumah Sendiri	Tembok	Keramik	54.838	1	0	LAINNYA	C
111	91	AMALIYA AYU ALFINA	1.000.000	1	Wiraswasta	900 W	25.991	1	Rumah Sendiri	Tembok	Keramik	87.652	2	0	LAINNYA	C

Lampiran K Dataset

NO	NP	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30	P31	P32	P33	P34	P35	P36	P37	P38	P39	P40	P41	P42	P43	P44	P45	P46	P47	P48	P49	P50	KL			
1	211	0	4	0	0	0	0	0	1	5	0	0	0	0	0	0	4	0	0	0	0	0	4	0	0	0	0	2	0	3	0	0	0	0	1	0	2	0	4	0	0	0	0	4	0	0	0	0	3	0	0	0	0	1	A
2	174	0	4	0	0	0	0	2	0	0	4	0	0	0	0	0	4	0	0	0	0	4	0	0	0	0	2	0	0	0	1	0	0	1	0	0	1	4	0	0	0	5	0	0	0	0	3	0	0	2	0	A			
3	187	0	4	0	0	0	0	0	1	5	0	0	0	0	0	0	4	0	0	0	0	4	0	0	0	0	2	0	0	0	1	0	0	1	0	2	0	4	0	0	0	0	4	0	0	0	3	0	0	2	0	A			
4	135	0	4	0	0	0	0	0	1	5	0	0	0	0	0	0	4	0	0	0	0	4	0	0	0	0	2	0	0	0	1	0	0	1	0	0	1	4	0	0	0	0	4	0	0	0	3	0	0	2	0	A			
5	137	0	4	0	0	0	0	0	1	0	4	0	0	0	0	0	4	0	0	0	0	4	0	0	0	0	4	0	0	0	1	0	0	1	0	2	0	0	3	0	0	5	0	0	0	0	3	0	0	0	1	A			
6	156	0	4	0	0	0	0	0	1	0	0	0	2	0	0	0	4	0	0	0	0	0	3	0	0	0	2	0	3	0	0	0	0	1	0	2	0	4	0	0	0	5	0	0	0	0	3	0	0	2	0	A			
7	214	0	0	3	0	0	0	0	1	5	0	0	0	0	0	0	4	0	0	0	0	4	0	0	0	0	2	0	0	0	1	0	0	1	0	2	0	4	0	0	0	0	4	0	0	0	3	0	0	2	0	A			
8	169	0	0	3	0	0	0	0	1	0	4	0	0	0	0	0	4	0	0	0	0	4	0	0	0	0	2	0	0	0	1	0	2	0	0	2	0	4	0	0	0	0	4	0	0	0	3	0	0	2	0	A			
9	273	0	0	3	0	0	0	0	1	0	4	0	0	0	0	0	4	0	0	0	0	4	0	0	0	0	2	0	0	0	1	0	0	1	3	0	0	4	0	0	0	0	4	0	0	0	3	0	0	2	0	A			
10	170	0	0	3	0	0	0	0	1	0	0	3	0	0	0	0	4	0	0	0	0	4	0	0	0	0	2	0	3	0	0	0	0	1	0	2	0	4	0	0	0	5	0	0	0	0	3	0	0	2	0	A			

1 1	1 6 3	0	4	0	0	0	0	0	1	5	0	0	0	0	0	0	4	0	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	2	0	0	0	2	0	5	0	0	0	0	3	0	0	0	1	A		
1 2	1 6 8	0	4	0	0	0	0	0	1	5	0	0	0	0	0	0	4	0	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	2	0	0	0	2	0	0	4	0	0	0	3	0	0	2	0	A		
1 3	2 5 4	0	4	0	0	0	0	0	1	5	0	0	0	0	0	0	4	0	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	0	1	0	3	0	0	5	0	0	0	0	3	0	0	0	1	A		
1 4	2 2 7	0	4	0	0	0	0	0	1	0	4	0	0	0	0	0	4	0	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	2	0	4	0	0	0	0	4	0	0	0	3	0	0	0	1	A		
1 5	1 4 0	0	4	0	0	0	0	0	1	0	4	0	0	0	0	0	4	0	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	0	1	4	0	0	0	0	4	0	0	0	3	0	0	2	0	A		
1 6	2 7 1	0	0	3	0	0	0	2	0	0	4	0	0	0	0	0	4	0	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	2	0	0	3	0	0	0	4	0	0	0	3	0	0	2	0	A		
1 7	1 5 2	0	0	3	0	0	0	0	1	0	4	0	0	0	0	0	4	0	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	3	0	0	4	0	0	0	0	4	0	0	0	3	0	0	0	1	A		
1 8	2 0 6	0	0	0	2	0	0	0	1	0	4	0	0	0	0	0	4	0	0	0	0	4	0	0	0	2	0	0	0	1	0	2	0	0	2	0	4	0	0	0	0	4	0	0	0	3	0	0	2	0	A		
1 9	1 3 8	0	0	3	0	0	0	0	1	5	0	0	0	0	0	0	4	0	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	0	1	4	0	0	0	0	4	0	0	0	3	0	0	0	1	A		
2 0	2 7 4	0	0	3	0	0	0	0	1	5	0	0	0	0	0	0	3	0	0	0	0	0	0	0	2	0	0	2	0	3	0	0	0	0	1	0	0	1	4	0	0	0	0	4	0	0	0	3	0	0	2	0	A
2 1	2 0 3	0	0	3	0	0	0	0	1	0	4	0	0	0	0	0	4	0	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	0	1	4	0	0	0	0	4	0	0	0	3	0	0	2	0	A		
2 2	1 6 2	0	0	3	0	0	0	0	1	0	0	3	0	0	0	0	4	0	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	2	0	4	0	0	0	5	0	0	0	0	3	0	0	0	1	A		

