SELEKSI DAN REKOMENDASI PENERIMA BEASISWA DI LEMBAGA GNOTA KEDIRI DENGAN IMPLEMENTASI METODE SVM DAN TOPSIS

SKRIPSI

Untuk memenuhi sebagian persyaratan

memperoleh gelar Sarjana Komputer

Disusun oleh:
M GILVY LANGGAWAN PUTRA
145150209111011

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

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

PENGESAHAN

SELEKSI DAN REKOMENDASI PENERIMA BEASISWA DI LEMBAGA GNOTA KEDIRI
DENGAN IMPLEMENTASI METODE SVM DAN TOPSIS

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :
M. Gilvy Langgawan Putra
NIM: 145150209111011

Skripsi ini telah diuji dan dinyatakan lulus pada
12 Mei 2016
Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I	Dosen Pembimbing II
Imam Cholissodin, S.Si, M.Kom	Ir.Heru Nurwarsito, M.Kom
NIK: 201201 850719 1 001	NIP: 19650402 199002 1 001

Mengetahui,
Ketua Program Studi Informatika / Ilmu Komputer

Issa Arwani, S.Kom, M.Sc
NIP: 19830922 201212 1 003

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

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, 12 Mei 2016

M. Gilvy Langgawan Putra

NIM: 145150209111011

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Tuhan Yang Maha Esa, karena berkat rahmat dan hidayah-Nya, penulis dapat menyelesaikan Skripsi ini dengan tepat waktu. Laporan Skripsi ini berjudul "*SELEKSI DAN REKOMENDASI PENERIMA BEASISWA DI LEMBAGA GNOTA KEDIRI DENGAN IMPLEMENTASI METODE SVM DAN TOPSIS*".

Dalam penulisan skripsi ini penulis mendapatkan banyak bantuan dari berbagai pihak baik secara moril maupun materiil. Dalam kesempatan ini penulis ingin mengucapkan terima kasih kepada :

1. Imam Cholissodin, S.Si, M.Kom selaku Dosen Pembimbing I dan Ir. Heru Nurwasito M.Kom selaku Dosen Pembimbing II selama pengerjaan skripsi.
2. 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. Issa Arwani, S.Kom., M.Sc selaku Ketua Program Studi Informatika/ilmu komputer Universitas Brawijaya Malang.
5. Candra Dewi, S.Kom, M.Sc selaku Penasehat Akademik yang memberikan nasehat baik kepada saya.
6. Kepada Pemerintah Daerah Kabupaten Kediri yang memberikan ijin dalam penelitian ini, serta seluruh instansi yang terkait.
7. Lembaga GNOTA Kabupaten kediri, yang telah bersedia dan memberikan fasilitas dalam penelitian ini.
8. Bapak, Ibu dan seluruh keluarga atas segenap dukungan dan kasih sayang yang telah diberikan.
9. Sahabat-sahabat saya, Herlambang, Khalid, Aldhie, Nanda, Denny, Thessar, Rany, Nunu yang selalu memberikan canda tawanya dan berbagi cerita.
10. Teman-teman SAP 2014 Ilmu Komputer/Informatika tercinta yang selalu memberikan semangat, dorongan, dan bantuan pikiran, khusus buat whenty dan badi.
11. Seluruh civitas academica informatika fakultas ilmu komputer universitas brawijaya, yang telah membantu dalam proses penyelesaian skripsi

Penulis menyadari bahwa dalam penyusunan Laporan skripsi ini masih banyak kekurangan baik format laporan maupun isinya. Untuk itu penulis sangat mengharapkan kritik dan saran yang membangun. Semoga laporan skripsi ini dapat bermanfaat baik bagi pembaca maupun penulis, amin.

Malang, 12 Mei 2016

M. Gilvy Langgawan Putra
gilvylanggawan11@gmail.com

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

ABSTRAK

Gerakan Nasional Orang Tua Asuh (GNOTA). Gerakan tersebut merupakan gerakan organisasi sosial nirlaba, independen dan transparan yang berdiri sejak 29 Mei 1996. Dari hasil wawancara, menemukan beberapa permasalahan pada saat proses seleksi penerima beasiswa yaitu masih menggunakan perhitungan manual seperti pada proses pengklasifikasian, banyaknya data maka sulit untuk menentukan salah satu peserta termasuk dalam klasifikasi apa dan dalam masalah perangkingan masih menggunakan manual, apabila dilakukan secara manual maka panitia mengurutkan satu persatu tidak secara otomatis. sehingga membutuhkan waktu yang terbilang lama yaitu tiga bulan dari proses pendaftaran hingga hasil seleksi, data yang dimasukan juga bisa mengalami kesalahan, *human error*, apabila penyimpanan data akan mengalami penumpukan dan sumber daya yang terbatas. Maka digunakanlah metode SVM untuk klasifikasi data, data yang digunakan sebanyak 111 data, 3 kelas, 13 kriteria, dan menggunakan metode TOPSIS untuk rekomendasi penerima beasiswa, dari hasil semua pengujian dengan rasio perbandingan 90%:10% memiliki akurasi terbaik 100%, pemilihan nilai λ (*Lambda*) = 0.1, konstanta γ (*gamma*) = 0.05, ε = 0.0001, *Iterasi* Maksimum = 1000, rasio perbandingan 90%:10% dan nilai *d*=2, *C* (*Complexity)* = 1. Sehingga akurasi terbaik yaitu sebesar 100% dan rata-rata akurasi sebesar 95.24%, serta kecepatan rata-rata proses yaitu 19.16 detik, waktu tercepat yaitu 13,39 detik dan waktu terlama yaitu 24,07 detik.

Kata kunci: GNOTA, Kediri, Beasiswa, SVM, TOPSIS

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

ABSTRACT

Gerakan Nasional Orang Tua Asuh (GNOTA). That's movement is a movement of non-profit social organization, independent and transparent established since May 29, 1996. From the interviews, found several problems during the process of selection of scholarship recipients is still using manual calculation as in the classification process, the amount of data it is difficult to determine one of the participants included in any classification and the problems for rank still use manual, if done manually, the committee does not sort one by one automatically. so it takes a fairly long time of three months from the registration process until the result of selection, entered data can also experience an error, human error, when the data storage will be congested, the limited resources. Method is used SVM for data classification, data used as many as 111 of data, 3 classes, 13 criteria, and using TOPSIS method for recommendation grantee, of the results of all tests with a ratio of 90%: 10% have the best accuracy 100 %, the selection of values λ (Lambda) = 0.1, constant γ (gamma) = 0.05 ε = 0.0001, Iteration Maximum = 1000, a ratio of 90%: 10% and the value of d = 2, C (Complexity) = 1. So the best accuracy of the total of 100 % and an average accuracy of 95.24%, and the average speed of the process is 19.16 seconds, the fastest time is 13.39 seconds and the longest time is 24.07 seconds.

Keywords: GNOTA, Kediri, Scholarships, SVM, TOPSIS

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

DAFTAR ISI

PENGESAHAN ii
PERNYATAAN ORISINALITAS iii
KATA PENGANTAR iv
ABSTRAK v
ABSTRACT vi
DAFTAR ISI vii
DAFTAR TABEL xii
DAFTAR GAMBAR xv
DAFTAR KODE PROGRAM xvii
DAFTAR LAMPIRAN xviii
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 4
BAB 2 LANDASAN KEPUSTAKAAN 6
2.1 Tinjauan Pustaka 6
2.2 Beasiswa 9
2.3 GNOTA 9
2.3.1 *Visi* 10
2.3.2 *Misi* 10
2.4 Sistem Pendukung Keputusan 11
2.4.2 Tahapan Proses Pengambilan Keputusan 12
2.4.3 Karakteristik Sistem Pendukung Keputusan 12
2.4.4 Manfaat Sistem Pendukung Keputusan 13
2.5 Klasifikasi 13
2.6 Support Vector Machine (SVM) 14
2.6.1 *Support Vector Machine Linear* 16
2.6.2 *Support Vector Machine Non-Linear* 17

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

2.7 Multi-Class SVM 18
2.7.1 *Metode one-against-all* 19
2.7.2 *Metode one-againts-one* 19
2.8 Metode *Sequential Training* pada SVM 20
2.9 Metode *Technique for Order Preference by Similarity to Ideal Solution* (TOPSIS) 22
2.10 Evaluasi 25
BAB 3 METODOLOGI 27
3.1 Studi Literatur 27
3.2 Analisis Kebutuhan 28
3.3 Pengumpulan Data 29
3.4 Perancangan Sistem 29
3.4.1 Arsitektur Pada Sistem Pendukung Keputusan 30
3.4.2 Diagram Blok Sistem 30
3.5 Implementasi Sistem 32
3.6 Pengujian dan Analisis 32
3.7 Pengambilan kesimpulan dan saran 32
BAB 4 PERANCANGAN SISTEM 33
4.1 Pengumpulan Data 34
4.1.1 Observasi 34
4.1.2 Wawancara 34
4.2 Perancangan Sistem Pendukung Keputusan 36
4.2.1 Subsistem Basis Pengetahuan 37
4.2.2 Subsistem Managemen Data 38
4.2.3 Subsistem Managemen Model 39
4.2.3.1 Perhitungan Metode SVM 39
4.2.3.2 Perhitungan dengan Metode TOPSIS 75
4.2.3.3 Perhitungan Evaluasi 85
4.2.4 Subsistem Antarmuka 86
4.2.5 Subsistem Skenario Pengujian 87
4.2.5.1 Pengujian untuk mengetahui perbandingan jumlah data training dan data testing terhadap hasil akurasi. 88

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

4.2.5.2 Pengujian untuk mengetahui waktu eksekusi pada sistem selama proses perhitungan 88
4.2.5.3 Pengujian terhadap tingkat akurasi dari pengaruh nilai parameter lambda 89
4.2.5.4 Pengujian untuk mengetahui pengaruh nilai konstanta gamma terhadap nilai akurasi. 90
4.2.5.5 Pengujian tingkat akurasi terhadap nilai epsilon..... 90
4.2.5.6 Pengujian tingkat akurasi terhadap nilai iterasi maksimum 91
4.2.5.7 Pengujian untuk mengetahui pengaruh nilai C terhadap akurasi..... 91
BAB 5 IMPLEMENTASI SISTEM 93
5.1 Spesifikasi Sistem 94
5.1.1 Spesifikasi Perangkat Keras 94
5.1.2 Spesifikasi Perangkat Lunak 94
5.2 Batasan Implementasi 95
5.3 Implementasi Algoritma 95
5.3.1 Implementasi Algoritma Perhitungan Kernel *Polynomial Degree d* 95
5.3.2 Implementasi Algoritma Perhitungan Matrik Hessian..... 96
5.3.3 Implementasi Algoritma Perhitungan Iterasi..... 97
5.3.4 Implementasi Algoritma Perhitungan Nilai $W.X^{+}$ dan $W.X^{-}$ 99
5.3.5 Implementasi Algoritma Perhitungan Nilai Bias (*b*)..... 100
5.3.6 Implementasi Algoritma Perhitungan Nilai *f(x)* 100
5.3.7 Implementasi Algoritma Perhitungan Nilai Normalisasi Matrik Keputusan 102
5.3.8 Implementasi Algoritma Perhitungan Pembobotan Matrik Normalisasi..... 102
5.3.9 Implementasi Algoritma Perhitungan Solusi Ideal Positif dan Negatif..... 102
5.3.10 Implementasi Algoritma Perhitungan *Separation Measure* ... 103
5.3.11 Implementasi Algoritma Perhitungan Preferensi 104
5.4 Implementasi Antarmuka 104
5.4.1 Antarmuka Halaman Home..... 104
5.4.2 Antarmuka Halaman Load Data 105
5.4.3 Antarmuka Halaman Normalisasi 106

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

5.4.4 Antarmuka Halaman Kernel Polynomial dan Matriks Hessian . 106
5.4.5 Antarmuka Halaman Iterasi 107
5.4.6 Antarmuka Halaman Nilai *f(x)* 108
5.4.7 Antarmuka Halaman TOPSIS 109
BAB 6 PENGUJIAN DAN ANALISIS 112
6.1 Pengujian Untuk Mengetahui Perbandingan Data *Training* dan Data *Testing* Terhadap Hasil Akurasi 112
6.1.1 Skenario pengujian Rasio Perbandingan 112
6.1.2 Analisa Pengujian Rasio Perbandingan 113
6.2 Pengujian Untuk Mengetahui Waktu Eksekusi Pada Sistem Selama Proses Perhitungan. 114
6.2.1 Skenario Pengujian Waktu Eksekusi 114
6.2.2 Analisa Pengujian Waktu Eksekusi 115
6.3 Pengujian Terhadap Tingkat Akurasi dari Pengaruh Nilai Prameter *Lambda*. 115
6.3.1 Skenario Pengujian *Lambda* 115
6.3.2 Analisa Pengujian *Lambda* 116
6.4 Pengujian Untuk Mengetahui Pengaruh Nilai Konstanta *Gamma* Terhadap Nilai Akurasi 117
6.4.1 Skenario Pengujian Konstanta *Gamma* 117
6.4.2 Analisa Pengujian Konstanta *Gamma* 117
6.5 Pengujian Tingkat Akurasi Terhadap Nilai *Epsilon* 118
6.5.1 Skenario Pengujian Nilai *Epsilon* 118
6.5.2 Analisa Pengujian Nilai *Epsilon* 119
6.6 Pengujian Tingkat Akurasi Terhadap Nilai Iterasi Maksimum 120
6.6.1 Skenario Pengujian Nilai Iterasi Maksimum 120
6.6.2 Analisa Pengujian Nilai Iterasi Maksimum 121
6.7 Pengujian Untuk Mengetahui Pegaruh Nilai *C* Terhadap Akurasi 121
6.7.1 Skenario Pengujian Nilai *C* (*Complexity*) 121
6.7.2 Analisa Pengujian Nilai *C* (*Complexity*) 122
6.8 Analisis Tingkat Akurasi 123
BAB 7 PENUTUP 124
7.1 Kesimpulan 124