23	2	0	0	0	2	0	0	0	1	0	4	0	0	0	0	0	4	0	0	0	0	2	0	3	0	0	0	2	0	0	2	0	0	0	0	0	3	0	0	0	1	A									
24	2	0	0	0	2	0	0	0	1	0	4	0	0	0	0	0	4	0	0	0	0	2	0	0	0	1	0	0	1	0	2	0	4	0	0	0	0	4	0	0	0	3	0	0	2	0	A				
25	2	0	0	0	2	0	0	0	1	0	0	3	0	0	0	0	4	0	0	0	0	3	0	0	0	2	0	3	0	0	3	0	0	0	0	1	4	0	0	0	0	4	0	0	0	3	0	0	0	1	A
26	1	0	0	0	0	0	2	0	5	0	0	0	0	0	0	0	3	0	0	0	4	0	0	0	0	2	0	0	0	1	0	0	1	0	2	0	4	0	0	0	5	0	0	0	0	3	0	0	2	0	A
27	1	5	0	0	0	0	0	1	0	4	0	0	0	0	0	4	0	0	0	0	4	0	0	0	0	2	0	0	0	1	0	0	1	0	0	1	0	3	0	0	0	0	3	0	0	3	0	0	0	1	A
28	1	0	4	0	0	0	0	1	0	4	0	0	0	0	0	4	0	0	0	0	4	0	0	0	0	2	0	0	0	1	0	0	1	0	2	0	0	0	2	0	0	0	3	0	0	3	0	0	2	0	A
29	1	0	4	0	0	0	0	1	0	4	0	0	0	0	0	0	3	0	0	0	0	3	0	0	0	2	0	0	0	1	0	0	1	0	0	1	4	0	0	0	5	0	0	0	0	3	0	0	0	1	A
30	2	0	0	3	0	0	2	0	0	0	0	2	0	0	0	4	0	0	0	0	0	3	0	0	0	2	0	3	0	0	0	0	1	0	2	0	0	3	0	0	0	4	0	0	0	3	0	0	0	1	A
31	2	0	0	3	0	0	0	1	5	0	0	0	0	0	0	4	0	0	0	0	4	0	0	0	0	2	0	0	0	1	0	0	1	0	0	1	0	3	0	0	0	0	3	0	0	3	0	0	2	0	A
32	2	0	0	3	0	0	0	1	5	0	0	0	0	0	0	3	0	0	0	0	0	3	0	0	0	2	0	0	0	1	0	0	1	0	0	1	4	0	0	0	0	4	0	0	0	3	0	0	2	0	A
33	2	0	0	3	0	0	0	1	0	4	0	0	0	0	0	4	0	0	0	0	4	0	0	0	0	2	0	0	0	1	0	0	1	0	2	0	0	3	0	0	0	4	0	0	0	3	0	0	0	1	A
34	2	0	0	3	0	0	0	1	0	4	0	0	0	0	0	4	0	0	0	0	4	0	0	0	0	2	0	0	0	1	0	0	1	0	2	0	0	0	2	0	0	4	0	0	0	3	0	0	2	0	A

35	139	0	0	3	0	0	0	0	1	0	4	0	0	0	0	4	0	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	0	1	4	0	0	0	0	4	0	0	0	3	0	0	0	1	A	
36	133	0	0	0	2	0	0	0	1	5	0	0	0	0	0	4	0	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	0	1	4	0	0	0	0	4	0	0	0	3	0	0	0	1	A	
37	225	0	0	0	2	0	0	0	1	0	4	0	0	0	0	4	0	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	2	0	0	3	0	0	0	4	0	0	0	3	0	0	2	0	A	
38	232	0	0	0	2	0	0	0	1	0	4	0	0	0	0	4	0	0	0	0	3	0	0	0	2	0	0	0	1	3	0	0	0	0	1	4	0	0	0	0	3	0	0	3	0	0	2	0	A		
39	269	0	0	0	2	0	0	0	1	0	0	3	0	0	0	4	0	0	0	0	3	0	0	0	2	0	3	0	0	0	0	1	0	2	0	4	0	0	0	0	3	0	0	3	0	0	2	0	A		
40	180	0	4	0	0	0	0	0	1	0	4	0	0	0	0	4	0	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	0	1	0	0	2	0	0	4	0	0	0	3	0	0	0	1	A	
41	178	0	4	0	0	0	0	0	1	0	0	3	0	0	0	4	0	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	0	1	4	0	0	0	0	3	0	0	3	0	0	0	1	A		
42	182	0	0	3	0	0	0	2	0	5	0	0	0	0	0	0	3	0	0	0	0	3	0	0	0	2	0	0	0	1	0	0	1	0	0	1	0	0	2	0	0	4	0	0	0	3	0	0	2	0	A
43	200	0	0	3	0	0	0	0	1	5	0	0	0	0	0	0	3	0	0	0	0	3	0	0	0	2	0	0	0	1	0	0	1	0	2	0	4	0	0	0	0	0	2	0	3	0	0	2	0	A	
44	159	0	0	3	0	0	0	0	1	0	4	0	0	0	0	4	0	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	2	0	0	0	0	1	0	4	0	0	0	3	0	0	2	0	A	
45	231	0	0	3	0	0	0	0	1	0	4	0	0	0	0	4	0	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	0	1	0	3	0	0	0	4	0	0	0	3	0	0	0	1	A	
46	243	0	0	3	0	0	0	0	1	0	4	0	0	0	0	4	0	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	0	1	0	3	0	0	0	0	3	0	0	3	0	0	2	0	A	

47	258	0	0	3	0	0	0	0	1	0	4	0	0	0	0	0	0	3	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	0	1	4	0	0	0	0	4	0	0	0	3	0	0	0	1	A	
48	247	0	0	3	0	0	0	0	1	0	0	3	0	0	0	0	4	0	0	0	0	4	0	0	0	2	0	0	0	1	3	0	0	0	0	1	0	0	2	0	0	4	0	0	0	3	0	0	0	1	A	
49	230	0	0	3	0	0	0	0	1	0	0	0	2	0	0	0	0	3	0	0	0	0	3	0	0	2	0	3	0	0	0	1	0	0	1	4	0	0	0	0	4	0	0	0	3	0	0	2	0	A		
50	221	0	0	0	2	0	0	0	1	0	4	0	0	0	0	0	4	0	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	2	0	0	3	0	0	0	4	0	0	0	3	0	0	0	1	A	
51	251	0	0	0	2	0	0	0	1	0	4	0	0	0	0	0	4	0	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	2	0	0	3	0	0	0	0	3	0	0	3	0	0	2	0	A	
52	236	0	0	0	2	0	0	0	1	0	4	0	0	0	0	0	4	0	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	2	0	0	0	2	0	0	4	0	0	0	3	0	0	2	0	A	
53	87	0	0	0	2	0	0	0	1	0	4	0	0	0	0	0	4	0	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	0	1	4	0	0	0	0	4	0	0	0	3	0	0	0	1	A	
54	266	0	0	0	2	0	0	0	1	0	4	0	0	0	0	0	4	0	0	0	0	0	3	0	0	0	2	0	0	0	1	0	0	1	0	2	0	0	3	0	0	0	4	0	0	0	3	0	0	2	0	A
55	234	0	0	0	2	0	0	0	1	0	0	3	0	0	0	0	4	0	0	0	0	4	0	0	0	2	0	3	0	0	0	0	1	0	0	1	0	3	0	0	0	4	0	0	0	3	0	0	0	1	A	
56	89	0	0	0	2	0	0	0	1	0	0	3	0	0	0	0	4	0	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	0	1	4	0	0	0	0	4	0	0	0	3	0	0	2	0	A	
57	144	0	0	0	2	0	0	0	1	0	0	0	2	0	0	0	4	0	0	0	0	4	0	0	0	2	0	3	0	0	0	0	1	0	0	1	0	3	0	0	0	4	0	0	0	3	0	0	2	0	A	
58	196	5	0	0	0	0	0	0	1	0	0	3	0	0	0	0	0	3	0	0	0	0	3	0	0	0	2	0	3	0	0	0	0	1	0	0	1	0	0	0	1	0	0	3	0	0	3	0	0	2	0	A
59	66	0	4	0	0	0	0	0	1	0	4	0	0	0	0	0	0	3	0	0	0	0	3	0	0	0	2	0	0	0	1	0	0	1	0	0	1	4	0	0	0	0	0	3	0	0	3	0	0	0	1	A

60	2	0	4	0	0	0	0	0	1	0	0	3	0	0	0	0	3	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	2	0	0	0	2	0	0	4	0	0	0	3	0	0	0	1	A		
61	1	0	0	3	0	0	0	0	1	0	4	0	0	0	0	4	0	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	2	0	0	0	2	0	0	0	3	0	0	3	0	0	0	1	A		
62	8	0	0	3	0	0	0	0	1	0	4	0	0	0	0	4	0	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	0	1	0	3	0	0	0	0	3	0	0	3	0	0	0	1	B		
63	2	0	0	3	0	0	0	0	1	0	4	0	0	0	0	0	3	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	0	1	0	3	0	0	0	4	0	0	0	3	0	0	0	1	B		
64	1	0	0	3	0	0	0	0	1	0	4	0	0	0	0	0	3	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	0	1	0	0	2	0	0	4	0	0	0	3	0	0	2	0	B		
65	2	0	0	3	0	0	0	0	1	0	4	0	0	0	0	0	3	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	0	1	0	0	2	0	0	4	0	0	0	3	0	0	2	0	B		
66	1	0	0	3	0	0	0	0	1	0	4	0	0	0	0	0	3	0	0	0	0	3	0	0	2	0	0	0	1	0	0	1	0	0	1	4	0	0	0	0	4	0	0	0	3	0	0	0	1	B		
67	2	0	0	3	0	0	0	0	1	0	0	3	0	0	0	0	4	0	0	0	0	0	3	0	0	0	1	0	0	1	0	0	1	4	0	0	0	0	0	3	0	0	3	0	0	2	0	B				
68	2	0	0	3	0	0	0	0	1	0	0	0	2	0	0	0	4	0	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	2	0	0	0	2	0	0	4	0	0	0	3	0	0	2	0	B	
69	1	0	0	0	2	0	0	2	0	0	0	0	2	0	0	0	3	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	2	0	4	0	0	0	0	0	3	0	0	3	0	0	2	0	B		
70	2	0	0	0	2	0	0	0	1	0	4	0	0	0	0	0	4	0	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	2	0	0	0	0	1	0	4	0	0	0	3	0	0	2	0	B	
71	2	0	0	0	2	0	0	0	1	0	4	0	0	0	0	0	4	0	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	0	1	0	3	0	0	0	4	0	0	0	3	0	0	0	1	B	
72	1	0	0	0	2	0	0	0	1	0	0	3	0	0	0	0	4	0	0	0	0	0	3	0	0	0	2	0	0	0	1	0	0	1	0	0	1	4	0	0	0	0	4	0	0	0	3	0	0	2	0	B

98	265	0	0	0	0	0	0	0	0	1	0	0	3	0	0	0	0	4	0	0	0	0	4	0	0	0	0	2	0	3	0	0	0	0	1	0	0	1	0	0	2	0	0	4	0	0	0	3	0	0	0	1	C	
99	183	0	4	0	0	0	0	1	0	0	0	2	0	0	0	4	0	0	0	0	4	0	0	0	0	2	0	0	0	1	0	0	1	0	0	1	0	0	1	0	0	0	0	0	4	0	0	0	3	0	0	0	1	C
100	185	0	0	3	0	0	0	1	0	4	0	0	0	0	0	3	0	0	0	0	3	0	0	0	2	0	0	0	1	0	0	1	0	0	1	0	0	1	0	3	0	0	0	0	2	0	3	0	0	0	1	C		
101	270	0	0	3	0	0	0	1	0	4	0	0	0	0	0	3	0	0	0	0	0	2	0	0	2	0	0	0	1	0	0	1	0	0	1	0	0	1	0	0	0	1	0	4	0	0	0	3	0	0	2	0	C	
102	401	0	0	0	2	0	0	1	0	0	0	2	0	0	0	4	0	0	0	0	4	0	0	0	0	2	0	0	0	1	0	0	1	0	0	1	0	0	2	0	0	4	0	0	0	3	0	0	0	1	C			
103	195	0	0	3	0	0	0	1	0	0	0	2	0	0	0	4	0	0	0	0	4	0	0	0	0	2	0	0	0	1	0	0	1	0	0	1	0	0	1	0	0	0	1	0	0	3	0	0	3	0	0	0	1	C
104	604	0	0	0	2	0	0	1	0	0	0	0	1	0	0	0	3	0	0	0	0	3	0	0	0	2	0	0	0	1	0	0	1	0	0	1	4	0	0	0	0	4	0	0	0	3	0	0	0	1	C			
105	700	0	0	0	0	1	0	0	1	0	0	3	0	0	0	0	0	0	2	0	0	0	0	0	2	0	0	0	1	0	0	1	0	2	0	4	0	0	0	0	4	0	0	0	3	0	0	0	1	C				
106	900	0	0	0	0	1	0	0	1	0	0	0	2	0	0	0	4	0	0	0	0	4	0	0	0	2	0	0	0	1	0	0	1	0	0	1	0	3	0	0	0	0	3	0	0	3	0	0	0	1	C			
107	201	0	0	0	0	1	0	0	1	0	0	0	2	0	0	0	0	3	0	0	0	0	0	0	0	1	0	2	0	0	0	1	0	0	1	0	0	1	4	0	0	0	5	0	0	0	0	3	0	0	2	0	C	
108	400	0	0	0	0	1	0	0	1	0	0	0	0	1	0	0	4	0	0	0	0	4	0	0	0	0	2	0	0	0	1	0	0	1	0	2	0	0	0	2	0	0	0	3	0	0	3	0	0	2	0	C		
109	205	0	0	0	0	0	0	1	0	4	0	0	0	0	0	4	0	0	0	0	0	0	0	2	0	0	2	0	0	0	1	0	0	1	0	0	1	0	0	2	0	0	4	0	0	0	3	0	0	2	0	C		
110	119	0	0	0	2	0	0	0	1	0	0	0	0	1	0	0	0	3	0	0	0	0	3	0	0	0	2	0	0	0	1	0	0	1	0	0	1	0	3	0	0	0	4	0	0	0	3	0	0	0	1	C		

Lampiran L Normalisasi Data Level 1

Data Training

NP	P1	P2	P3	P4	P6	P7	P8	P9	P10	P11	P14	P15	P16	P20	P21	P25	P26	P28	P31	P34	P37	P41	P46	AC	C
211	0.1	0.7	0.1	0.1	0.1	0.1	0.5	0.9	0.1	0.1	0.1	0.1	0.9	0.1	0.9	0.1	0.6	0.9	0.1	0.1	0.9	0.1	0.9	A	1
174	0.1	0.7	0.1	0.1	0.1	0.9	0.1	0.1	0.7	0.1	0.1	0.1	0.9	0.1	0.9	0.1	0.6	0.1	0.1	0.1	0.9	0.9	0.9	A	1
187	0.1	0.7	0.1	0.1	0.1	0.1	0.5	0.9	0.1	0.1	0.1	0.1	0.9	0.1	0.9	0.1	0.6	0.1	0.1	0.1	0.9	0.1	0.9	A	1
86	0.1	0.1	0.6	0.1	0.1	0.1	0.5	0.1	0.7	0.1	0.1	0.1	0.9	0.1	0.9	0.1	0.6	0.1	0.1	0.1	0.1	0.1	0.9	B	-1
256	0.1	0.1	0.6	0.1	0.1	0.1	0.5	0.1	0.7	0.1	0.1	0.1	0.1	0.1	0.9	0.1	0.6	0.1	0.1	0.1	0.1	0.1	0.9	B	-1
166	0.1	0.1	0.6	0.1	0.1	0.1	0.5	0.1	0.7	0.1	0.1	0.1	0.1	0.1	0.9	0.1	0.6	0.1	0.1	0.1	0.1	0.1	0.9	B	-1
233	0.1	0.1	0.1	0.1	0.1	0.1	0.5	0.1	0.7	0.1	0.1	0.1	0.9	0.1	0.1	0.1	0.6	0.1	0.1	0.1	0.9	0.1	0.9	C	-1
265	0.1	0.1	0.1	0.1	0.1	0.1	0.5	0.1	0.1	0.6	0.1	0.1	0.9	0.1	0.9	0.1	0.6	0.9	0.1	0.1	0.1	0.1	0.9	C	-1
183	0.1	0.7	0.1	0.1	0.1	0.1	0.5	0.1	0.1	0.1	0.1	0.1	0.9	0.1	0.9	0.1	0.6	0.1	0.1	0.1	0.1	0.1	0.9	C	-1