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

7.2 Saran 124
DAFTAR PUSTAKA..... 125
LAMPIRAN 127

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

DAFTAR TABEL

Tabel 2.1 : Kajian Pustaka 7
Tabel 2.2 *One-against-all* dengan 3 kelas 19
Tabel 2.3 metode *one-against-one* dengan contoh 6 SVM biner 20
Tabel 2.4 *Confusion Matrix* 2x2 25
Tabel 2.5 *Confusion Matrix* 3x3 26
Tabel 3.1 Analisa Kebutuhan Data 28
Tabel 4.1 Contoh Data Dari GNOTA Kediri 36
Tabel 4.2 Data Perserta Beasiswa GNOTA Kediri. 38
Tabel 4.3 Penilaian Alternatif atau Normalisasi Untuk Data Training Level 1 41
Tabel 4.4 Penilaian Alternatif atau Normalisasi Untuk data Testing Level 1 41
Tabel 4.5 Matriks Kernel Polynomial Degree d Level 1 43
Tabel 4.6 Matriks Hessian Level 1 44
Tabel 4.7 Hasil Perhitungan Nilai *Ei* Level 1 47
Tabel 4.8 Hasil Menghitung Nilai Delta Alpha Level 1 48
Tabel 4.9 Hasil Menghitung Nilai Alpha Level 1 49
Tabel 4.10 Nilai α Pada Masing-masing Data Training Dari Iterasi ke-2 Level 1... 49
Tabel 4.11 Nilai Bobot Kelas Positif Data *Training* Level 1 51
Tabel 4.12 Nilai Bobot Kelas Negatif Data *Training* Level 1 51
Tabel 4.13 Penilaian Alternatif atau Normalisasi Untuk Data Testing Level 1 53
Tabel 4.14 Nilai Kernel Terhadap Masing-masing Data *Training* Level 1 55
Tabel 4.15 Nilai Bobot Data Terhadap Masing-masing Data *Training* Level 1..... 56
Tabel 4.16 Nilai Kernel Terhadap Masing-masing Data *Training* Level 1 57
Tabel 4.17 Nilai Bobot Data Terhadap Masing-masing Data *Training* Level 1..... 57
Tabel 4.18 Nilai Kernel Terhadap Masing-masing Data *Training* Level 1 58
Tabel 4.19 Nilai Bobot Data Terhadap Masing-masing Data *Training* Level 1..... 58
Tabel 4.20 Nilai Kernel Terhadap Masing-masing Data *Training* Level 1 59
Tabel 4.21 Nilai Bobot Data Terhadap Masing-masing Data *Training* Level 1..... 60
Tabel 4.22 Nilai Kernel Terhadap Masing-masing Data *Training* Level 1 61
Tabel 4.23 Nilai Bobot Data Terhadap Masing-masing Data *Training* Level 1..... 61
Tabel 4.24 Penilaian Alternatif atau Normalisasi Data Training Level 2 62
Tabel 4.25 Penilaian Alternatif atau Normalisasi Data Testing Level 2 63

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 4.26 Matriks Kernel Polynomial Degree *d* Level 2 64
Tabel 4.27 Matriks Hessian Level 2 64
Tabel 4.28 Hasil Perhitungan *Ei* Level 2 66
Tabel 4.29 Hasil Menghitung Nilai Delta Alpha Level 2 66
Tabel 4.30 Hasil Menghitung Nilai Alpha Level 2 67
Tabel 4.31 Nilai α Pada Masing-Masing Data Training Hasil Dari Iterasi ke-2 Level 2 68
Tabel 4.32 Nilai Kelas Positif Data *Training* Level 2 68
Tabel 4.33 Nilai Bobot Kelas Negatif Data *Training* Level 2 69
Tabel 4.34 Penilaian Alternatif atau Normalisasi Untuk Data Testing Level 2 70
Tabel 4.35 Nilai Kernel Terhadap Masing-masing Data *Training* Level 2 70
Tabel 4.36 Nilai Bobot Data Terhadap Masing-masing Data *Training* Level 2..... 71
Tabel 4.37 Nilai Kernel Terhadap Masing-masing Data *Training* Level 2 72
Tabel 4.38 Nilai Bobot Data Terhadap Masing-masing Data *Training* Level 2..... 72
Tabel 4.39 Nilai Kernel Terhadap Masing-masing Data *Training* Level 2 73
Tabel 4.40 Nilai Bobot Data Terhadap Masing-masing Data *Training* Level 2..... 74
Tabel 4.41 Data Hasil Perhitungan *f(X)* dan Klasifikasi 74
Tabel 4.42 Data Testing Hasil Perhitungan dengan Metode SVM 77
Tabel 4.43 Bobot Kriteria 77
Tabel 4.44 Matriks Normalisasi 78
Tabel 4.45 Matriks pembobotan pada matriks yang telah dinormalisasi 80
Tabel 4.46 Solusi Ideal Positif 81
Tabel 4.47 Solusi Ideal Negatif 81
Tabel 4.48 Separation Measure Positif 83
Tabel 4.49 Separation Measure Negatif 83
Tabel 4.50 Nilai Preferensi 85
Tabel 4.51 Model Evaluasi *Confusion* Matriks 86
Tabel 4.52 Skenario Pengujian Rasio Perbandingan 88
Tabel 4.53 Skenario Pengujian Waktu Eksekusi 89
Tabel 4.54 Skenario Pengujian Parameter Lambda 89
Tabel 4.55 Skenario Pengujian Nilai Konstanta Gamma 90
Tabel 4.56 Skenario Pengujian nilai *epsilon* 90
Tabel 4.57 Pengujian Tingkat iterasi maksimum 91

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 4.58 Pengujian untuk mengetahui pengaruh nilai *C* 91
Tabel 5.1 Spesifikasi Perangkat Keras 94
Tabel 5.2 Spesifikasi Perangkat Lunak 94
Tabel 6.1 Hasil Pengujian untuk mengetahui perbanndingan data *training* dan data *testing* 113
Tabel 6.2 Pengujian Untuk Mengetahui Waktu Eksekusi Pada Sistem Selama Proses Perhitungan 114
Tabel 6.3 Hasil Pengujian *Lambda* 116
Tabel 6.4 Hasil Pengujian Kosntanta $\boldsymbol{\gamma}$ (*gamma)* 117
Tabel 6.5 Hasil Pengujian Nilai $\boldsymbol{\varepsilon}$ (*epsilon)* 119
Tabel 6.6 Skenario Pengujian Nilai Iterasi Maksimum 120
Tabel 6.7 Skenario Pengjian Nilai *C* (*Complexity*) 122

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

DAFTAR GAMBAR

Gambar 2.1 Logo GNOTA 10
Gambar 2.2 Komponen Sistem Pendukung Keputusan 12
Gambar 2.3 Penerapan Metode SVM 15
Gambar 2.4 Pengaruh nilai *C* 15
Gambar 2.5 Klasifikasi *one-against-all* untuk 3 kelas 19
Gambar 2.6 Klasifikasi menggunakan metode *one-againts-one* 20
Gambar 3.1 Diagram Alir Tahapan Penelitian 27
Gambar 3.2 Arsitektur Pada Sistem Pendukung Keputusan 30
Gambar 3.3 Diagram Blok Sistem Pendukung Keputusan Rekomendasi Penerima Beasiswa GNOTA 31
Gambar 4.1 Diagram Perancangan 33
Gambar 4.2 Alur Kerja SPK Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri 37
Gambar 4.3 Diagram Alir Perhitungan Menggunakan Metode SVM 40
Gambar 4.4 Diagram Alir Perhitungan Matriks Kernel Polynomial Degree *d* 42
Gambar 4.5 Diagram Alir Perhitungan Matriks Hessian 44
Gambar 4.6 Diagram Alir Proses *Sequential Training SVM* 45
Gambar 4.7 Diagram Alir Menghitung Nilai Ei 46
Gambar 4.8 Diagram Alir Maksimum Nilai Delta Alpha 50
Gambar 4.9 Diagram Alir Menghitung Nilai Bias 52
Gambar 4.10 Diagram Alir Menghitung Nilai dari Fungsi *f(x)* 54
Gambar 4.11 Proses *One-Against-All* 54
Gambar 4.12 Diagram Alir Perhitungan Metode TOPSIS 76
Gambar 4.13 Diagram Alir Membuat Normalisasi Matriks 78
Gambar 4.14 Diagram Alir Perhitungan pembobotan pada matriks yang telah dinormalisasi 79
Gambar 4.15 Diagram Alir Menentukan nilai solusi ideal positif dan solusi ideal negatif 80
Gambar 4.16 Diagram Alir Menghitung Nilai Separation Measure 82
Gambar 4.17 Diagram Alir Menghitung Nilai Preferensi 84
Gambar 4.18 Perancangan Antarmuka Sistem 87
Gambar 5.1 Diagram Implementasi Sistem 93

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Gambar 5.2 Antarmuka Halaman Home 105
Gambar 5.3 Halaman Antarmuka Load Data 105
Gambar 5.4 Halaman Antarmuka Normalisasi 106
Gambar 5.5 Halaman Antarmuka Kernel Polynomial da Matriks Hessian Level 1 106
Gambar 5.6 Halaman Antarmuka Kernel Polynomial dan Matrik Hessian Level 2 107
Gambar 5.7 Halaman Antarmuka Iterasi Level 1 107
Gambar 5.8 Halaman Antarmuka Iterasi Level 2 108
Gambar 5.9 Halaman Antarmuka Nilai *f(x)* Level 1 108
Gambar 5.10 Halaman Antarmuka Nilai *f(X)* Level 2 109
Gambar 5.11 Halaman Antarmuka *Load* Kriteria 109
Gambar 5.12 Antarmuka Halaman Matrik Normalisasi 110
Gambar 5.13 Antarmuka Halaman Pembobotan Matriks Normalisasi 110
Gambar 5.14 Antarmuka Halaman Ideal Positif dan Ideal Negatif 111
Gambar 5.15 Antarmuka Halaman Perangkingan 111
Gambar 6.1 Diagram Proses Pengujian dan Analisis 112
Gambar 6.2 Grafik Tingkat Akurasi Hasil Pengujian Rasio Perbandingan 113
Gambar 6.3 Grafik Waktu Eksekusi Pada Sistem Selama Proses Perhitungan ... 115
Gambar 6.4 Grafik Tingkat Akurasi Hasil Pengujian nilai *Lambda* 116
Gambar 6.5 Grafik Tingkat Akurasi Hasil Pengujian Konstanta *Gamma* 118
Gambar 6.6 Grafik Tingkat Akurasi Hasil Pengujian nilai *Epsilon* 119
Gambar 6.7 Grafik Tingkat Akurasi Hasil Pengujian Iterasi Maksimum 121
Gambar 6.8 Grafik Tingkat Akurasi Hasil Pengujian Nilai *Complexity* 122

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

DAFTAR KODE PROGRAM

Kode Program 5.1 Implementasi Algoritma Perhitungan Kernel *Polynomial Degree d* 96

Kode Program 5.2 Implementasi Algoritma Perhitungan Matrik *Hessian* 97

Kode Program 5.3 Implementasi Algoritma Perhitungan Nilai Ei 98

Kode Program 5.4 Implementasi Algoritma Perhitungan Nilai Delta Alpha 99

Kode Program 5.5 Implementasi Algoritma Perhitungan Nilai Alpha Baru 99

Kode Program 5.6 Implementasi Algoritma Perhitungan ($W.X^+$) dan ($W.X^-$) 100

Kode Program 5.7 Implementasi Algoritma Perhitungan Nilai Bias 100

Kode Program 5.8 Implementasi Algoritma Perhitungan Nilai *f(x)* 101

Kode Program 5.9 Implementasi Algoritma Perhitungan Nilai Normalisasi Matriks Keputusan 102

Kode Program 5.10 Implementasi Algoritma Perhitungan Pembobotan Matriks Normalisasi 102

Kode Program 5.11 Implementasi Algoritma Perhitungan Solusi Ideal Positif dan Negatif 103

Kode Program 5.12 Implementasi Algoritma Perhitungan *Separation Measure* 103

Kode Program 5.13 Implementasi Algoritma Perhitungan Preferensi 104

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

DAFTAR LAMPIRAN

Lampiran A Form pendaftaran 127
Lampiran B Data Pendaftar Beasiswa 131
Lampiran C Parameter Dari Setiap Kriteria 136
Lampiran D Perhitungan Manualisasi Metode SVM Level 1 138
Lampiran E Perhitungan Manualisasi Metode SVM Level 2 147
Lampiran F Perhitungan Manulisasi Metode TOPSIS 154
Lampiran G Hasil Pengujian 158
Lampiran H Surat Izin Penelitian Pemerintah Kabupaten Kediri 188
Lampiran I Bobot Metode TOPSIS 189
Lampiran J Inisialisasi Data 190
Lampiran K Surat Persetujuan Menjadi Pakar 191

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

BAB 1 PENDAHULUAN

Bab ini menguraikan mengenai latar belakang, rumusan masalah, tujuan, manfaat, batasan masalah dan sistematika pembahasan Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri dengan Implementasi Metode SVM dan TOPSIS.

1.1 Latar belakang

Undang-Undang Dasar 1945 pada alinea 4 yang berbunyi "kemudian dari pada itu untuk membentuk suatu pemerintah negara Indonesia yang melindungi segenap bangsa indonesia dan seluruh tumpah darah Indonesia dan untuk memajukan kesejahteraan umum, mencerdaskan kehidupan bangsa.." dari penggalan alinea 4, dapat disimpulkan bahwa semua masyarakat atau organisasi di Indonesia berhak ikut serta mencerdaskan kehidupan bangsanya sendiri.

Merujuk pada UUD tersebut maka berdirilah sebuah gerakan yang berinisiatif dari masyarakat untuk menjaga agar anak-anak Indonesia mendapatkan pendidikan sebagai landasan meraih masa depan mereka yang lebih baik. Gerakan yang mendukung dalam permasalahan mencerdaskan kehidupan bangsa yaitu sebuah gerakan yang bernama Gerakan Nasional Orang Tua Asuh (GNOTA). Gerakan tersebut merupakan gerakan organisasi sosial nirlaba, independen dan transparan yang berdiri sejak 29 Mei 1996. GNOTA sendiri telah mendistribusikan bantuan sebesar 2,3 juta paket pendidikan dan donasi uang yang tersebar di seluruh Indonesia, untuk membantu anak-anak dari keluarga yang kurang mampu agar dapat meneruskan sekolah dan menyelesaikan pendidikannya (GNOTA,2015). Banyaknya penerima beasiswa yaitu terdiri dari 160 orang mahasiswa dan 160 orang tua siswa yang kurang mampu penerima beasiswa 2014 dari GNOTA Kediri (Pemkab kediri, 2014). Dari hasil wawancara, menemukan beberapa permasalahan pada saat proses seleksi penerima beasiswa yaitu masih menggunakan perhitungan manual seperti pada proses pengklasifikasian yang mana ada tiga klasifikasi dengan banyaknya data maka sulit untuk menentukan salah satu peserta termasuk dalam klasifikasi apa dan dalam masalah perangkingan masih menggunakan manual, apabila dilakukan secara manual maka panitia mengurutkan satu persatu tidak secara otomatis. Pengisian data secara manual dengan menggunakan form pendaftaran kemudian dimasukkan ke program *excel* sehingga membutuhkan waktu yang terbilang lama yaitu tiga bulan dari proses pendaftaran hingga hasil seleksi, data yang dimasukan juga bisa mengalami kesalahan seperti tidak samanya data di *excel* dengan data pada saat pengisian *form* pendaftaran dikarenakan faktor *human error*, apabila data dari tahun ketahun bertambah banyak dari pendaftar beasiswa maka proses penyimpanan data akan mengalami penumpukan berupa berkas yang tidak dapat tertangani oleh sumber daya yang terbatas dan tidak ada jejak *record*-nya, kurangnya sumber daya manusia apabila melakukan survei satu-persatu dikarenakan jumlah pegawai lembaga GNOTA Kediri hanya terdiri dari satu orang kepala lembaga dan dua orang karyawan. Dari banyak beberapa pemasalahan

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

tersebut maka lembaga GNOTA Kediri membutuhkan sebuah sistem yang mampu mendukung dalam merekomendasikan siapa yang berhak menerima bantuan dan mana yang tidak, dan pada akhir nantinya dari organisasi GNOTA dapat menentukan penerima beasiswa dengan melihat hasil rekomendasi dari sistem.