Data Testing

NP	P1	P2	P3	P4	P6	P7	P8	P9	P10	P11	P14	P15	P16	P20	P21	P25	P26	P28	P31	P34	P37	P41	P46	AC	C
135	0.1	0.74	0.1	0.1	0.1	0.1	0.5	0.9	0.1	0.1	0.1	0.1	0.9	0.1	0.9	0.1	0.6	0.1	0.1	0.1	0.9	0.1	0.9	A	1
226	0.1	0.1	0.58	0.1	0.1	0.1	0.5	0.1	0.7	0.1	0.1	0.1	0.1	0.1	0.9	0.1	0.6	0.1	0.1	0.1	0.1	0.1	0.9	B	-1
185	0.1	0.1	0.58	0.1	0.1	0.1	0.5	0.1	0.7	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.6	0.1	0.1	0.1	0.1	0.1	0.9	C	-1

Lampiran M Perkalian Bobot Sub Kriteria Dengan Hasil Normalisasi Level 1

Data Training

NP	P1	P2	P3	P4	P6	P7	P8	P9	P10	P11	P14	P15	P16	P20	P21	P25	P26	P28	P31	P34	P37	P41	P46
211	0.01	0.05	0.003	0.002	0.008	0.004	0.008	0.063	0.003	0.002	0.003	0.003	0.016	0.003	0.015	0.006	0.011	0.023	0.003	0.003	0.020	0.002	0.015
174	0.01	0.05	0.003	0.002	0.008	0.033	0.002	0.007	0.024	0.002	0.003	0.003	0.016	0.003	0.015	0.006	0.011	0.003	0.003	0.003	0.020	0.015	0.015
187	0.01	0.05	0.003	0.002	0.008	0.004	0.008	0.063	0.003	0.002	0.003	0.003	0.016	0.003	0.015	0.006	0.011	0.003	0.003	0.003	0.020	0.002	0.015
86	0.01	0.007	0.017	0.002	0.008	0.004	0.008	0.007	0.024	0.002	0.003	0.003	0.016	0.003	0.015	0.006	0.011	0.003	0.003	0.003	0.002	0.002	0.015
256	0.01	0.007	0.017	0.002	0.008	0.004	0.008	0.007	0.024	0.002	0.003	0.003	0.002	0.003	0.015	0.006	0.011	0.003	0.003	0.003	0.002	0.002	0.015
166	0.01	0.007	0.017	0.002	0.008	0.004	0.008	0.007	0.024	0.002	0.003	0.003	0.002	0.003	0.015	0.006	0.011	0.003	0.003	0.003	0.002	0.002	0.015
233	0.01	0.007	0.003	0.002	0.008	0.004	0.008	0.007	0.024	0.002	0.003	0.003	0.016	0.003	0.002	0.006	0.011	0.003	0.003	0.003	0.020	0.002	0.015
265	0.01	0.007	0.003	0.002	0.008	0.004	0.008	0.007	0.003	0.012	0.003	0.003	0.016	0.003	0.015	0.006	0.011	0.023	0.003	0.003	0.002	0.002	0.015
183	0.01	0.05	0.003	0.002	0.008	0.004	0.008	0.007	0.003	0.002	0.003	0.003	0.016	0.003	0.015	0.006	0.011	0.003	0.003	0.003	0.002	0.002	0.015

Data Testing

NP	P1	P2	P3	P4	P6	P7	P8	P9	P10	P11	P14	P15	P16	P20	P21	P25	P26	P28	P31	P34	P37	P41	P46
135	0.009	0.049	0.003	0.002	0.008	0.0037	0.008	0.063	0.003	0.002	0.003	0.003	0.016	0.003	0.015	0.006	0.011	0.003	0.003	0.003	0.020	0.002	0.015
226	0.009	0.007	0.017	0.002	0.008	0.0037	0.008	0.007	0.024	0.002	0.003	0.003	0.002	0.003	0.015	0.006	0.011	0.003	0.003	0.003	0.002	0.002	0.015
185	0.009	0.007	0.017	0.002	0.008	0.0037	0.008	0.007	0.024	0.002	0.003	0.003	0.002	0.003	0.002	0.006	0.011	0.003	0.003	0.003	0.002	0.002	0.015

Lampiran N Perhitungan Kernel Polynomial Degree 2 Level 1

$K(x, y)$	211	174	187	86	256	166	233	265	183
211	0.000073	0.000022	0.000065	0.000005	0.000004	0.000004	0.000005	0.000006	0.000017
174	0.000022	0.000034	0.000021	0.000005	0.000004	0.000004	0.000006	0.000003	0.000015
187	0.000065	0.000021	0.000065	0.000004	0.000004	0.000004	0.000005	0.000004	0.000017
86	0.000005	0.000005	0.000004	0.000005	0.000004	0.000004	0.000003	0.000002	0.000003
256	0.000004	0.000004	0.000004	0.000004	0.000003	0.000003	0.000002	0.000002	0.000002
166	0.000004	0.000004	0.000004	0.000004	0.000003	0.000003	0.000002	0.000002	0.000002
233	0.000005	0.000006	0.000005	0.000003	0.000002	0.000002	0.000004	0.000002	0.000002
265	0.000006	0.000003	0.000004	0.000002	0.000002	0.000002	0.000002	0.000004	0.000003
183	0.000017	0.000015	0.000017	0.000003	0.000002	0.000002	0.000002	0.000003	0.000014