Pada penelitian yang pernah dilakukan oleh Sahar A. Mochtar dan Alaa. M. Elyasad mengenai perbandingan tiga metode klasifikasi yaitu *Desicion Tree* (DT), *Artifical Neural Network* (ANN), dan *Support Vector Machine* pada studi kasus memprediksi tingkat keparahan penyakit kanker payudara dengan nilai akurasi dari masing-masing metode yaitu 78,12%, 80,56%, dan 81,25. dari hasil nilai akurasi tersebut metode SVM yang memiliki nilai tertinggi untuk nilai akurasinya untuk metode klasifikasi (Sahar A., *et al,* 2010). Ada juga penelitian untuk perbandingan klasifikasi yaitu antara KNN dan SVM yang mana diterapkan pada studi kasus pengenalan karakter bahasa Oriya (bahasa dinegara bagian India), dengan nilai akurasi untuk KNN 96,47% dan metode SVM nilai akurasinya 98,9%, (Mohanty dan Bebartta, 2011). Nilai akurasi tersebut dapat kita ketahui tingkat akurasi dengan metode klasifikasi SVM sangat tinggi dibanding dengan metode lain, sehingga itu mendukung penelitian ini untuk pengklasifikasian penerima beasiswa menggunakan metode Support Vector Machine.

Selanjutnya penelitian untuk metode TOPSIS yang dilakukan oleh S. Lestari, metode TOPSIS diterapkan pada seleksi penerimaan karyawan dengan menggunakan tujuh kriteria dan sepuluh alternatif, dengan hasil menggunakan TOPSIS komputasinya yang efisien serta memimiliki kemampuan relatif dalam mengukur alternatif-alternatif keputusan dalam bentuk matematis sederhanaan cocok untuk menyelesaikan permasalahan multi dimensi (Lestari,S., 2011). Dan penelitian selanjutnya yang dilakukan oleh Basjir *et all* (2014) yang berjudul "pengembangan model penentuan prioritas perbaikan terhadap komponen dangan metodologi FMEA, Fuzzy dan TOPSIS" maka hasil rekomendasi perbaikan dengan TOPSIS mampu memperbaiki hasil prioritas metode FMEA dan Fuzzy-FMEA, metode TOPSIS yang lebih *reliable*, sehinga dipilih karena juga mampu melakukan perangkingan terhadap alternatif terpilih, logikanya sederhana sehingga menghasilkan komputasi yang baik. Maka dari itulah memilih metode TOPSIS untuk perangkingan.

Sehingga diperoleh penggabungan dua metode yaitu SVM dan TOPSIS yang didasari dari penelitian sebelumnya oleh Yung-Ming Li *et all* (2010) dengan mengangkat topik *Building a qualitative recruitment system via SVM with MCDM approach,* dari penelitian tersebut pada metode TOPSIS nya mengalami permasalahan yaitu kesulitan memberikan pendekatan rekomendasi penempatan pegawai yang sesuai, sehingga menambahkan metode SVM untuk mengatasi proses menentukan rekomendasi posisi pegawai. Dan adapun untuk perbandingan metode SVM dan AHP apada penelitian Yang-yi *et all (*2010*)* yang berjudul "*Logistic System and Intelegent Management, The Application of Site Selection based on AHP-SVM in 500KV substaton*" dari hasil penelitiannya yang mana AHP untuk menentukan bobot dari 20 kriteria, kemudian terpilih menjadi 12

kriteria, sehingga hasil evaluasi AHP-SVM hanya memperoleh nilai akurasi sebesar 80%.

Dari beberapa penelitian sebelumnya, maka dalam penelitian rekomendasi penerima beasiswa GNOTA menggunakan metode SVM-TOPSIS, sehingga nantinya skripsi ini berjudul "Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri dengan Implementasi Metode SVM dan TOPSIS". Diharapkan pada Implementasi Metode SVM dan TOPSIS menghasilkan nilai akurasi terbaik, sehingga dapat meminimalkan kesalahan dalam pengrekomendasian penerima beasiswa.

1.2 Rumusan masalah

Berdasarkan latar belakang yang telah diuraikan, maka dapat dirumuskan permasalahan sebagai berikut :

1. Bagaimana kecepatan proses seleksi dan rekomendasi penerimaan beasiswa dengan sistem dari implementasi metode SVM dan TOPSIS dalam proses seleksi beasiswa di lembaga GNOTA Kediri ?
2. Bagaimana tingkat akurasi dari implementasi metode SVM dan TOPSIS untuk seleksi dan rekomendasi penerima beasiswa di lembaga GNOTA Kediri ?

1.3 Tujuan

Berdasarkan dari perumusan masalah, maka tujuan yang ingin dicapai dalam penelitian ini adalah :

1. Mengimplementasikan metode SVM dan TOPSIS untuk seleksi dan rekomendasi penerima beasiswa di lembaga GNOTA Kediri.
2. Menguji tingkat akurasi dari dari implementasi metode SVM dan TOPSIS untuk seleksi dan rekomendasi penerima beasiswa di lembaga GNOTA Kediri.

1.4 Manfaat

Manfaat yang diharapkan dengan adanya penelitian ini adalah :

1. Membantu lembaga GNOTA kediri untuk mendapat rekomendasi penerima beasiswa secara efektif.
2. Meningkatkan efisiensi waktu dan tenaga dalam proses rekomendasi penerima beasiswa GNOTA Kediri.
3. Memberikan perhitungan yang sistematis untuk mendapatkan hasil rekomendasi yang optimal.

1.5 Batasan masalah

Batasan masalah yang akan dijadikan pedoman dalam pelaksanaan penelitian ini adalah sebagai berikut :

1. Metode yang digunakan dalam Implementasi untuk Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri adalah metode SVM dan TOPSIS.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

2. Penelitian ini sebatas Implementasi Metode SVM dan TOPSIS untuk Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri.
3. Penelitian ini mengambil data dari hasil seleksi penerima beasiswa GNOTA Kediri pada tahun 2015 dengan ruang lingkup GNOTA Kediri.
4. Parameter yang akan diuji adalah tingkat akurasi dari hasil rekomendasi penerima beasiswa GNOTA Kediri dengan menggunakan metode SVM dan TOPSIS.

1.6 Sistematika pembahasan

Sistematika penulisan penelitian ditunjukkan untuk memberikan gambaran dan uraian dari penyusun tugas akhir secara garis besar yang meliputi beberapa bab sebagai berikut :

BAB I PENDAHULUAN

Menguraikan mengenai latar belakang, rumusan masalah, tujuan, manfaat, batasan masalah dan sistematika pembahasan Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri dengan Implementasi Metode SVM dan TOPSIS.

BAB II LANDASAN KEPUSTAKAAN

Menguraikan tentang dasar teori dan referensi atau kajian pustaka yang mendukung proses perancangan pada Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri dengan Implementasi Metode SVM dan TOPSIS.

BAB III METODOLOGI

Membahas mengenai metode dan langkah-langkah kerja yang dilakukkan dalam penelitian yang terdiri dari studi literatur, analisis kebutuhan, pengumpulan data, perancangan sistem,implementasi sistem, pengujian dan analisis, pengambilan kesimpulan dan saran dari Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri dengan Implementasi Metode SVM dan TOPSIS.

BAB IV PERANCANGAN SISTEM

Membahas mengenai pengumpulan data dan perancangan sistem pengklasifikasian dan perangkingan dari Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri dengan Implementasi Metode SVM dan TOPSIS.

BAB V IMPLEMENTASI

Menguraikan proses-proses dari Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri dengan Implementasi Metode SVM dan TOPSIS.

BAB VI PENGUJIAN DAN ANALISIS

Membahas proses dari pengujian Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri dengan Implementasi Metode SVM dan TOPSIS, dengan proses yang diuji adalah mengukur tingkat akurasi sistem.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

BAB VII PENUTUP

Menguraikan kesimpulan dan saran dari seluruh proses penelitian sistem pendukung keputusan Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri dengan Implementasi Metode SVM dan TOPSIS.

BAB 2 LANDASAN KEPUSTAKAAN

Bab ini akan menguraikan tentang tinjauan pustaka dan dasar teori yang mendukung penelitian ini, sumbernya dari beberapa refrensi diantaranya kajian pustaka, dan teori Organisasi GNOTA, Beasiswa, SPK, Metode SVM, Metode TOPSIS, dan Evaluasi.

2.1 Tinjauan Pustaka

SVM (*Support Vector Machine*) adalah seperangkat metode pembelajaran mesin pada teori stastistika Vladimir Vapnik (Kartal and Cebi, 2013). Metode SVM merupakan salah satu metode yang biasanya digunakan untuk pemilihan variabel dan proses klasifikasi data (Noviyanti and Purnami, 2012). Teknik SVM dilakukan dengan cara analisis berdasarkan hitungan matematis sehingga dapat membantu di dalam pemecahan masalah. Teknik yang ada dalam memecahkan masalah yang menggunakan SVM adalah dengan menentukan suatu *hyperplane* yang terbaik. *Hyperplane* merupakan suatu bidang keahlian pemisah yang terletak ditengah antara dua set objek dari dua kelas.

Support Vector Machine dapat melakukan proses generalisasi untuk menentukan pola tertentu berdasarkan data *testing* dan data *training*. SVM juga cepat dalam menyelesaikan suatu masalah karena metode SVM memiliki struktur sederhana. Metode SVM ini mempunyai kelebihan menemukan fungsi pemisah atau pengklasifikasian yang optimal (Damayanti *et all,* 2010). Penerapan metode SVM dapat digunakan hanya untuk klasifikasi dua kelas dan selalu memberikan hasil yang sama dalam setiap kali melakukan proses.

Metode TOPSIS adalah metode klasik MCDM yang alternatif dihitung dengan atribut dan setiap masalah dianggap sebagai sistem geometris yang terdiri alternatif titik dan atribut dimensi ruang (Yang, 2007). TOPSIS adalah multi atribut dalam teknik pengambilan keputusan yang diterima secara luas karena logika yang bagus, pertimbangan simultan yang ideal dan solusi anti-ideal dan prosedur perhitungan yang mudah diprogram (Onüt dan Soner, 2008). Dan menurut kusuma mukhtar atmaja(2010) metode TOPSIS memperhatikan dengan baik jarak kesolusi ideal positif maupun jarak ke solusi ideal negatif dengan mengambil hubungan untuk menuju solusi yang ideal, dengan melakukan perbandingan pada keduanya, urutan pilihan dapat ditentukan.

Pada penelitian pertama yang dilakukan oleh Sahar A. Mochtar dan Alaa. M. Elyasad mengenai perbandingan tiga metode klasifikasi yaitu *Desicion Tree* (DT), *Artifical Neural Network* (ANN), dan *Support Vector Machine* pada studi kasus memprediksi tingkat keparahan penyakit kanker payudara dengan nilai akurasi dari masing-masing metode yaitu 78,12%, 80,56%, dan 81,25. yang mana dari hasil nilai akurasi metode SVM yang memiliki nilai tertinggi untuk nilai akurasinya untuk metode klasifikasi.

Penelitian kedua membahas perbandingan klasifikasi antara KNN dan SVM yang diterap di studi kasus untuk pengenalan karakter bahasa Oriya (bahasa

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

dinegara bagian India), dengan nilai akurasi untuk metode KNN 96,47% dan dengan metode SVM nilai akurasinya 98,9%, (Mohanty dan Bebartta, 2011).

Penelitian ketiga membahas metode TOPSIS yang mana dilakukan oleh S. Lestari, yang metode TOPSIS diterapkan pada seleksi penerimaan karyawan dengan menggunakan tujuh kriteria dan sepuluh alternatif, dengan hasil menggunakan metode TOPSIS komputasinya yang efisien serta memiliki kemampuan relatif dalam mengukur alternatif-alternatif keputusan dalam bentuk matematis sederhanaan cocok untuk menyelesaikan permasalahan multi dimensi (Lestari,S., 2011). Dan penelitian keempat yang dilakukan oleh Basjir *et all* (2014) yang berjudul "pengembangan model penentuan prioritas perbaikan terhadap komponen dangan metodologi FMEA, Fuzzy dan TOPSIS" maka hasilnya rekomendasi perbaikan dengan TOPSIS mampu memperbaiki hasil prioritas metode FMEA dan Fuzzy-FMEA, metode TOPSIS yang lebih *reliable*, sehingga dipilih karena juga mampu melakukan perangkingan terhadap alternatif terpilih, logikanya sederhana sehingga menghasilkan komputasi yang baik.

Penelitian kelima penggabungan dua metode yaitu SVM dan TOPSIS yang didasari dari penelitian sebelumnya oleh Yung-Ming Li *et all* (2010) dengan mengangkat topik *Building a qualitative recruitment systemvia SVM with MCDM approach* yang mana dari penelitian tersebut pada metode TOPSIS nya mengalami permasalahan yaitu kesulitan memberikan pendekatan rekomendasi penempatan pegawai yang sesuai, sehingga menambahkan metode SVM untuk mengatasi proses menentukan rekomendasi posisi pegawai. Dan adapun penelitian keenam untuk perbandian metode SVM dan AHP pada penelitian Yang-yi *et all (*2010*)* yang berjudul "*Logistic System and Intelegent Management, The Application of Site Selection based on AHP-SVM in 500KV substaton*" dari hasil penelitiannya yang mana AHP untuk menentukan bobot dari 20 kriteria, kemudian terpilih menjadi 12 kriteri, sehingga hasil evaluasi AHP-SVM hanya memperoleh nilai akurasi sebesar 80%.

Secara garis besar rangkuman dari penelitian diatas ditampilkan pada Tabel 2.1.

Tabel 2.1 : Kajian Pustaka

No	Judul	Objek	Metode	Hasil
1	Predicting the Severity of Breast Masses with Data Mining Methods	Breast Masses	DT, SVM, dan ANN	*Desicion Tree* (DT), *Artifical Neural Network* (ANN), dan *Support Vector Machine* pada studi kasus memprediksi tingkat keparahan penyakit kanker payudara yang mana nilai akurasi dari masing-masing metode yaitu 78,12%, 80,56%, dan 81,25 yang mana dari hasil nilai akurasi SVM lah yang memiliki nilai tertinggi

				untuk nilai akurasinya untuk metode klasifikasi.
2	Performance Comparison of SVM dan K-NN for Oriya Character Recognition	Tipe image dan size dari sample seperti Bold and small, Bold and big, normal s and small, normal and bold.	SVM-KNN	Pengenalan karakter bahasa Oriya (bahasa di negara bagian India), yang mana nilai akurasi untuk KNN 96,47% dan dengan metode SVM nilai akurasinya 98,9%.
3	Implementasi Metode Fuzzy TOPSIS untuk Seleksi Penerimaan Karyawan	Penerimaan Karyawan Nilai hasil dari beberapa tes untuk masing-masing alternatif calon karyawan	TOPSIS	Hasil menggunakan TOPSIS komputasinya yang efisien serta memimiliki kemampuan relatif dalam emngukur alternatif-alternatif keputusan dalam bentuk matematis sederhanaan cocok untuk menyelesaikan permasalahan multi dimensi.
4	Pengembangan model Penentuan Prioritas Perbaikan Terhadap Mode Kegagalan Komponen dengan Metodologi FMEA, Fuzzy dan TOPSIS yang terintegrasi	Komponen permesinan seperti Panel SAW, Tabel SAW Blade sander, edging, drilling	FMEA, Fuzzy dan TOPSIS	Hasilnya rekomendasi perbaikan dengan TOPSIS mampu memperbaiki hasil prioritas metode FMEA dan Fuzzy-FMEA, metode TOPSIS yang lebih *reliable*, sehinga dipilih karena juga mampu melakukan perangkingan terhadap alternatif terpilih, logikanya sederhana sehingga menghasilkan komputasi yang baik.
5	*Building a qualitative recruitment system via SVM with MCDM approach*	Nilai dari 5 faktor klasisfikasi: 1. *Extraversion* 2. *Agreeableness* 3. *Neuroticism* 4. *Conscientiousness* 5. *Opennes* Nilai parameter TOPSIS 1. skor keahlian 2. skor ketergantungan skor kesesuaian	SVM dan MCDM TOPSIS	TOPSIS nya mengalami permasalahan yaitu kesulitan memberikan pendekatan rekomendasi penempatan pegawai yang sesuai, sehingga menambahkan metode SVM untuk mengatasi proses menentukan rekomendasi posisi pegawai.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

6	Logistic System and Intelegent Management, The Application of Site Selection Based on AHP-SVM in 500KV Substations	Indeks evaluasi proyek transmisi dan stasiun transmisi dalam pasokan	AHP-SVM	Hasil penelitiannya yang mana AHP untuk menentukan bobot dari 20 kriteria, kemudian terpilih menjadi 12 kriteri, sehingga hasil evaluasi AHP-SVM hanya memperoleh nilai akurasi sebesar 80%.
7	Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri dengan Implementasi Metode SVM dan TOPSIS	Pendaftar Beasiswa GNOTA kediri untuk MABA Perguturan Tinggi Negeri	SVM dan TOPSIS	Hasil penelitian nantinya pada metode SVM untuk pengklasifikasian penerima beasiswa di GNOTA Kediri yang terdiri dari Sangat layak, Layak dan Tidak layak, setelah dilakukan pengklasifikasian akan dilakukan pengrangkingan dengan metode TOPSIS, hanya data yang termasuk dalam Klasifikasi Sangat layak yang akan diproses dalam metode TOPSIS, sehingga dapat menghasilkan rekomendasi penerima beasiswa GNOTA Kediri

Sumber : Sahar A. Mochtar dan Alaa. M. Elyasad (2010), Mohanty dan Bebartta (2011), Lestari,S (2011), Basjir *et all*(2014),Yung-Ming Li *et all* (2010),Yang-yi *et all* (2010).

2.2 Beasiswa

Pada dasarnya, beasiswa merupakan penghasilan bagi penerimanya, yang mana tertuang dalam pasal 4 ayat (1) UU PPh/2000/ yang disebutkan pengertian penghasilan merupakan tambahan kemampuan ekonomis dengan nama dan dalam bentuk apapun yang diterima atau diperoleh dari sumber Indonesia atau luar Indonesia yang dapat digunakan untuk konsumsi atau menambah kekayaan Wajib Pajak (WP). Dengan demikian beasiswa dapat diartikan sebagai penambahan kemampuan secara ekonomi bagi penerimanya, yang mana beasiswa merupakan penghasilan (Jawa Pos, 2015).

2.3 GNOTA

GNOTA merupakan sebuah organisasi sosial nirlaba, independen dan trasnparan yang didirikan pada tanggal 1996. GNOTA sendiri sebuah gerakan inisiatif dari masyarakat untuk menjaga agar anak-anak indonesia mendapatkan

pendidikan dasar sebagai landasan meraih masa depan yang lebih baik lagi. Sejak awal beridiri sampai sekarang GNOTA telah mendistribusikan 2,3 juta paket bantuan pendidikan dan donasi uang untuk membantu anak-anak dari keluarga yang kurang mampu agar terus dapat bersekolah dan menuntaskan pendidikan dasarnya.

Negara bertanggung jawab untuk menyediakan kesempatan pendidikan dasar bagi anak-anak bangsa, apapun itu status ekonomi keluarganya. Program yang dicanangkan yaitu wajib belajar 9 tahun menunjukan komitmen pemerintah untuk memastikan agar anak-anak wajib mendapatkan pendidikan sekolah dasar. Walaupun sekoleh negeri telah membebaskan biaya sekolah dan siswa tidak harus membeli buku-buku pelajaran, ribuan anak Indonesia berhenti sekolah karena kondisi ekonomi keluarganya, sehingga itu merupakan permasalahan kompleks yang tidak dapat diselesaikan dalam semalam.

Sehingga dengan adanya lembaga GNOTA ini dapat membantu menyediakan peralatan sekolah dan berbagai kebutuhan lainya untuk bersekolah, dengan demikian anak-anak dapat fokus belajar (GNOTA,2015).

Gambar 2.1 Logo GNOTA

Sumber: (http://www.gn-ota.or.id/)

Gambar 2.1 merupakan logo dari GNOTA (Gerakan Nasional Orang Tua Asuh) yang berdiri sejak 1996.

2.3.1 *Visi*

Yayasan kami memfokuskan diri pada 2 (dua) hal yaitu, Pendidikan dan Kemanusiaan. Dalam kedua bidang ini, kami ingin meningkatkan kesadaran serta tanggung jawab sosial untuk mendukung Program Wajib Belajar Pendidikan Dasar Indonesia.

Dengan kesadaran yang meningkat, kami berharap masyarakat akan lebih peduli terhadap kualitas Pendidikan Dasar sebagai landasan untuk pendidikan lanjutan serta bersama-sama melakukan tindakan nyata untuk menjamin keberlangsungan Pendidikan Dasar bagi anak-anak Indonesia, terutama yang tinggal di daerah terpencil.

2.3.2 *Misi*

Kami secara aktif turut serta mendukung pemerataan dan penyelesaian dari Wajib Belajar Pendidikan Dasar 9 Tahun yang diluncurkan pemerintah Indonesia. Kami terus mensosialisasikan kegiatan GNOTA, sehingga dapat mengembangkan,

meningkatkan serta berbagi kepedulian, peranan dan menjadikan orang tua asuh sebagai sebuah tradisi yang baik.

Dengan mengumpulkan dan mengintegrasikan sumber-sumber yang ada serta dana masyarakat, kami dapat membantu menjamin kelangsungan pendidikan anak-anak dari keluarga kurang mampu, dari orang tua yang cacat dan mereka yang tinggal di tempat-tempat terpencil untuk belajar pendidikan dasar wajib. Kami mempermudah orang tua asuh dengan mengumpulkan dan menyediakan data yang akurat tentang calon anak asuh, anak asuh terkini, calon orang tua asuh dan orang tua asuh terkini. Yayasan GNOTA merupakan organisasi sosial yang profesional, bertanggungjawab serta memiliki reputasi sangat baik dengan jaringan serta hubungan kerja yang luas, mulai dari pemerintah pusat hingga desa-desa di seluruh Indonesia.

2.4 Sistem Pendukung Keputusan

Sistem pendukung keputusan merupakan sistem komputer yang dapat memberikan kamampuan untuk pemecahan masalah maupun kemampuan pengkomunikasian masalah secara terstruktur. Secara khusus sistem pendukung keputusan merupakan sistem yang mendukung pekerjaan seseorang manager maupun sekelompok manager dalam pemecahan masalah semi terstruktur dengan memberikan informasi atau usulan terhadap keputusan tertentu (Sugiyono, 2008).

Menurut Sprague, R.H and Watson H.J komponen-komponen sistem pendukung keputusan terdiri dari :

a. Pengolahan Data (*Database Management*)

   Merupakan bagian dari subsistem data yang teroganisasi dalam basis data. Data yang merupakan sistem pendukung keputusan dapat berasal dari lingkungan luar maupun lingkungan dalam. Untuk kebutuhan sebuah sistem pendukung keputusan diperlukan data yang saling berhubungan dengan suatu permasalahan yang hendak dipecahkan melalui simulasi.

b. Pengolahan Model (*Model Base*)

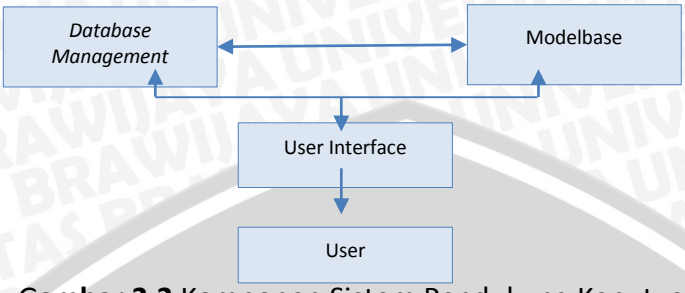
   Merupakan sebuah model yang mempresentasikan suatu permasalahan dari model atau format kuantitatif sebagai dasar pengambilan keputusan, termasuk didalamnya tujuan dari suatu permasalahan, komponen terkait, batas-batasan yang ada, dan hal-hal terkait lainya. Pada pengolahan model atau *model base* memungkinkan pengambilan keputusan menganalisa secara utuh dengan mengambangkan dan melakukan pembandingan solusi alternatif.

c. Pengolahan Dialog (*User Interface*)

   Merupakan penggabungan dari pengolahan data dan pengolahan model yang mana disatukan dalam komponen ketiga yaitu pengolahan dialog, yang sebelumnya sudah dipresentasikan dalam bentuk model yang dimengerti komputer. Pengolahan dialog atau *User interface* menampilkan keluaran sistem

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

bagi user atau penguna dan menerima masukan dari *User* kedalam sistem pendukung keputusan.



Gambar 2.2 Komponen Sistem Pendukung Keputusan
Sumber : Sprague, R.H and Watson H.J (1993)

Dengan alur komponen dapat kita lihat pada Gambar 2.2 yang menunjukkan dari komponen pengolahan data, pengolahan model, pengolahan dialog sampai ke *user*.

2.4.2 Tahapan Proses Pengambilan Keputusan

Tahapan dalam proses pengambilan pada sistem pendukung keputusan terdiri dari beberapa langkah sebagai berikut: (Turban *et al.,2005*)

1. Penelurusan (*Intelligence*)

   Tahap pertama yaitu penelusuran, yaitu merupakan proses pendeteksian dari ruang lingkup problematika proses pengenalan masalah. Data didapatkan diproses dan diuji dalam rangka mengidentifikasi masalah.

2. Perancangan (*Design*)

   Tahap kedua yaitu perancangan yaitu merupakan proses menemukan, mengembangkan dan menganalisa suatu tindakan yang mungkin akan dilakukan, hal ini meliputi pemahaman terhadap suatu masalah dan menguji solusi yang layak.

3. Pemilihan (*Choice*)

   Tahap ketiga yaitu merupakan proses yang mana dibuatnya suatu keputusan nyata dan diambil suatu komitmen untuk mengikuti suatu tindakan tertentu.

4. Implementasi (*Implementation*)

   Pada tahap keempat atau tahap terakhir yaitu merupakan dibuatnya solusi yang dapat direkomendasikan untuk bekerja atau implementasi dari ketiga proses sebelumnya untuk menemukan solusi dari suatu masalah.

2.4.3 Karakteristik Sistem Pendukung Keputusan

Karakteristik dari sistem pendukung keputusan adalah sebagai berikut : (Turban *et al.,2005*)

1. Mendukung pengambilan keputusan dalam membahas masalah-masalah yang terstruktur, semi terstruktur, dan tidak terstruktur.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

2. Mendukung disemua fase proses pengambilan keputusan yaitu penelusuran, perancangan dan pilihan.
3. *Output* ditujukan bagi personil organisasi dalam semua tingkatan.
4. Menggunakan model-model matematis dan statistik yang sesuai dengan pembahasan.
5. Adanya *interface* manusia maupun mesin, yang mana manusia tetap terkontrol dalam proses pengambilan keputusan.
6. Memiliki subsistem yang sudah terintegrasi sedemikian rupa sehingga dapat berfungsi sebagai kesatuan sistem.
7. Memiliki kemampuan dialog untuk memperoleh informasi yang sesuai dengan kebutuhan.
8. Pendekatan *easy to use.* Yang merupakan ciri sebuah sistem pendukung keputusan yang efektif adalah agar saat digunakan memungkinkan keleluasaan *user* untuk memilih atau mengembangkan pendekatan-pendekatan barudalam membahas masalah yang dihadapi.
9. Membutuhkan struktur data yang mampu menerima dengan baik atau komprehensif yang dapat melayani kebutuhan informasi secara keseluruhan pada tingkatan manajemen.
10. Kemampuan sistem untuk beradaptasi secara cepat, dimana pada saat pengambilan keputusan dapat menghadapi masalah-masalah baru dan pada saat yang sama dapat menangani dengan cara mengadaptasikan sistem terhadap kondisi perubahan yang terjadi.

2.4.4 Manfaat Sistem Pendukung Keputusan

Beberapa manfaat yang didapat apabila menggunakan sebuah sistem pendukung keputusan, yaitu :

a. Sistem Pendukung Keputusan dapat memperluas kemampuan dalam pengambilan keputusan dalam proses data bagi pemakainya.
b. Sistem Pendukung Keputusan menghasilkan solusi dari permasalahan yang diambil secara cepat dengan hasil yang dapat diandalkan.
c. Sistem Pendukung keputusan membantu pengambilan keputusan dalam memecahkan masalah terrutama berbagai masalah yang kompleks dan tidak teratur.

2.5 Klasifikasi

Pengertian klasifikasi yaitu suatu pengelompokan objek ke dalam beberapa kelompok berdasarkan variabel yang diamati. Klasifikasi juga dapat diartikan sebagai salah satu teknik yang digunakan dalam melakukan prediksi terhadap kelas atau properti dari sebuah data. Proses klasifikasi dimulai dari pengumpulan

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

data kemudian akan dibuat aturan yang mana akan memisahan beberapa bagian data menjadi ke dalam proses klasifikasi.

Teknik ini dapat memberikan klasifikasi pada data baru dengan memanipulasi data yang ada dan yang telah diklasifikasi dengan menggunakan hasilnya untuk memberikan sejumlah aturan. Aturan tersebut akan digunakan pada masing-masing data baru untuk diklasifikasi (Mujiasih, 2011). Dalam prosesnya klasifikasi dibagi menjadi dua fase, yaitu :

1. Fase *Learning*

   Pada fase ini, menjelaskan bahwa sebagian dari data yang telah diketahui kelas datanya diumpamakan untuk membentuk model perkiraan.

2. Fase *Testing*

   Pada fase ini, menjelaskan dari model yang sudah terbentuk, kemudian diuji dengan sebagian data lainya, hal ini dilakukan untuk mengetahui akurasi dari model yang diuji tersebut. Dan apabila nilai akurasinya mencukupi, maka model ini dapat digunakan untuk melakukan prediksi terhadap kelas data yang belum diketahui.

Model yang sering digunakan karena model yang mudah untuk diinterprestasikan oleh manusia yaitu model *Decision Tree* atau yang berstruktur hirarki. Selain karena itu pembangunannya yang relatif cepat, hasil dari model *Decision Tree* yang dibangun mudah untuk dipahami. Algoritma dari desicion tree yang paling terkenal adalah C4.5., namun akhir-akhir ini sudah mulai dikembangkan sebuah algoritma yang dapat menangani data daam skala yang besar.

Selain model *Decision Tree*, model klasifikasi lainya yang berupa aturan "jika-maka", formula matematis, *Bayesian, Neural Network, Genetic Algorithm, Fuzzy, Case Based Reasoning* dan *K-nearest Neighbor*. Metode SVM (*Support Vector Machine*) juga merupakan salah satu metode yang dapat digunakan untuk proses klasifikasi. Metode ini berbasis *machine learning* yang memiliki nilai akurasi baik dalam proses data mining khususnya klasifikasi.