Lampiran O Normalisasi Data Level 2

Data Training

NP	P1	P2	P3	P4	P6	P7	P8	P9	P10	P11	P14	P15	P16	P20	P21	P25	P26	P28	P31	P34	P37	P41	P46	AC	C
86	0.1	0.1	0.6	0.1	0.1	0.1	0.5	0.1	0.7	0.1	0.1	0.1	0.9	0.1	0.9	0.1	0.6	0.1	0.1	0.1	0.1	0.1	0.9	B	-1
256	0.1	0.1	0.6	0.1	0.1	0.1	0.5	0.1	0.7	0.1	0.1	0.1	0.1	0.1	0.9	0.1	0.6	0.1	0.1	0.1	0.1	0.1	0.9	B	-1
166	0.1	0.1	0.6	0.1	0.1	0.1	0.5	0.1	0.7	0.1	0.1	0.1	0.1	0.1	0.9	0.1	0.6	0.1	0.1	0.1	0.1	0.1	0.9	B	-1
233	0.1	0.1	0.1	0.1	0.1	0.1	0.5	0.1	0.7	0.1	0.1	0.1	0.9	0.1	0.1	0.1	0.6	0.1	0.1	0.1	0.9	0.1	0.9	C	-1
265	0.1	0.1	0.1	0.1	0.1	0.1	0.5	0.1	0.1	0.6	0.1	0.1	0.9	0.1	0.9	0.1	0.6	0.9	0.1	0.1	0.1	0.1	0.9	C	-1
183	0.1	0.7	0.1	0.1	0.1	0.1	0.5	0.1	0.1	0.1	0.1	0.1	0.9	0.1	0.9	0.1	0.6	0.1	0.1	0.1	0.1	0.1	0.9	C	-1

Data Testing

NP	P1	P2	P3	P4	P6	P7	P8	P9	P10	P11	P14	P15	P16	P20	P21	P25	P26	P28	P31	P34	P37	P41	P46	AC	C
226	0.1	0.1	0.58	0.1	0.1	0.1	0.5	0.1	0.7	0.1	0.1	0.1	0.1	0.1	0.9	0.1	0.6	0.1	0.1	0.1	0.1	0.1	0.9	B	-1
185	0.1	0.1	0.58	0.1	0.1	0.1	0.5	0.1	0.7	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.6	0.1	0.1	0.1	0.1	0.1	0.9	C	-1

Lampiran P Perkalian Bobot Sub Kriteria Dengan Hasil Normalisasi Level 2

Data Training

NP	P1	P2	P3	P4	P6	P7	P8	P9	P10	P11	P14	P15	P16	P20	P21	P25	P26	P28	P31	P34	P37	P41	P46
86	0.01	0.007	0.017	0.002	0.008	0.004	0.008	0.007	0.024	0.002	0.003	0.003	0.016	0.003	0.015	0.006	0.011	0.003	0.003	0.003	0.002	0.002	0.015
256	0.01	0.007	0.017	0.002	0.008	0.004	0.008	0.007	0.024	0.002	0.003	0.003	0.002	0.003	0.015	0.006	0.011	0.003	0.003	0.003	0.002	0.002	0.015
166	0.01	0.007	0.017	0.002	0.008	0.004	0.008	0.007	0.024	0.002	0.003	0.003	0.002	0.003	0.015	0.006	0.011	0.003	0.003	0.003	0.002	0.002	0.015
233	0.01	0.007	0.003	0.002	0.008	0.004	0.008	0.007	0.024	0.002	0.003	0.003	0.016	0.003	0.002	0.006	0.011	0.003	0.003	0.003	0.020	0.002	0.015
265	0.01	0.007	0.003	0.002	0.008	0.004	0.008	0.007	0.003	0.012	0.003	0.003	0.016	0.003	0.015	0.006	0.011	0.023	0.003	0.003	0.002	0.002	0.015
183	0.01	0.05	0.003	0.002	0.008	0.004	0.008	0.007	0.003	0.002	0.003	0.003	0.016	0.003	0.015	0.006	0.011	0.003	0.003	0.003	0.002	0.002	0.015

Data Testing

NP	P1	P2	P3	P4	P6	P7	P8	P9	P10	P11	P14	P15	P16	P20	P21	P25	P26	P28	P31	P34	P37	P41	P46
226	0.009	0.007	0.017	0.002	0.008	0.0037	0.008	0.007	0.024	0.002	0.003	0.003	0.002	0.003	0.015	0.006	0.011	0.003	0.003	0.003	0.002	0.002	0.015
185	0.009	0.007	0.017	0.002	0.008	0.0037	0.008	0.007	0.024	0.002	0.003	0.003	0.002	0.003	0.002	0.006	0.011	0.003	0.003	0.003	0.002	0.002	0.015

Lampiran Q Perhitungan Kernel Polynomial Degree 2 Level 2

$K(x,y)$	86	256	166	233	265	183
86	0.000005	0.000004	0.000004	0.000003	0.000002	0.000003
256	0.000004	0.000003	0.000003	0.000002	0.000002	0.000002
166	0.000004	0.000003	0.000003	0.000002	0.000002	0.000002
233	0.000003	0.000002	0.000002	0.000004	0.000002	0.000002
265	0.000002	0.000002	0.000002	0.000002	0.000004	0.000003
183	0.000003	0.000002	0.000002	0.000002	0.000003	0.000014

Lampiran R Hasil Pengujian Berdasarkan Nilai Threshold AHP

Nilai Threshold	Rata-rata Akurasi	Jumlah Sub Kriteria Terpilih dari 50 Sub Kriteria	Kriteria Terpilih	Rata-Rata Waktu Eksekusi
0.001	89.94	50	P1,P2,P3,P4,P5,P6,P7,P8,P9,P10,P11,P12,P13,P14,P15,P16,P17,P18,P19,P20,P21,P22,P23,P24,P25,P26,P27,P28,P29,P30,P31,P32,P33,P34,P35,P36,P37,P38,P39,P40,P41,P42,P43,P44,P45,P46,P47,P48,P49,P50	23.89
0.0035	89.94	47	P1,P2,P3,P4,P5,P6,P7,P8,P9,P10,P11,P12,P13,P14,P15,P16,P17,P18,P19,P20,P21,P22,P23,P24,P25,P26,P27,P28,P29,P31,P32,P33,P34,P35,P36,P37,P38,P39,P41,P42,P43,P44,P46,P47,P48,P49,P50	23.84
0.01	89.94	33	P1,P2,P3,P4,P5,P6,P7,P8,P9,P10,P11,P12,P14,P15,P16,P17,P20,P21,P25,P26,P28,P31,P32,P34,P35,P37,P38,P41,P42,P46,P47,P49,P50	23.55
0.015	88.56	23	P1,P2,P3,P4,P6,P7,P8,P9,P10,P11,P14,P15,P16,P20,P21,P25,P26,P28,P31,P34,P37,P41,P46	23.33
0.02	75.94	16	P1,P2,P3,P6,P7,P9,P10,P11,P14,P15,P20,P25,P28,P31,P34,P37	23.18
0.025	79.13	14	P1,P2,P3,P6,P7,P9,P10,P14,P15,P20,P25,P28,P31,P34	23.16
0.03	77.28	13	P1,P2,P3,P6,P7,P9,P10,P14,P15,P20,P25,P31,P34	23.19
0.035	52.83	7	P1,P2,P6,P7,P9,P14,P25	23.16
0.04	49.13	5	P1,P2,P6,P9,P25	23.19
0.07	38.03	3	P1,P1,P6,P9	23.35

Lampiran S Hasil Pengujian

Hasil Perhitungan $f(x)$

Berdasarkan skenario pengujian ke-1 pada rasio 90%:10%, jumlah data *training* 101 dan jumlah data *testing* 10, nilai *threshold* AHP = 0.01, nilai parameter lamda = 0.5, nilai konstanta *gamma* = 0.003, nilai epsilon=0.001, banyak iterasi maksimum=100, nilai parameter C (*Complexity*)=1 dengan akurasi sebesar 93.10%.