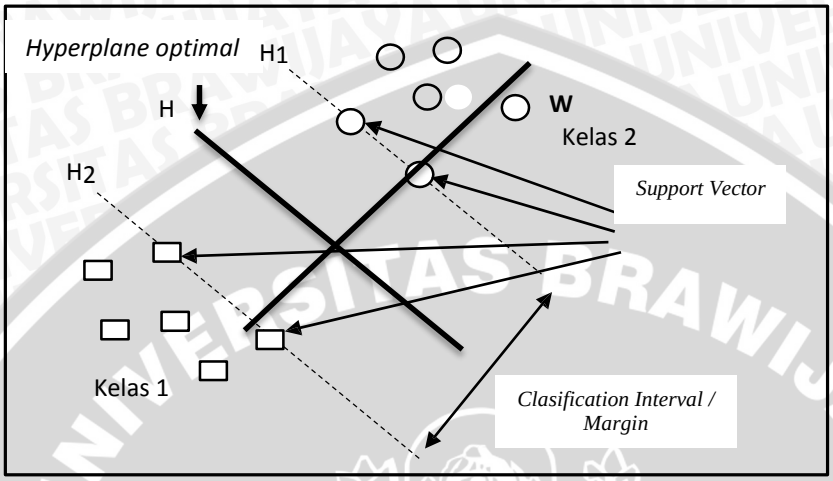
2.6 Support Vector Machine (SVM)

SVM (*Support Vector Machine*) adalah seperangkat metode pembelajaran mesin pada teori stastistika Vladimir Vapnik (Kartal and Cebi, 2013). Metode SVM merupakan salah satu metode yang biasanya digunakan untuk pemilihan variabel dan proses klasifikasi data (Noviyanti and Purnami, 2012). Teknik SVM dilakukan dengan cara analisis berdasarkan hitungan matematis sehingga dapat membantu di dalam pemecahan masalah. Teknik yang ada dalam memecahkan masalah yang menggunakan SVM adalah dengan menentukan suatu *hyperplane* yang terbaik. *Hyperplane* merupakan suatu bidang keahlian pemisah yang terletak ditengah antara dua set obek dari dua kelas.

Support Vector Machine dapat melakukan proses generalisasi untuk menentukan pola tertentu berdasarkan data *testing* dan data *training*. SVM juga

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

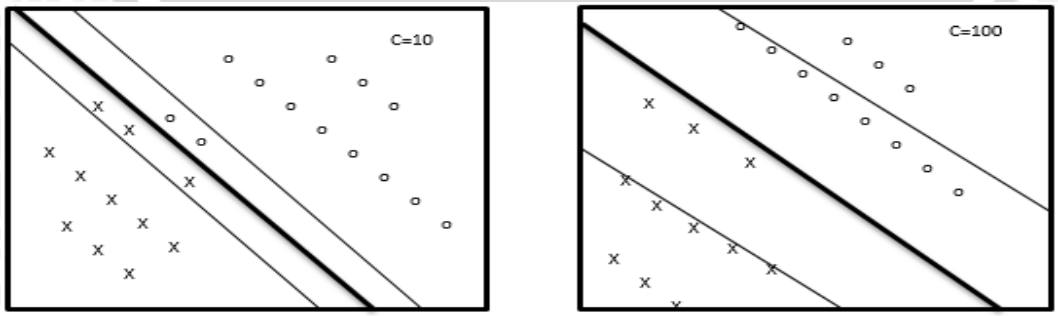
cepat dalam menyelesaikan suatu masalah karena metode SVM memiliki struktur sederhana. Metode SVM ini mempunyai kelebihan menemukan fungsi pemisah atau pengklasifikasian yang optimal (Damayanti *et all,* 2010). Penerapan metode SVM dapat digunakan hanya untuk klasifikasi dua kelas dan selalu memberikan hasil yang sama dalam setiap kali melakukan proses.



Gambar 2.3 Penerapan Metode SVM

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

Gambar 2.3 menunjukan dari penerapan metode SVM dengan dua dimensi. Simbol lingkaran merupakan H_1 dan simbol persegi merupakan H_2 , yang mana H_1 dan H_2 dipindahkan oleh *hyperlane* (H). Jarak antara kedua jenis sampel (*Support Vector)* yang dipisahkan oleh *hyperplane* disebut sebagai *classification interval* atau *margin.* Dengan vektor terdekat yang memisahkan *hyperplane* disebut dengan *support vector*. Menurut Yang Yi Du *et all,* karakteristik metode klasifikasi dengan menggunakan SVM (*Support Vector Machine*) dibagi menjadi dua, yaitu *linear* dan *non-linear.* SVM yang dikembangkan menggunakan *linear* yaitu dengan memisahkan kedua *class* secara terpisah oleh *hyperplane* yaitu dengan teknik *soft margin.* Sedangkan *non-linear* yaitu menerapkan fungsi *kernel trick* terhadap ruang berdimensi tinggi (Novianti, Purnami, 2012). Adapun menurut C.Corttes dan V.Vapnik,1995 bahwa jika nilai $C > 0$ relatif penting unutk memaksimumkan nilai *margin*, dan meminimumkan jumlah *slack,* adapun penjelasaanya dapat dilihat pada Gambar 2.4.



Gambar 2.4 Pengaruh nilai *C*

Sumber: Asa Ben-hur, Jason Weston (2008)

Berdasarkan Gambar 2.4 bahwa pengaruh nilai *C* terhadap memaksimumkan nilai *margin*, dan meminumkan jumlah *slack,* semakin besar nilai *C* maka lebar margin akan melebar begitu juga sebaliknya, jika nilai margin semakin kecil maka lebar *margin* juga akan menyempit atau mengecil.

2.6.1 *Support Vector Machine Linear*

Support Vector Machine Linear yang menggunakan metode SVM dibagi menjadi dua jenis, yang terdiri dari *Sparable dan Non-separable* (Novianti, Purnami, 2012).

Dari dua jenis klasifikasi pada linear tersebut, SVM melakukan klasifikasi data latih yang berpasangan dari dua kelas. Klasifikasi linear merupakan data yang dapat dipisah secara *linear.* Contohnya diketahui nilai x_i = *{x1,..., xn}* merupakan titik pada dataset, dan $y_i = y \in \{+1, -1\}$ merupakan kelas pada data x_i. Bidang pemisah yang terbaik adalah yang dapat memisahkan semua data set yang sesuai dengan kelas dan memiliki *margin* yang paling besar. Bidang pemisah bagian pertama membatasi kelas pertama, sedangkan bagian pemisah kedua membatasi kelas kedua, sehingga diperoleh persamaan (2.1) dan persaman (2.2).

$$(w.x_i + b) \geq +1 \; untuk \; y_i = +1 \quad (2.1)$$

$$(w.x_i + b) \geq -1 \; untuk \; y_i = -1 \quad (2.2)$$

Keterangan :
w = Vector yang tegak lurus dengan Hyperplane.
b = Nilai bias.
x_i = data ke-i.
y_i = Kelas sampel data ke-i

Variabel w merupakan bidang normal antara bidang pemisah terhadap pusat koordinat, dan variabel b merupakan posisi bidang relatif terhadap pusat koordinat. Jarak *hyperplane* dan titik terdekat untuk menentukan *margin* terbesar adalah dengan memaksimalkan fungsi $\frac{1}{\|w\|}$ atau dengan meminimalkan $\|w\|^2$. Bidang pemisah atau *hyperplane* dari kedua kelas dijelaskan dengan persamaan (2.3).

$$y_i(w.x_i + b) \geq 1, i = 1,2, \dots, n \quad (2.3)$$

Hal ini dapat dirumuskan sebagai *Quadratic Programming* dengan mencari titik minimal. *Hyperplane* terbaik dengan *margin* terbesar dapat dirumuskan sebagai masalah optimasi *constrain* pada persamaan (2.4).

$$Min = \frac{1}{2} \|w\|^2 \quad (2.4)$$

Dengan syarat $y_i(w.x_i + b) \geq 1$, persamaan (2.4) untuk menyelesaikan permasalahan optimasi yang bisa diselesaikan dengan menggunakan fungsi *Lagrange*. Fungsi *Lagrange* yang digunakan disini adalah *Lagrange multiplier*, yaitu seperti pada persamaan (2.5).

$$Min\ L(w,b,\alpha) = \frac{1}{2}\ \|w\|^2\ \sum_{i=1}^{n} \alpha_i\ [y_i(w.x_i + b) - 1] \qquad (2.5)$$

Dimana α_i merupakan nilai bobot dari setiap titik data. Yang mana notasi ini memiliki nilai nol atau positif jika ($\alpha_i\ \geq 0$), dengan miminimalkan nilai *L* terhadap *w* dan *b* berdasarkan permasamaan (2.6).

$$w = \sum_{i=1}^{n} \alpha_i\ y_i x_i \qquad b = -\frac{1}{2}(w.x^+ + w.x^-) \qquad (2.6)$$

Dimana x^+ merupakan nilai dari salah satu *support vector* dari *class positive.* Sedangkan nilai x^- merupakan nilai dari salah satu *support vector* dari *class negative.* Pada persamaan (2.6) bisa dimodifikasi menjadi fungsi maksimal yang bertujuan mencari *hyperplane* terbaik dan dapat dirumuskan dalam persamaan (2.7).

$$Max\ \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i\ \alpha_j y_i y_j x_i x_j$$

dengan batasan (2.7)

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

Hasil dari perhitungan menggunakan persamaan (2.7) diperoleh nilai α_i bernilai positif yang nanti akan digunakan untuk menemukan *w.* Terdapat α_i untuk setiap data *training* yang memiliki nilai $\alpha_i = 0$. dengan demikian, fungsi keputusan yang dihasilkan hanya dipengaruhi oleh *support vector.* Untuk mendapatkan klasifikasi yang optimal data *testing x* dapat digunakan persamaan (2.8). hasil klasifikasi dapat dirumuskan dipersamaan (2.9).

$$f(x) = (w.x + b) \text{ atau } \sum_{i=1}^{n} \alpha_i y_i x_i\ .x + b \qquad (2.8)$$

$$\text{Fungsi klasifikasi} = sign\ (f(x)) \qquad (2.9)$$

Keterangan variabel *m* adalah jumlah *support vector* atau titik data yang memiliki α_i > 0, dan nota *x* merupakan data yang akan diklasifikasikan.

2.6.2 *Support Vector Machine Non-Linear*

Support Vector Machine Non-Linear ini dapat ditranformasikan dalam permasalahan *linear* dengan menggunakan dimensi yang tinggi (Kartal dan Cebi, 2013). Sebuah fungsi dari *kernel* dapat didefinisikan menjadi *input kernel trick. Kernel trick* merupakan bagian dari pembelajaran dalam metode *Support Vector Machine,* dimana untuk mengetahu suatu fungsi *kernel* tanpa harus mengetahui wujud dari fungsi *non-linear. Kernel trick* dapat dirumuskan pada persamaan (2.10).

$$K(x_i, x_j) = \emptyset(x_i).\emptyset(x_j) \qquad (2.10)$$

Notasi x_i, x_j dengan *dot product* bisa diganti dengan simbol *K,* untuk mendapat *hyperplane* yang optimal dapat menggunakan model *quadratic problem* yang

dijelaskan sebelumnya, sperti pada persamaan berikut. Klasifikasi pada non-linear yang dijelaskan menggunakan persamaan (2.11).

$$\min \sum_{i=1}^{n} a_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \, y_i K(x_i . x_j) \tag{2.11}$$

Dengan batasan berikut

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

Nilai dari *K* merupakan fungsi kernel, menunjukan bahwa pemetaan *linear* pada *feature space.* Dengan demikian diporelahlah solusi untuk fungsi keputusan klasifikasi yang optimal pada persamaan (2.12).

$$f(x) = sign(\sum_{i=1}^{n} \alpha_i \, y_i K(x, x_i) + b) \tag{2.12}$$

Keterangan :
$f(x)$ = Data Uji.
n = Jumlah *support vector.*

Masing-masing *kernel* bisa digunakan dalam percobaan yang berfungsi sebagai paramater *kernel* dan memberikan keakuratan yang terbaik dalam proses klasifikasi. Pada proses metode SVM dimulai dengan perhitungan fungsi dari *kernel.* Adapun beberapa fungsi *kernel* yang dapat digunakan, sebagai berikut:

a. Fungsi *linear* didefinisan pada persamaan (2.13).
$$K(x, x_i) = x . x_i \tag{2.13}$$
b. Fungsi *polynomial* didefiniskan pada persamaan (2.14).
$$K(x, x_i) = (x . x_i)^d \tag{2.14}$$
c. Fungsi *gaussian* RBF didefinisakan pada persamaan (2.15).
$$K(x, x_i) = exp\left(-\frac{\|x - x_i\|^2}{2\sigma^2}\right) \tag{2.15}$$
d. Fungsi *sigmoid* didefinisikan pada persamaan (2.16).
$$K(x, x_i) = \tanh(\sigma(x . x_i) + C) \tag{2.16}$$

Kernel linear sehingga dapat digunakan ketika data yang akan diklasifikasikan dapat terpisahkan dengan sebuah garis atau *hyperplane.* Sedangkan *kernel non-linear* dapat digunakan ketika datanya dapat dipisahkan dengan garis lengkung atau pasa bidang ruang yang berdimensi tinggi.

2.7 Multi-Class SVM

Metode SVM pertama kali dikenalkan oleh Vapnik, yang mana dulu SVM hanya dapat digunakan untuk pengklasifikasian data kedalam dua kelas yaitu *binary classifier.* Metode SVM dapat diterapkan untuk pengklasifikasian data yang bersifat *multi-class*. Yang mana pada implementasinya ada beberapa pendekatan yang umum dilakukan. Pendekatan pertama yaitu dengan cara menggabungkan beberapa SVM biner dan pendekatan kedua yaitu dengan cara menggabungkan

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

semua data yang terdiri dari beberapa kelas ke dalam permasalahan mengenai optimasi (Novianti,Purnami, 2012).

Pada penggunaannya, untuk pendekatan yang kedua jauh lebih rumit dalam permasalahan optimasi yang harus diselesaikan. Yang mana pada pendekatan pertama pada umumnya menggunakan metode untuk mengimplementasikan *multi-class* SVM antara lain dengan metode *one-against-all, one-against-one,* dan *Directed Acylic Graph Support Vector Machine (*DAGSVM*).* Pada penelitian ini nantinya menggunakan metode yang dapat memecahkan permasalahan pengklasifikasian dengan *multi-class* sehingga dari permasalahan tersebut dapat menggunakan metode *one-againts-all.*

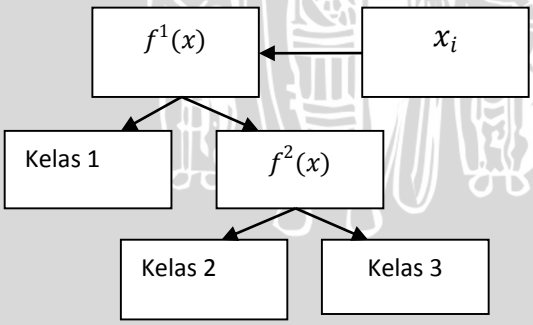
2.7.1 *Metode one-against-all*

Nantinya pada penelitian ini menggunakan metode *one-against-all* yang menangani permasalahan penentuan *multi-class.* Dengan menggunakan metode ini yang dibangun berdasarkan sejumlah k SVM biner (k = jumlah kelas). SVM ke-*i* dilatih dengan menggunakan seluruh data sampel yang terdapat pada ke-*i* dengan memiliki kelas label positif (+1) dan kelas label negatif (-1). Jika terdapat suatu permasalahan dalam klasifikasi 3 buah kelas, untuk pelatihan kelas-kelas tersebut menggunakan 3 buah SVM biner yang ditunjukan pada Tabel 2.2 dan Gambar 2.5.

Tabel 2.2 *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: Yang Yi Du, Quishi, Zhao Jinying (2010)



Gambar 2.5 Klasifikasi *one-against-all* untuk 3 kelas

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

2.7.2 *Metode one-againts-one*

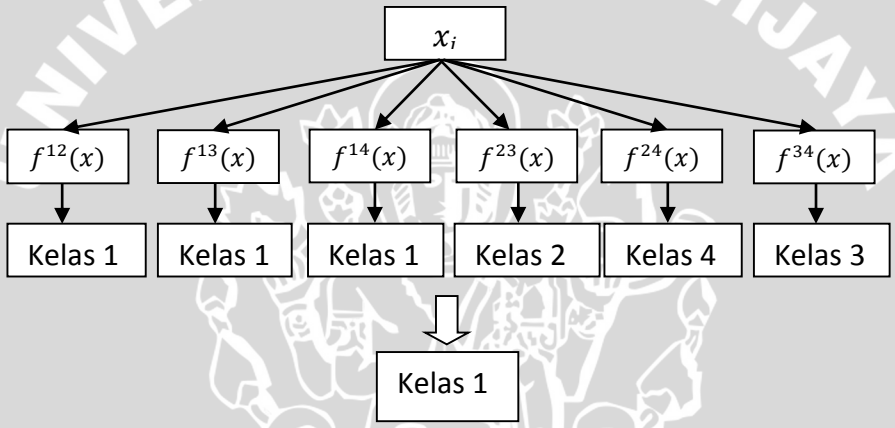
Metode *one-against-one* dibangun berdasarkan nilai $k(k-1)/2$ buah model klasifikasi SVM biner. Metode ini dilatih dari dua kelas yaitu kelas ke-*i* dan kelas ke-*j,* dilakukan pencarian solusi untuk menghadapi persoalan optimasi konstrain. Terdapat beberapa metode untuk m=pengujian setelah keseluruhan

terhadap $k(k-1)/2$ model klasifikasi setelah selesai dibangun. Salah satu dari metode tersebut adalah metode voting yang ditunjukan tabel 2.3 dan Gambar 2.6.

Tabel 2.3 metode *one-against-one* dengan contoh 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: Yang Yi Du, Quishi, Zhao Jinying (2010)



Gambar 2.6 Klasifikasi menggunakan metode *one-againts-one*

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

Gambar 2.6 menunjukan data x_i dapat dilakukan klasisifikasi sebagai kelas 1 karena kelas 1 mempunyai suranya terbanyak dari hasil voting, jika x_i dimasukan kedalam fungsi hasil pada persamaan (2.17).

$$f(x) = (w^{ij})\emptyset(X_i) + b \tag{2.17}$$

Dari persamaan (2.17) tersebut hasilnya menyatakan bahwa data x_i adalah kelas 1 ditambah satu.kelas dari x_i akan ditentukan dari jumalah suara terbanyak. Jika ditemukan terdapat dua kelas yang memiliki jumlah sura yang sama, maka kelas yang indeksnya lebih kecil dinyatakan sebagai kelas data x_i. jadi pada metode *one-against-one* ini terdapat $k(k-1)/2$ buah untuk permasalahan *quadratic programming* yang masing-masing memiliki nilai $2n/k$ variabel (n merupakan jumlah data pelatih).

2.8 Metode *Sequential Training* pada SVM

Hyperplane yang bisa bekerja optimal pada metode SVM dapat ditemukan dengan merumuskan *quadratic problem* dan selesaikan dengan analsisa numerik.

Metode yang dikembangkan Vijaya Kumar untuk mencari nilai *a,* dapat dijelaskan dalam beberapa tahap sebagai berikut (VijayaKumar,1999):

1. Tahap awal apabila perhitungan *kernel* telah dilakukan, dengan proses selanjutnya melakukan inisialisasi untuk parameter SVM, misalnya nilai $\alpha_i = 0$.
2. Langkah kedua menghitung komponen matriks pada persamaan (2.18)

   $$D_{ij} = y_i y_j (K(x_i x_j) + \lambda^2) \quad (2.18)$$

   Notasi merupakan data ke-*i* dan data ke-*j. Notasi y merupakan kelas dari* ke-*i* dan data ke-*j* . *K(*$x_i x_j$*)* merupakan fungsi kernel yang digunakan. Lakukan lahkah 1 dan 2 untuk *i, j* = 1,2,…,*n.*
3. Untuk setiap *i,* =1,2,…,*n* dihitung dengan menggunakan persamaan (2.19) sampai (2.20).

   a. $E_i = \sum_{j=1}^{n} \alpha_i D_{ij}$ (2.19)

   b. $\delta\alpha_i = \min\{\text{maks}[\gamma(1 - E_i), -\alpha_i], C - \alpha_i\}$ (2.20)

   $\delta\alpha_i$ merupakan variabel tunggal, bukan bentuk dari perkalian dari δ dan α_i yang ditunjukkan pada persamaan (2.21).

   c. $\alpha_i = \alpha_i + \delta\alpha_i$ (memperbarui nilai α_i) (2.21)

   Notasi γ adalah parameter untuk mengontrol kecepatan proses *learning (learning rate).* Nilai konstanta untuk parameter γ, dapat dilihat pada persamaan (2.22)

   $$\gamma = \frac{konstanta\ \gamma}{\max\limits_{\{i\}} D_{ii}} \quad (2.22)$$

   $max_{\{i\}} D_{ii}$ merupakan nilai maksimum yang diperoleh dari nilai dialog pada matriks. Dengan menggunakan nilai konstanta untuk nilai yang diinputkan oleh pengguna dalam pengujian 0.0001. fungsi dari δa_i merupakan fungsi konvergensi untuk memantau perubahan suatu fungsi dari *Lagrange multiplier.* Jika data latih telah mencapai nilai konvergen $Max(|\delta\alpha_i|) < \varepsilon$, dan ketika nilai maksimum dari iterasi mencapai nilai yang ditentukan maka iterasi akan dihentikan.

Keterangan :

D_{ij} = Matriks yang menampung perkalian *kernel.*

α_i = Bobot setiap titik data.

$\delta\alpha_i$ = Variabel tunggal.

E_i = Bentuk Perkalian α_i dan D_{ij}

ε = *Epsilon,* digunakan untuk *error* klasifikasi

4. Langkah selanjutnya mengulang nilai α untuk mencapai nilai konvergen. Konvergen dapat didefinisikan dari tingkat perubahan pada nilai α.

5. Terakhir mendapatkan nilai *Support Vector* dapat diketahui dengan *SV* = ($\alpha_i > ThresholdSV$). Nilai *ThresholdSV* ditentukan dari hasil beberapa kali melakukan uji coba. Nilai *ThresholdSV* tersebut adalah *ThresholdSV* ≥ 0.

2.9 Metode *Technique for Order Preference by Similarity to Ideal Solution* (TOPSIS)

Metode TOPSIS adalah metode klasik MCDM yang mana m alternatif dihitung dengan n atribut dan setiap masalah dianggap sebagai sistem geometris yang terdiri dari m titik dan n dimensi ruang (Yang, 2007). TOPSIS adalah multi atribut dalam teknik pengambilan keputusan yang diterima secara luas karena logika yang bagus, pertimbangan simultan yang ideal dan solusi anti-ideal dan prosedur perhitungan yang mudah diprogram (Onüt dan Soner, 2008). Dan menurut kusuma mukhtar atmaja(2010) metode TOPSIS memperhatikan dengan baik jarak ke solusi ideal positif maupun jarak ke solusi ideal negatif dengan mengambil hubungan kedetakan menuju solusi ideal, dengan melakukan perbandingan pada keduanya, urutan pilihan dapat ditentukan.

Metode TOPSIS ini biasanya sering digunakan pada beberapa model *Multi Attribute Desicion Making* (MADM) dikarenakan metode TOPSIS ini memiliki beberapa kelebihan, yaitu (Kusuma,2010) :

a. Komputasinya efisien.

b. Konsepnya yang sederhana dan mudah dimengerti.

c. Memiliki kemampuan untuk mengukur kinerja yang relatif dari alternatif-alternatif yang mendukung keputusan dalam bentuk matematis yang sederhana.

Adapun persamaan untuk TOPSIS dari matriks *D* yang memiliki nilai *m* alternatif dengan *n* sebagai kriteria, dimana x_{ij} merupakan pengukuran pilihan dari alternatif ke-*i* dalam hubungannya dengan kriteria ke-*j* (Ding,2011), untuk mencari matriks *D* maka persamaanya ditunjukkan pada persamaan (2.23):

$$D = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1m} \\ x_{21} & x_{21} & \dots & x_{2m} \\ \vdots & & & \vdots \\ x_{m1} & x_{m1} & \dots & x_{mn} \end{pmatrix} \tag{2.23}$$

Adapun beberapa langkah-langkah untuk menyelesaikan masalah dengan menggunakan metode TOPSIS adalah sebagai berikut (Ding,2011):

1. Normalisasi Matriks Keputusan

   Dalam langkah normalisasi ini, setiap atribut diubah menjadi nilai yang *camparable,* setiap normalisasi dari nilai r_{ij} bisa dilakukan dengan perhitungan membagi nilai setiap elemen pada matriks, dengan jumlah akar kuadrat dari nilai elemen, dapat di lihat pada persamaan (2.24).

   $$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^{m} x_{ij^2}}} \tag{2.24}$$

   Untuk *i* = 1,2,3,....,*m*

$j = 1,2,3,\ldots,m$

Keterangan:

r_{ij} = normalisasi matriks pada baris ke-*i* dan kolom ke-*j*

x_{ij} = nilai data pada baris ke-*i* dan kolom ke- *j*

$\sqrt{\sum_{i=1}^{m} x_{ij^2}}$ = akar dari jumlah baris ke *i* kolom ke *j* kuadratkan

Dari persamaan (2.24) sehingga didapatkan matriks R hasil normaslisasi ditunjukan pada persamaan (2.25).

$$r_{ij} = \begin{pmatrix} r_{11} & r_{12} & \cdots & r_{1j} \\ r_{21} & r_{22} & \ldots & r_{2j} \\ \vdots & & & \vdots \\ r_{i1} & r_{i1} & \ldots & r_{ij} \end{pmatrix} \quad (2.25)$$

Yang mana *R* merupakan matriks yang telah dinormalisasi, dimana m merupakan alternatif, *n* menyatakan sebuah kriteria dan r_{ij} adalah normalisasi pengukuran pilihan dari alternatif ke-*i* dalam hubungannya kriteria ke-*j*.

2. Pembobotan Pada Matriks yang telah dilakukan Normalisasi
   Setelah dilakukannya tahap pertama yaitu normalisasi, maka setiap kolom matriks R dikalikan dengan bobot-bobot (*wj*) yang telah ditentukan oleh si pembuat keputusan seperti pada pada persamaan (2.26). sehingga, *weighted normalized matrix* atau matriks ternomalsisasi terbobot ditunjukan pada persamaan (2.26)

$$V = \begin{pmatrix} w_1 r_{11} & w_1 r_{12} & \cdots & w_1 r_{1j} \\ w_1 r_{21} & w_1 r_{22} & \ldots & w_1 r_{2j} \\ \vdots & & & \vdots \\ w_1 r_{i1} & w_1 r_{i1} & \ldots & w_1 r_{ij} \end{pmatrix} \quad (2.26)$$

   Dengan W merupakan bobot prioritas, bentuk dari matriks W ditunjukkan pada persamaan (2.27).

$$W = \begin{pmatrix} w_1 & 0 & \cdots & 0 \\ 0 & w_2 & \ldots & 0 \\ \vdots & & & \vdots \\ 0 & 0 & \ldots & w_n \end{pmatrix} \quad (2.27)$$

$$v_{ij} = w_j \cdot r_{ij} \quad (2.28)$$

   Keterangan:

   V_{ij} = matriks ternormalisasi terbobot baris ke-*i* dan kolom ke-*j*

   W = bobot prioritas

   r_{ij} = nilai data pada baris ke-*i* dan kolom ke-*j*

   w_1 = bobot prioritas ke-1

3. Menentukan solusi ideal positif dan negatif
   Untuk menentukan solusi ideal positif dan negatif maka untuk solusi ideal dinotasikan dengan A^+ sedangkan untuk solusi ideal negatif dinotasikan dengan A^+. Cara untuk mencari solusi ideal positif (A+) dilakukan dengan

pencarian nilai maksimum pada matriks normalisasi terbobot untuk setiap kriterianya, adapun untuk pencarian solusi ideal positif ditunjukan pada persamaan (2.29).

$$A^{+} = \{(\max v_{ij} \,|j \in J), (\min v_{ij} \,|j \in J'), i = 1,2,3, \dots, m\} \tag{2.29}$$

$$= \{v_1^{+}, v_2^{+}, \dots, v_n^{+}\}$$

Dan untuk mencari solusi ideal negatif (A-), dilakukan dengan pencarian nilai minimum pada matriks normalisasi terbobot untuk setiap kriterianya. Adapun untuk pencarian solusi ideal negatif ditunjukan pada persamaan (2.30).

$$A^{-} = \{(\min v_{ij} \,|j \in J), (\max v_{ij} \,|j \in J'), i = 1,2,3, \dots, m\} \tag{2.30}$$

$$= \{v_1^{-}, v_2^{-}, \dots, v_n^{-}\}$$

Dengan keterangan :

A^{+} = Solusi ideal positif/nilai maksimum dari matrik ternormalisasi terbobot

A^{-} = Solusi ideal negative/nilai minimum dari matrik ternormalisasi terbobot

v_{ij} = nilai matriks normalisasi terbobot baris ke-*i* dan kolom ke-*j*

4. Menghitung *separation measure*

   separation measure merupakan pengukuran jarak dari suatu alternatif ke solusi ideal positif dan solusi ideal negatif. Perhitungan matematisnya untuk menghitung *separation measure* adalah sebagai berikut :

   a. *separation measure* untuk menghitung solusi ideal positif dilakukan dengan cara perhitungan akar dari matriks ternormalisai yang terbobot kemudian dikurangi dengan solusi ideal positif, yang mana ditunjukkan pada persamaan (2.31).

   $$s_i^{+} = \sqrt{\sum_{i=1}^{n}(v_{ij} - v_i^{+})^2}, \text{ untuk } i = 1,2,3,\dots,m \tag{2.31}$$

   b. *separation measure* untuk perhitungan solusi ideal negatif dilakukan dengan cara perhitungan akar dari matriks ternormalisasi yang terbobot kemudian dikurangi dengan solusi ideal negatif, yang mana ditunjukkan pada persamaan (2.32).

   $$s_i^{-} = \sqrt{\sum_{i=1}^{n}(v_{ij} - v_i^{-})^2}, \text{ untuk } i = 1,2,3,\dots,m \tag{2.32}$$

   Keterangan:

   s_i^{+} = jarak antar alternatif dengan solusi ideal positif

   s_i^{-} = jarak antar alternatif dengan solusi ideal negatif

5. Menghitung kedekatan relatif terhadap solusi ideal
   Untuk tahap ini proses perhitungan selanjutnya yaitu menghitung kedekatan relatif dari alternatif terhadap solusi ideal, dikatakan pada alternatif A_i dekat dengan solusi ideal A^+ apabila C_i^+ mendekati 1. Sehingga diperoleh perhitungan kedekatan relatif terhadap solusi ideal pada persamaan (2.33).

   $$C_i^+ = \frac{S_i^-}{S_i^- + S_i^{-\prime}} \text{ dimana } 0 < C_i^+ < 1 \text{ dan } i = 1,2,3,\ldots,m \quad (2.33)$$

   Jika alternatif Ai, dekat terhadap solusi ideal apabila C_i^+ mendekati 1. Jadi $C_i^+ = 1$ jika $A_i = A^+$ dan $C_i^- = 0$, jika $A_i = A^-$.