Nomor Peserta	$f^1(x)$	$f^2(x)$	Hasil Klasifikasi	<i>Actual Class</i>
211	0.000568073	-	1 dilevel 1 = sangat layak	Sangat layak
187	0.000549869	-	1 dilevel 1 = sangat layak	Sangat layak
135	0.000544963	-	1 dilevel 1 = sangat layak	Sangat layak
271	0.000032013	-	1 dilevel 1 = sangat layak	Sangat layak
217	0.00028573	-	1 dilevel 1 = sangat layak	Sangat layak
232	0.0000249845	-	1 dilevel 1 = sangat layak	Sangat layak
221	0.00000314501	-	1 dilevel 1 = sangat layak	Sangat layak
196	0.0000611544	-	1 dilevel 1 = sangat layak	Layak
199	-0.0000114091	-0.00000145278	-1 dilevel 2 = tidak layak	Tidak layak
195	-0.0000122449	-0.00000164866	-1 dilevel 2 = tidak layak	Tidak layak

Nilai Bias di Level 1 = -0,00007874498987803

Nilai Bias di Level 2 = -0,00001444388726668

Berdasarkan skenario pengujian ke-2 pada rasio 90%:10%, jumlah data *training* 101 dan jumlah data *testing* 10, nilai *threshold* AHP = 0.01, nilai parameter lamda = 0.5, nilai konstanta *gamma* = 0.003, nilai epsilon=0.001, banyak iterasi maksimum=100, nilai parameter C (*Complexity*)=1 dengan akurasi sebesar 77.77%.

Nomor Peserta	$f^1(x)$	$f^2(x)$	Hasil Klasifikasi	Actual Class
137	0.000198151	-	1 dilevel 1 = sangat layak	Sangat layak
273	0.000052989	-	1 dilevel 1 = sangat layak	Sangat layak
217	0.000349165	-	1 dilevel 1 = sangat layak	Sangat layak
159	0.0000154968	-	1 dilevel 1 = sangat layak	Sangat layak
247	0.00000659152	-	1 dilevel 1 = sangat layak	Sangat layak
236	0.00000815958	-	1 dilevel 1 = sangat layak	Sangat layak
268	-0.00000746865	-0.00000258711	-1 dilevel 2 = tidak layak	Layak
132	-0.0000187085	-0.00000509619	-1 dilevel 2 = tidak layak	Layak
142	-0.0000118578	-0.00000163044	-1 dilevel 2 = tidak layak	Layak
400	-0.0000086943	-0.00000376031	-1 dilevel 2 = tidak layak	Tidak layak

Nilai Bias di Level 1 = -0,00007634828216396

Nilai Bias di Level 2 = -0,00001464540582674

Berdasarkan skenario pengujian ke-3 pada rasio 90%:10%, jumlah data *training* 101 dan jumlah data *testing* 10, nilai *threshold* AHP = 0.01, nilai parameter lamda = 0.5, nilai konstanta *gamma* = 0.003, nilai *epsilon*=0.001, banyak iterasi maksimum=100, nilai parameter C (*Complexity*)=1 dengan akurasi sebesar 93.10%.

Nomor Peserta	$f^1(x)$	$f^2(x)$	Hasil Klasifikasi	<i>Actual Class</i>
156	0.000165734	-	1 dilevel 1 = sangat layak	Sangat layak
167	0.000368295	-	1 dilevel 1 = sangat layak	Sangat layak
216	0.000012385	-	1 dilevel 1 = sangat layak	Sangat layak
236	0.00000799147	-	1 dilevel 1 = sangat layak	Sangat layak
66	0.000126127	-	1 dilevel 1 = sangat layak	Sangat layak
215	0.0000933624	-	1 dilevel 1 = sangat layak	Sangat layak
256	-0.00001357	0.00000721974	1 dilevel 2 = layak	Layak
85	-0.0000215786	0.00000434724	1 dilevel 2 = layak	Layak
199	-0.0000119122	-0.000000854162	-1 dilevel 2 = tidak layak	Layak
198	-0.0000468193	-0.00000665704	-1 dilevel 2 = tidak layak	Tidak layak

Nilai Bias di Level 1 = -0,00008317585161421

Nilai Bias di Level 2 = -0,00001371017879137

Berdasarkan skenario pengujian ke-4 pada rasio 90%:10%, jumlah data *training* 101 dan jumlah data *testing* 10, nilai *threshold* AHP = 0.01, nilai parameter lamda = 0.5, nilai konstanta *gamma* = 0.003, nilai *epsilon*=0.001, banyak iterasi maksimum=100, nilai parameter C (*Complexity*)=1 dengan akurasi sebesar 100%.

Nomor Peserta	$f^1(x)$	$f^2(x)$	Hasil Klasifikasi	Actual Class
206	0.0000388206	-	1 dilevel 1 = sangat layak	Sangat layak
162	0.0000228741	-	1 dilevel 1 = sangat layak	Sangat layak
159	0.0000155811	-	1 dilevel 1 = sangat layak	Sangat layak
243	0.0000110203	-	1 dilevel 1 = sangat layak	Sangat layak
89	0.0000128168	-	1 dilevel 1 = sangat layak	Sangat layak
196	0.000059752	-	1 dilevel 1 = sangat layak	Sangat layak
194	-0.000000205626	0.00000644481	1 dilevel 2 = layak	Layak
272	-0.00000501523	0.00000484988	1 dilevel 2 = layak	Layak
257	-0.0000216112	0.00000428435	1 dilevel 2 = layak	Layak
201	-0.0000181477	-0.00000232013	-1 dilevel 2 = tidak layak	Tidak layak

Nilai Bias di Level 1 = -0,0000822310539661

Nilai Bias di Level 2 = -0,000013389151718

Berdasarkan skenario pengujian ke-4 pada rasio 90%:10%, jumlah data *training* 101 dan jumlah data *testing* 10, nilai *threshold* AHP = 0.01, nilai parameter lamda = 0.5, nilai konstanta *gamma* = 0. 003, nilai *epsilon*=0.001, banyak iterasi maksimum=100, nilai parameter C (*Complexity*)=1 dengan akurasi sebesar 85.71%.