   Keterangan:
   C_i^+ = nilai preferensi atau kedekatan relatif.
   s_i^- = jarak solusi negatif
   s_i^+ = jarak solusi postif

6. Mengurutkan pilihan
   Langkah terakhir yaitu mengurutkan pilihan berdasarkan nilai C_i^+, sehingga alternatif dengan jarak terpendek terhadap solusi ideal adalah alternatif terbaik.

2.10 Evaluasi

Evaluasi yaitu merupakan tahapan yang dilakukan untuk mengetahui tingkat akurasi dan hasil dari klasifikasi dengan cara menghitung jumalah *record* uji yang mana kelasnya diprediksi secara akurat. Pada metode SVM evaluasi performansi yang digunakan pada penelitian ini adalah *classification accuracy, classification accuracy* merupakan ukuran suatu ketepatan dari kejadian-kejadian yang diinginkan (Purnami dan Embong,2009). Adapun persamaan dari *classification accuracy* terdapat pada persamaan (2.34)

$$Classification\ Accuration = \frac{TP+TN}{TP+FP+FN+TN} \quad (2.34)$$

Persamaan 2.34 dapat ditentukan dengan menggunakan nilai yang sudah ditentukan pada *confusion matrix, confusion matrix* merupakan klasifikasi aktual dan prediksi pada sistem klasifikasi, adapun *confusion matrix* 2x2 ditunjukan pada Tabel 2.4.

Tabel 2.4 *Confusion Matrix* 2x2

	Predicated Class	
	Positive	*Negative*
Actual Class Positive	*True Positive* (TP)	*False Negative* (FN)

Actual Class Negative	*False Positive* (FP)	*True Negative* (TN)

Adapun istilah pada persamaan 2.34 untuk proses pengujian klasifikasi, sebagai berikut :

a. TP merupakan jumlah nilai *positive* yang diklasifikasikan *positive.*
b. TN merupakan jumlah nilai *negative* yang diklasifikasikan sebagai *negative.*
c. FP merupakan nilai *negative* yang diklasifikasikan sebagai *positive.*
d. FN merupakan nilai *positive* yang diklasifikasikan sebagai negative.

Sedangkan Persamaan 2.33 juga dapat digunakan *untuk confusion matrix* 3x3 ditunjukan pada Tabel 2.5.

Tabel 2.5 *Confusion Matrix* 3x3

		Predicated 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 :

$TP = X_{11} + X_{12} + X_{13}$

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

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

$FN = (X_{22} + X_{33}) + (X_{11} + X_{33}) + (X_{11} + X_{22})$

Dengan jumlah nilai prediksi benar adalah jumlah *record* data uji yang diprediksi kelasnya menggunakan metode klasifikasi dan hasilnya sama dengan kelas sebenarnya. Sedangkan total keseluruhan prediksi adalah jumlah keseluruhan *record* yang diprediksi kelasnya atau dari seluruh data uji.

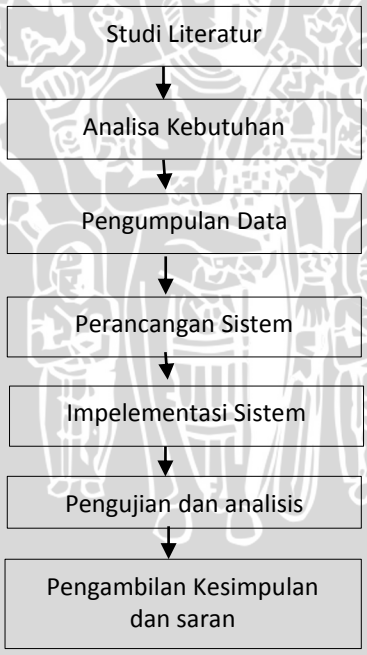
repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

BAB 3 METODOLOGI

Pada bab metodologi dijelaskan langkah-langkah yang akan ditempuh dalam penyusunan skripsi Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri dengan Implementasi Metode SVM dan TOPSIS yaitu perancangan, implementasi dan pengujian dari sistem yang akan dibuat. Tahap penelitian disini akan membahas langkah-langkah Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri Adapun tahapannya adalah sebagai berikut :

1. Melakukan studi literatur.
2. Melakukan analisa Kebutuhan.
3. Melakukan pengumpulan data.
4. Melakukan perancangan sistem.
5. Melakukan implementasi sistem
6. Melakukan pengujian dan analisis.
7. Melakukan pengambilan kesimpulan dan saran.

Diagram alir tahapan penelitian pada gambar 3.1.



Gambar 3.1 Diagram Alir Tahapan Penelitian

Berdasarkan gambar 3.1 adapun tahapan metodologi dapat dijelaskan sebagai berikut :

3.1 Studi Literatur

Studi literatur dilakukan dengan cara mengumpulkan serta mempelajari literatur-literatur yang berkaitan maupun yang mendukung untuk membangun

Implementasi Metode SVM dan TOPSIS untuk Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri.

1. Metode SVM (*Support Vector Machine*).
2. Metode TOPSIS (*Technique For Others Reference by Similarity to Ideal Solution*).
3. Sistem Pendukung Keputusan.
4. Evaluasi.

Studi literatur yang digunakan pada penelitian ini bersumber dari jurnal internasional maupun nasional, buku, karya ilmiah dan artikel di media internet. Selain itu dilakukkan wawancara secara langsung kepada lembaga GNOTA Kediri.

3.2 Analisis Kebutuhan

Analisa kebutuhan bertujuan untuk mengetahui kebutuhan yang diperlukan dalam membangun Implementasi Metode SVM dan TOPSIS untuk Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri. Adapun kebutuhan yang digunakan dalam implementasi penelitian terdiri dari kebutuhan *hardware*, kebutuhan *software* dan kebutuhan data, adapun setiap kebutuhan terdiri dari :

1. Kebutuhan *hardware,* yaitu :
   a. *Laptop dengan memory* RAM 4 GB.
   b. Posesor Intel(R) Core(TM) i5-4200U 1.6GHz 2.3 GHz.
   c. *Harddisk* dengan kapasitas 1 TB.
2. Kebutuhan *software,* yaitu :
   a. Sistem operasi Microsoft Windows 8.1 *Single language.*
   b. Mysql sebagai DBMS (*DataBase Management System*).
   c. XAMPP *Control Panel* V3.2.1.
   d. NetBeans IDE Versi 7.3
   e. Bahasa Pemograman Java.
3. Kebutuhan data, yaitu :

   Adapun kebutuhan data untuk penelitian ini yakni didapat langsung dari Lembaga GNOTA Kediri melalui Wawancara yang berupa data kuantitatif dan kualitatif dengan cara pengajuan permohonan beasiswa dan data parameter yang akan digunakan untuk proses seleksi. Analisa kebutuhan dalam penelitian dapat dilihat pada tabel 3.1 .

Tabel 3.1 Analisa Kebutuhan Data

No	Kebutuhan Data	Sumber Data	Metode	Kegunaan Data

1	Data pengajuan beasiswa	Lembaga GNOTA Kediri	Permohonan data pengajuan beasiswa	Sebagai data utama yang akan dilakukan seleksi
2	Data penerima beasiswa	Lembaga GNOTA Kediri	Permohonan data penerima beasiwa	Data penerima beasiswa Lembaga GNOTA yang sudah diketahui layak tidaknya menerima beasiswa sebagai data latih dalam menentukan nilai bobot kepentingan kriteria
3	Data parameter penilaian	Lembaga GNOTA Kediri	Permohonan data parameter penilaian	Sebagai bahan acuan untuk parameter penilaian dalam proses seleksi

3.3 Pengumpulan Data

Pengumpulan data untuk penelitian ini dilakukan dengan cara studi pustaka, observasi dan melakukan wawancara terhadap GNOTA Kediri. Data yang didapat adalah data penerima beasiswa GNOTA tahun 2015 dan kriteria penilaian. Adapun tahapan pengumpulan data pada penelitian ini terdiri dari :

1. Studi Pustaka

   Studi pustaka dilakukan guna menunjang metode pada penelitian yaitu dalam pengumpulan data maupun pada metode SVM dan TOPSIS.

2. Observasi

   Observasi dilakukan dengan cara melakukan observasi ke lembaga GNOTA yang berada di Kediri guna memperoleh data yang dibutuhkan dalam penelitian ini sebagai bahan pertimbangan dalam menentukan rekomendasi penerima beasiswa agar menghasilkan perhitungan dengan nilai akurasi yang lebih baik.

3. Wawancara

   Wawancara dilakukan terhadap pegawai GNOTA yang menangani data pendaftar seleksi beasiswa, setelah mendapatkan persetujuan dari BANGKESPOL Kabupaten Kediri dan Kepala Lembaga GNOTA Kediri, pada proses wawancara menanyakan seputar alur proses seleksi, serta mendapatkan data pendaftar beasiswa berupa *softcopy* dan *hardcopy*.

3.4 Perancangan Sistem

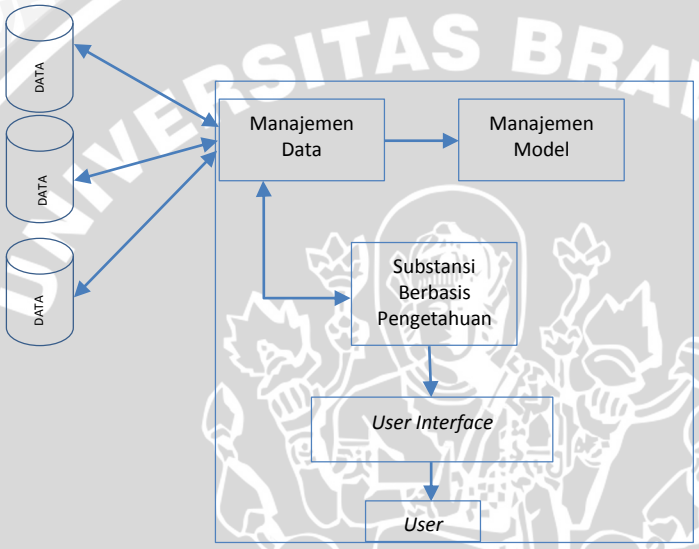
Perancangan dari Implementasi Metode SVM dan TOPSIS untuk Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri merupakan tahap

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

dimana dijelaskannya proses perancangan mulai dari kebutuhan, desain sistem secara keseluruhan baik itu dari modelnya, arsitektur yang menunjang sistem. Perancangan sistem nantinya berdasarkan hasil yang diperoleh dari tahap pengumpulan data serta dari analisa kebutuhan sistem. Perancangan sistem ini dilakukan agar pada saat pengimplementasi sistemnya menjadi mudah dengan adanya perancangan sistem.

3.4.1 Arsitektur Pada Sistem Pendukung Keputusan

Pada perancangan arsitektur sistem pendukung keputusan terbagi dalam beberapa bagian yang mana bagian tersebut saling berkaitan satu dengan yang lainnya.



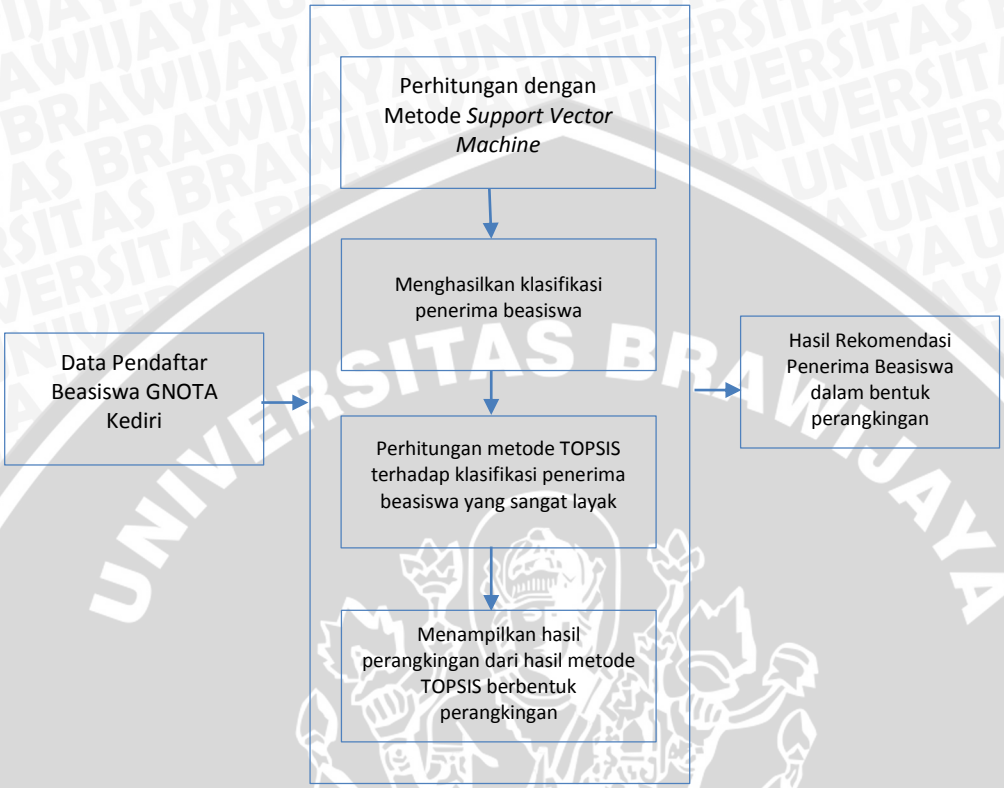
Gambar 3.2 Arsitektur Pada Sistem Pendukung Keputusan
Sumber: Sprague, R. H. and Watson H. J. (1993)

Gambar 3.2 menunjukan arsitektur dari sistem pendukung keputusan yang mewakili beberapa komponen pendukung pada sistem pendukung keputusan. Yang mana pada masing-masing subtansi berbasis pengetahuan menjelaskan proses pembentukan dari alternatif yang sesuai dengan kriteria yang telah dibuat. Subsistem pada manajemen model merupakan proses untuk metode SVM yang melakukan pengklasifikasian data pendaftar penerima beasiswa GNOTA kemudian akan dilakukan metode TOPSIS untuk melakukan rekomendasi melalui hasil akhir perangkingan penerima beasiswa, pada tahap menggunakan metode TOPSIS, hanya data yang termasuk pada klasifikasi sangat layak yang akan diproses pada metode TOPSIS, sedangkan *User Interface* merupakan perantara antara sistem dan *user* atau pengguna.