Nomor Peserta	$f^1(x)$	$f^2(x)$	Hasil Klasifikasi	<i>Actual Class</i>
169	0.0000472446	-	1 dilevel 1 = sangat layak	Sangat layak
274	0.00032841	-	1 dilevel 1 = sangat layak	Sangat layak
157	0.00015518	-	1 dilevel 1 = sangat layak	Sangat layak
218	0.00000184964	-	1 dilevel 1 = sangat layak	Sangat layak
133	0.000347792	-	1 dilevel 1 = sangat layak	Sangat layak
221	0.00000369248	-	1 dilevel 1 = sangat layak	Sangat layak
194	-0.00000723534	0.00000653038	1 dilevel 2 = layak	Layak
268	-0.0000094105	-0.00000236291	-1 dilevel 2 = tidak layak	Layak
249	-0.00000970138	-0.00000111485	-1 dilevel 2 = tidak layak	Layak
401	-0.0000161432	-0.00000404283	-1 dilevel 2 = tidak layak	Tidak layak

Nilai Bias di Level 1 = -0,00008168356725946

Nilai Bias di Level 2 = -0,00001411468292645

Nilai α (*alpha*)

Nilai Alpha pada Skenario Pengujian ke-1					
Nomor Peserta	Alpha Level 1	Alpha Level 2	Nomor Peserta	Alpha Level 1	Alpha Level 2
174	0.98032	-	66	0.9803827	-
137	0.9803242	-	215	0.98040456	-
156	0.9803571	-	173	0.98046637	-
214	0.98024964	-	86	1.0	0.5275537
169	0.9804415	-	256	1.0	0.5275542
273	0.9804353	-	166	1.0	0.5275539
170	0.980456	-	226	1.0	0.5275539
163	0.980149	-	194	1.0	0.5275556
168	0.9801519	-	224	1.0	0.5275601
254	0.98015183	-	264	1.0	0.52756
227	0.98033893	-	165	1.0	0.5275579
140	0.9803372	-	250	1.0	0.52755535
152	0.9804388	-	241	1.0	0.5275558
206	0.98044896	-	192	1.0	0.52756214
138	0.9802575	-	220	1.0	0.52756196

274	0.98027873	-	189	1.0	0.52756107
203	0.9804435	-	160	1.0	0.5275572
162	0.9804554	-	67	1.0	0.5275572
228	0.9804598	-	267	1.0	0.52756244
246	0.9804497	-	229	1.0	0.5275561
253	0.98045903	-	263	1.0	0.5275564
167	0.9802619	-	242	1.0	0.5275556
197	0.9804024	-	85	1.0	0.5275568
153	0.9803636	-	65	1.0	0.52755713
157	0.98037434	-	272	1.0	0.52755743
218	0.98046815	-	268	1.0	0.5275617
245	0.9802935	-	188	1.0	0.5275623
216	0.98046404	-	249	1.0	0.5275612
248	0.9804609	-	210	1.0	0.5275579
139	0.9804475	-	177	1.0	0.5275558
133	0.9802716	-	68	1.0	0.5275615
225	0.980467	-	132	1.0	0.5275624
269	0.98046464	-	142	1.0	0.5275612

180	0.98036736	-	179	1.0	0.5275564
178	0.9803671	-	255	1.0	0.5275585
182	0.9803021	-	257	1.0	0.5275568
200	0.9802943	-	204	1.0	0.52755314
159	0.9804609	-	233	1.0	1.0
231	0.9804655	-	265	1.0	1.0
243	0.9804646	-	183	1.0	1.0
258	0.9804649	-	185	1.0	1.0
247	0.980468	-	270	1.0	1.0
230	0.9804746	-	401	1.0	1.0
251	0.9804691	-	69	1.0	1.0
236	0.98046726	-	70	1.0	1.0
87	0.98045534	-	90	1.0	1.0
266	0.98047715	-	201	1.0	1.0
234	0.98047286	-	400	1.0	1.0
89	0.98046404	-	205	1.0	1.0
144	0.98047125	-	198	1.0	1.0
			91	1.0	1.0

Nilai Alpha pada Skenario Pengujian ke-2

Nomor Peserta	Alpha Level 1	Alpha Level 2	Nomor Peserta	Alpha Level 1	Alpha Level 2
211	0.9081555	-	144	0.90859526	-
174	0.90844315	-	196	0.90851575	-
187	0.9081715	-	66	0.9085057	-
135	0.90817505	-	215	0.90852505	-
156	0.90847206	-	173	0.90859723	-
214	0.90834033	-	86	1.0	0.560444
169	0.90857273	-	256	1.0	0.56044406
170	0.9085814	-	166	1.0	0.5604436
163	0.90821725	-	226	1.0	0.5604436
168	0.90821564	-	194	1.0	0.56044525
254	0.9082202	-	224	1.0	0.56045115
227	0.90846354	-	264	1.0	0.5604507
140	0.90846103	-	165	1.0	0.5604469
271	0.90858036	-	250	1.0	0.5604446
152	0.9085726	-	241	1.0	0.56044513

206	0.908578	-	192	1.0	0.56045234
138	0.9083498	-	220	1.0	0.5604519
274	0.90836805	-	189	1.0	0.5604512
203	0.9085748	-	160	1.0	0.5604474
162	0.9085827	-	67	1.0	0.5604474
228	0.9085881	-	267	1.0	0.56045336
246	0.9085792	-	229	1.0	0.56044555
253	0.9085816	-	263	1.0	0.56044585
167	0.9083531	-	242	1.0	0.5604448
197	0.9084992	-	85	1.0	0.56044596
153	0.9084888	-	65	1.0	0.5604466
157	0.9084991	-	272	1.0	0.56044674
218	0.9085914	-	188	1.0	0.5604524
245	0.90838623	-	199	1.0	0.5604507
216	0.90859544	-	249	1.0	0.5604507
248	0.90859234	-	210	1.0	0.5604473
139	0.9085788	-	177	1.0	0.56044555
133	0.90836185	-	68	1.0	0.5604522

225	0.90859693	-	179	1.0	0.56044614
232	0.9085842	-	255	1.0	0.5604484
269	0.90858674	-	257	1.0	0.56044614
180	0.90849334	-	204	1.0	0.5604424
178	0.9084847	-	233	1.0	1.0
182	0.9083979	-	265	1.0	1.0
200	0.90838665	-	183	1.0	1.0
231	0.9085972	-	185	1.0	1.0
243	0.9085951	-	270	1.0	1.0
258	0.9085954	-	401	1.0	1.0
230	0.9085974	-	195	1.0	1.0
221	0.9085995	-	69	1.0	1.0
251	0.9085984	-	70	1.0	1.0
87	0.9085843	-	90	1.0	1.0
266	0.90860486	-	201	1.0	1.0
234	0.90859705	-	205	1.0	1.0
89	0.9085889	-	198	1.0	1.0
			91	1.0	1.0

Nilai Alpha pada Skenario Pengujian ke-3

Nomor Peserta	Alpha Level 1	Alpha Level 2	Nomor Peserta	Alpha Level 1	Alpha Level 2
211	0.9081704	-	234	0.90859276	-
174	0.9084538	-	89	0.90858537	-
187	0.90818506	-	144	0.90859115	-
135	0.9081881	-	196	0.9085145	-
137	0.90845805	-	173	0.90858823	-
214	0.9083363	-	86	1.0	0.5604444
169	0.9085644	-	166	1.0	0.5604443
273	0.90855885	-	226	1.0	0.5604443
170	0.90857977	-	194	1.0	0.56044513
163	0.9082262	-	224	1.0	0.56044954
168	0.9082244	-	264	1.0	0.56045043
254	0.90822834	-	165	1.0	0.56044674
227	0.9084696	-	250	1.0	0.56044537
140	0.908467	-	241	1.0	0.5604457
271	0.9085739	-	192	1.0	0.56045103

152	0.90856254	-	220	1.0	0.560451
206	0.9085712	-	189	1.0	0.5604507
138	0.9083451	-	160	1.0	0.5604471
274	0.9083676	-	67	1.0	0.5604471
203	0.9085666	-	267	1.0	0.560452
162	0.90857965	-	229	1.0	0.56044644
228	0.9085808	-	263	1.0	0.5604465
246	0.9085721	-	242	1.0	0.56044567
253	0.90857863	-	65	1.0	0.56044734
197	0.9084895	-	272	1.0	0.5604465
153	0.90849274	-	268	1.0	0.56045145
157	0.90850794	-	188	1.0	0.56045145
218	0.90858996	-	249	1.0	0.5604505
217	0.90837294	-	210	1.0	0.56044686
245	0.9083849	-	177	1.0	0.5604453
248	0.90858287	-	68	1.0	0.56045085
139	0.9085703	-	132	1.0	0.5604522
133	0.90835875	-	142	1.0	0.5604503

225	0.908588	-	179	1.0	0.5604468
232	0.9085778	-	255	1.0	0.56044805
269	0.9085862	-	257	1.0	0.5604468
180	0.9084966	-	204	1.0	0.5604412
178	0.9084951	-	233	1.0	1.0
182	0.90839964	-	265	1.0	1.0
200	0.90838605	-	183	1.0	1.0
159	0.90858287	-	185	1.0	1.0
231	0.908587	-	270	1.0	1.0
243	0.90858585	-	401	1.0	1.0
258	0.9085874	-	195	1.0	1.0
247	0.9085887	-	69	1.0	1.0
230	0.90859556	-	70	1.0	1.0
221	0.9085913	-	90	1.0	1.0
251	0.9085902	-	201	1.0	1.0
87	0.9085769	-	400	1.0	1.0
266	0.90859777	-	205	1.0	1.0
			91	1.0	1.0

Nilai Alpha pada Skenario Pengujian ke-4

Nomor Peserta	Alpha Level 1	Alpha Level 2	Nomor Peserta	Alpha Level 1	Alpha Level 2
211	0.90814334	-	234	0.9086106	-
174	0.90844053	-	144	0.9086085	-
187	0.9081587	-	66	0.9085041	-
135	0.9081626	-	215	0.9085295	-
137	0.9084509	-	173	0.90860546	-
156	0.9084771	-	86	1.0	0.5604446
214	0.90833634	-	256	1.0	0.56044465
169	0.90857995	-	166	1.0	0.5604443
273	0.90857416	-	226	1.0	0.5604443
170	0.90859437	-	224	1.0	0.5604491
163	0.90820277	-	264	1.0	0.56044966
168	0.9082028	-	165	1.0	0.5604459
254	0.9082051	-	250	1.0	0.56044537
227	0.9084617	-	241	1.0	0.56044614
140	0.90845966	-	192	1.0	0.56045055

271	0.9085885	-	220	1.0	0.56045043
152	0.9085772	-	189	1.0	0.5604504
138	0.90834475	-	160	1.0	0.56044775
274	0.90836513	-	67	1.0	0.56044775
203	0.90858227	-	267	1.0	0.56045204
228	0.9085972	-	229	1.0	0.56044656
246	0.90858674	-	263	1.0	0.5604466
253	0.90859365	-	242	1.0	0.5604455
167	0.9083487	-	85	1.0	0.56044686
197	0.9085439	-	65	1.0	0.56044704
153	0.9084881	-	268	1.0	0.56045103
157	0.90849644	-	188	1.0	0.56045115
218	0.9086038	-	199	1.0	0.5604494
217	0.90837544	-	249	1.0	0.5604494
245	0.9083818	-	210	1.0	0.56044656
216	0.90860236	-	177	1.0	0.5604459
248	0.9086003	-	68	1.0	0.5604507
139	0.9085857	-	132	1.0	0.5604524

133	0.90835804	-	142	1.0	0.5604498
225	0.90860474	-	179	1.0	0.56044686
232	0.90858996	-	255	1.0	0.5604488
269	0.9086013	-	204	1.0	0.56044126
180	0.9084922	-	233	1.0	1.0
178	0.9084902	-	265	1.0	1.0
182	0.90839374	-	183	1.0	1.0
200	0.9083833	-	185	1.0	1.0
231	0.9086043	-	270	1.0	1.0
258	0.90860134	-	401	1.0	1.0
247	0.90860635	-	195	1.0	1.0
230	0.9086099	-	69	1.0	1.0
221	0.90860707	-	70	1.0	1.0
251	0.9086067	-	90	1.0	1.0
236	0.90860534	-	400	1.0	1.0
87	0.9085915	-	205	1.0	1.0
266	0.9086129	-	198	1.0	1.0
			91	1.0	1.0

Nilai Alpha pada Skenario Pengujian ke-5

Nomor Peserta	Alpha Level 1	Alpha Level 2	Nomor Peserta	Alpha Level 1	Alpha Level 2
211	0.9086403	-	144	0.9091374	-
174	0.9089537	-	196	0.9090459	-
187	0.9086544	-	66	0.9090308	-
135	0.90865844	-	215	0.909051	-
137	0.90896165	-	173	0.909135	-
156	0.9089942	-	86	1.0	0.5622894
214	0.90885484	-	256	1.0	0.56229013
273	0.90909946	-	166	1.0	0.5622898
170	0.9091233	-	226	1.0	0.5622898
163	0.9087015	-	224	1.0	0.5622989
168	0.90870017	-	264	1.0	0.5622984
254	0.90870404	-	165	1.0	0.56229436
227	0.90897596	-	250	1.0	0.5622909
140	0.90897226	-	241	1.0	0.5622914
271	0.90911716	-	192	1.0	0.56230086

152	0.90910417	-	220	1.0	0.5623007
206	0.9091132	-	189	1.0	0.5622984
138	0.9088655	-	160	1.0	0.5622945
203	0.9091085	-	67	1.0	0.5622945
162	0.909122	-	267	1.0	0.5623017
228	0.9091264	-	229	1.0	0.56229186
246	0.90911436	-	263	1.0	0.56229246
253	0.9091228	-	242	1.0	0.56229126
167	0.9088733	-	85	1.0	0.5622931
197	0.9090165	-	65	1.0	0.5622937
153	0.90900296	-	272	1.0	0.5622943
217	0.90889585	-	188	1.0	0.5623011
245	0.90891165	-	199	1.0	0.56229955
216	0.9091316	-	210	1.0	0.56229377
248	0.9091278	-	177	1.0	0.56229275
139	0.9091139	-	68	1.0	0.56230074
225	0.90913373	-	132	1.0	0.5623019
232	0.9091209	-	142	1.0	0.56229997

269	0.9091303	-	179	1.0	0.5622927
180	0.9090096	-	255	1.0	0.5622956
178	0.9090043	-	257	1.0	0.56229323
182	0.9089236	-	204	1.0	0.5622872
200	0.90891314	-	233	1.0	1.0
159	0.9091278	-	265	1.0	1.0
231	0.9091336	-	183	1.0	1.0
243	0.9091318	-	185	1.0	1.0
258	0.9091348	-	270	1.0	1.0
247	0.9091336	-	195	1.0	1.0
230	0.9091447	-	69	1.0	1.0
251	0.90913564	-	70	1.0	1.0
236	0.909134	-	90	1.0	1.0
87	0.9091209	-	201	1.0	1.0
266	0.9091459	-	400	1.0	1.0
234	0.90913963	-	205	1.0	1.0
89	0.9091286	-	198	1.0	1.0
		-	91	1.0	1.0