3.4.2 Diagram Blok Sistem

Pada diagram blok sistem ini akan membahas tentang aliran proses dari komponen-komponen pada sistem pendukung keputusan yang memuat fungsi secara metematis dengan menggunakan sebuah diagram berbentuk blok-blok. Diagram blok ini akan menjelaskan cara kerja sistem pendukung keputusan yang

dimulai dari proses awal atau *input* ke sistem hingga *output* dari proses sistem pendukung keputusan. Adapaun diagram blok yang akan digunakan dapat dilihat pada Gambar 3.3 :



Gambar 3.3 Diagram Blok Sistem Pendukung Keputusan Rekomendasi Penerima Beasiswa GNOTA

Gambar 3.3 menunjukan dari proses sistem pendukung keputusan, yang mana dari tahap metode SVM berupa pengklasifikasian calon penerima beasiswa GNOTA kediri, kemudian setelah mendapatkan hasil klasifikasi dari metode SVM, akan diambil data yang menempati kelas sangat layak dari penerima beasiswa, dengan demikian hasil klasifikasi akan diambil datanya untuk dilakukan perangkingan dengan menggunakan metode TOPSIS dan akan menghasilkan peringkat dari rekomendasi penerima beasiswa di GNOTA kediri. Adapaun tahapan proses sistem pendukung keputusan rekomendasi penerima beasiswa sebagai berikut :

1. *Input*

   Pada tahapan *input*, dilakukan penginputan data calon penerima beasiswa seperti biodata dan beberapa kriteria seperti status pekerjaan orang tua, penghasilan orang tua, jumalah anak, tagihan PBB, tagihan listrik, kepemilikan kenderaan bermotor,rumah, serta biaya pendaftara, untuk lebih jelaskan terlampir pada lampiran A.
2. Proses

   Pada tahapan proses ini sistem menggunakan dua metode yaitu SVM dan TOPSIS. Yang mana pada metode SVM digunakan untuk melakukan klasifikasi

penerima beasiswa, kemudian data yang sudah diklasifikasi akan dirangking dengan menggunakan metode TOPSIS.

3. *Output*
   Output atau keluaran sistem nantinya berupa rekomendasi perangkingan dari penerima beasiswa GNOTA Kediri.

3.5 Implementasi Sistem

Pada implementasi sistem yang ada pada penelitian ini dilakukan dengan acuan terhadap rancangan sistem, nantinya implementasi sistem dilakukan dengan menggunakan bahasa pemograman PHP, manajemen *database* menggunakan MySQL, serta perangkat pendukung yang dibutuhkan lainya. Pada implementasi sistem ini nanti yang akan di *input* kedalam sistem adalah seperti terlampir pada lampiran A sehingga adapun untuk tahapan-tahapan implementasi pada sistem sebagai berikut :

1. Pembuatan *Interface* (antarmuka).
2. Perhitungan metode SVM untuk pengklasifikasian penerima beasiswa.
3. Perhitungan metode TOPSIS terhadap hasil dari pengklasifikasian yang menggunakan metode SVM.
4. Keluaran berupa rekomendasi dan perangkingan dari calon penerima beasiswa di lembaga GNOTA kediri.

3.6 Pengujian dan Analisis

Pengujian yang akan dilakukan pada penelitian ini ada dua pengujian yang terdiri dari pengujian pertama untuk kecepatan sistem untuk memproses seleksi dan rekomendasi beasiswa, yang mana nantinya sistem dapat bekerja secara optimal sehingga sistem dapat digunakan dengan sebaiknya untuk mendukung seleksi dan rekomendasi beaiswa GNOTA kediri. Pengujian kedua dilakukan untuk mengetahui tingkat akurasi sistem dengan beberapa paremeter pengujian.

Analisis sistem yang akan dilakukan pada penelitian ini adalah menganalisis metode SVM dan TOPSIS yang diimplementasi pada seleksi beasiswa GNOTA Kabupaten Kediri, apakah metode tersebut dapat berjalan baik pada sistem yang dibangun nantinya.

3.7 Pengambilan kesimpulan dan saran

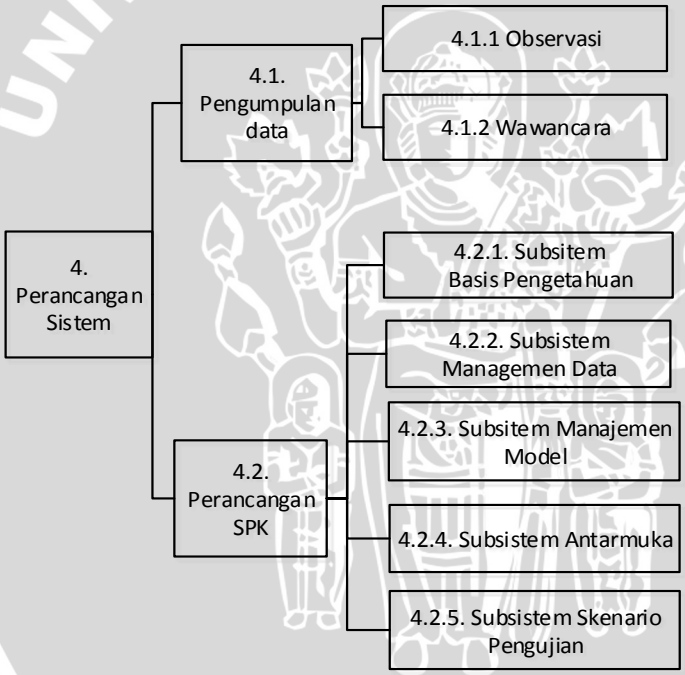
Pengambilan kesimpulan nantinya dilakukan setelah tahapan dari perancangan, implementasi dan pengujian sistem telah selesai semua yang berdasarkan pada kesesuaian antara teori dan implementasi atau praktik. Kesimpulan diambil berdasarkan dari hasil pengujian sistem dan analisis dari penggunakan metode SVM dan TOPSIS yang bertujuan untuk menjawab dari rumusan masalah yang telah ditetapkan. Tahap akhir dari penulisan adalah saran untuk memperbaiki dan menyempurnakan kesalahan yang terjadi pada penulisan serta memberikan pertimbangan atas pengembangan penelitian selanjutnya.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

BAB 4 PERANCANGAN SISTEM

Bab ini membahas cara pengumpulan data primer dan perancangan dari perhitungan manualisasi Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri dengan Implementasi Metode SVM dan TOPSIS. Adapun tahapan-tahapan perancangan sistem yang diperlukan yaitu, perancangan sistem pendukung keputusan dan kebutuhan perangkat lunak. Tahapan untuk melakukan perancangan sistem pendukung keputusan tediri dari perancangan subsistem basis pengetahuan, subsistem manajemen data, subsistem manajemen model dan subsistem *interface* atau antarmuka dan subsistem skenario pengujian.

Tahapan perancangan kebutuhan perangkat lunak terdiri dari identifikasi aktor dan daftar kebutuhan sistem. Untuk lebih jelasnya dapat dilihat dalam perancangan pada gambar 4.1



Gambar 4.1 Diagram Perancangan

Gambar 4.1 menunjukan Diagram Perancangan dari pengumpulan data primer yang dilakukan dengan observasi dan wawancara serta perancangan sistem Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri dengan Implementasi Metode SVM dan TOPSIS. Dari diagram tersebut perancangan terdiri dari perancangan SPK, yang memiliki 5 Subsistem yaitu Subsistem Basis Pengetahuan, Subsistem Managemen Data, Subsistem Manajemen Model, Subsistem Antarmuka dan yang terakhir Subsistem Skenario Pengujian.

4.1 Pengumpulan Data

Pengumpulan data merupakan bagaimana cara-cara pengumpulan data dilakukan pada lembaga GNOTA Kediri, yang mana data digunakan merupakan data primer, Data primer didapatkan dengan cara melakukan observasi dan wawancara langsung terhadap lembaga GNOTA Kediri kepada salah satu karyawan GNOTA yaitu bapak Aji, yang mana data primer didapatkan dengan cara data mentah yang diolah sendiri oleh peneliti dari berupa formulir pendaftaran ke format excel yang mana ditentukan oleh pihak lembaga GNOTA kediri itu sendiri, adapun cara pengumpulan data nya terdiri dari observasi dan wawancara.

4.1.1 Observasi

Observasi dilakukan guna mendapatkan data yang akan digunakan dalam penelitian ini nantinya, yang mana observasi dilakukan di kabupaten kediri yang dimulai dari surat pengajuan dari kampus Fakultas Ilmu Komputer Universitas Brawijaya yang mana surat diajukan ke Pemerintah Kabupaten Kediri Badan Kesatuan Bangsa dan Politik, dan kemudian diajukan ke Dinas Pendidikan Kabupaten Kediri, dan selanjutnya ke lembaga GNOTA Kediri sebagai bagian dari penelitian ini, dan mendapatkan beberapa data sampel serta permasalahan yang sering terjadi dalam proses seleksi beasiswa.

Sehingga pada observasi ini dilakukan pada beberapa data dari seleksi beasiswa GNOTA pada tahun 2015 dan ada pun beberapa permasalahan sudah dijelaskan pada bab sebelumnya yaitu pada Bab 1 sub bab latar belakang.

4.1.2 Wawancara

Wawancara dilakukan langsung terhadap ketua maupun karyawan dari lembaga GNOTA Kabupaten Kediri sendiri, yang pada tahap observasi dan wawancara kepada bapak Aji, adapun hal-hal yang ditanyakan untuk mendapatkan dan mengetahui proses penerimaan beasiswa GNOTA Kediri yaitu terdiri dari (Wawancara, 2015):

1. Permasalahan apa saja yang terjadi pada tahap seleksi beasiswa GNOTA Kediri pada tahun 2015 ?
2. Berapa banyak data pendaftar beasiswa GNOTA Kediri pada tahun 2015 ?
3. Bagaimana proses seleksi yang terjadi untuk seleksi beasiswa GNOTA Kediri pada tahun 2015?

Sehingga dari pihak GNOTA Kediri memberikan penjelasan menyangkut pertanyaan wawancara diatas dengan jawaban yang akan mendukung peneliti dalam mendapatkan dan mengetahui data maupun informasi mengenai proses seleksi, sebagai berikut (Wawancara, 2015):

1. Masih menggunakan perhitungan manual seperti pada proses pengklasifikasian yang mana ada tiga klasifikasi dengan banyaknya data maka sulit untuk menentukan salah satu peserta termasuk dalam kalsifikasi apa dan dalam masalah perangkingan masih menggunakan manual, apabila dilakukan secara manual maka panitia mengurutkan satu persatu tidak secara otomatis.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Pengisian data secara manual dengan menggunakan form pendaftaran kemudian dimasukkan ke program *excel* sehingga membutuhkan waktu yang terbilang lama yaitu tiga bulan dari proses pendaftaran hingga hasil seleksi, data yang dimasukan juga bisa mengalami kesalahan seperti tidak samanya data di *excel* dengan data pada saat pengisian *form* pendaftaran dikarenakan faktor *human error*, apabila data dari tahun ketahun bertambah banyak dari pendaftar beasiswa maka proses penyimpanan data akan mengalami penumpukan berupa berkas yang tidak dapat tertangani oleh sumber daya yang terbatas dan tidak ada jejak *record*-nya, kurangnya sumber daya manusia apabila melakukan survei satu-persatu dikarenakan jumlah pegawai lembaga GNOTA Kediri hanya terdiri dari satu orang kepala lembaga dan dua orang karyawan.

2. Banyaknya data pendaftar beasiswa GNOTA kediri pada tahun 2015 sebanyak 111 pendaftar, yang mana setiap pendaftar diberikan formulir pendaftaran seperti pada Lampiran A, dan kemudian dilengkapi dengan berkas-berkas pendukung, dan data itu kemudian diolah oleh peneliti yang diinputkan di *excel* dan diproses lagi melalui proses seleksi.

3. Proses seleksi sendiri ada beberapa tahap, yang mana pada tahap pertama peserta mengisikan biodata sesuai formulir pada Lampiran A, sebelum penyeberan formulir disetiap sekolah pihak GNOTA Kediri melakukan sosialisasi terhadap sekolah-sekolah dan bisa juga mendapatkan dari pihak sekolah melalui jalur prestasi siswa,kemudian dilakukan pengisian formulir dari formulir tersebut akan dilihat domisilinya, apabila peserta tersebut berdomisili di Kabupaten Kediri maka akan dilakukan proses seleksi berikutnya, dan apabila peserta berdomisili dikota Kediri tidak akan melanjutkan ketahap berikutnya, kemudian dilihat dari beberapa aspek kriteria yaitu 13 kriteria berupa Penghasilan, Jumlah Tanggungan, Pekerjaan, Daya Listrik, Biaya Listrik, Jumlah Rumah, Status Rumah, Status Dinding, Status Lantai, PBB, Jumlah Motor, Jumlah Mobil, dan Program Beasiswa. Apabila telah memenuhi kriteria dengan beberapa nilai parameter dari sub kriteria, penerima beasiswa akan diseleksi berdasarkan 3 klasifikasi, yaitu sangat layak, layak dan tidak layak. Penerimaan terhadapa kuota penerima beasiswa setiap tahun nya menignkat sehingga jika penerima sangat layak masih kurang dari kuota makan akan ditambahkan dari klasifikasi penerima layak, dan tidak layak tidak akan menerima beasiswa GNOTA Kediri.

Hasil wawancara juga mendapatkan beberapa data yang didapatkan sejumlah 111 data sampel atau data, dan data dari Lembaga GNOTA Kediri pada tahun 2015 yang mana data tersebut menjadikan beberapa kriteria seperti pada formulir pedaftaran yang ditunjukan pada Lampiran A.

Dijadikan data seminimal mungkin menjadi 13 kriteria untuk dilakuakn proses perhitungan dengan metode SVM dan TOPSIS, berikut contoh data yang didapatkan dari hasil wawancara langsung kepada Bapak Aji, pada Tabel 4.1.

Tabel 4.1 Contoh Data Dari GNOTA Kediri

NO	NO PESERTA	NAMA	TOTAL PENGHASILAN	JUMLAH ANAK	PEKERJAAN	....	PROGRAM BEASISWA	KLASIFIKASI
1	263	NERY DWI EVITHELISARI	1.300.000	1	Kuli Bangunan	.....	LAINNYA	Sangat Layak
4	266	LAILIL AFIF AGUSTINA	1.200.000	1	Buruh Tani	.....	BIDIK MISI	Sangat Layak
5	267	YUSUF	1.000.000	1	Makelar	.....	LAINNYA	Sangat Layak
70	185	NILA PUTRI ANGGRAINI	1.000.000	2	Konstruksi	.....	LAINNYA	Layak
71	187	MUHAMMAD DAMANURI	500.000	2	tidak bekerja	.....	BIDIK MISI	Layak
67	180	M IQBAL BAIHAQI	500.000	2	Konstruksi	.....	LAINNYA	Layak
107	400	M MUSYAFA YAZIO	4.200.000	2	Petani	.....	BIDIK MISI	Tidak Layak
108	89	MIRA SAGOLA	1.500.000	2	Tukang Las	.....	BIDIK MISI	Tidak Layak

Sumber : [Wawancara]

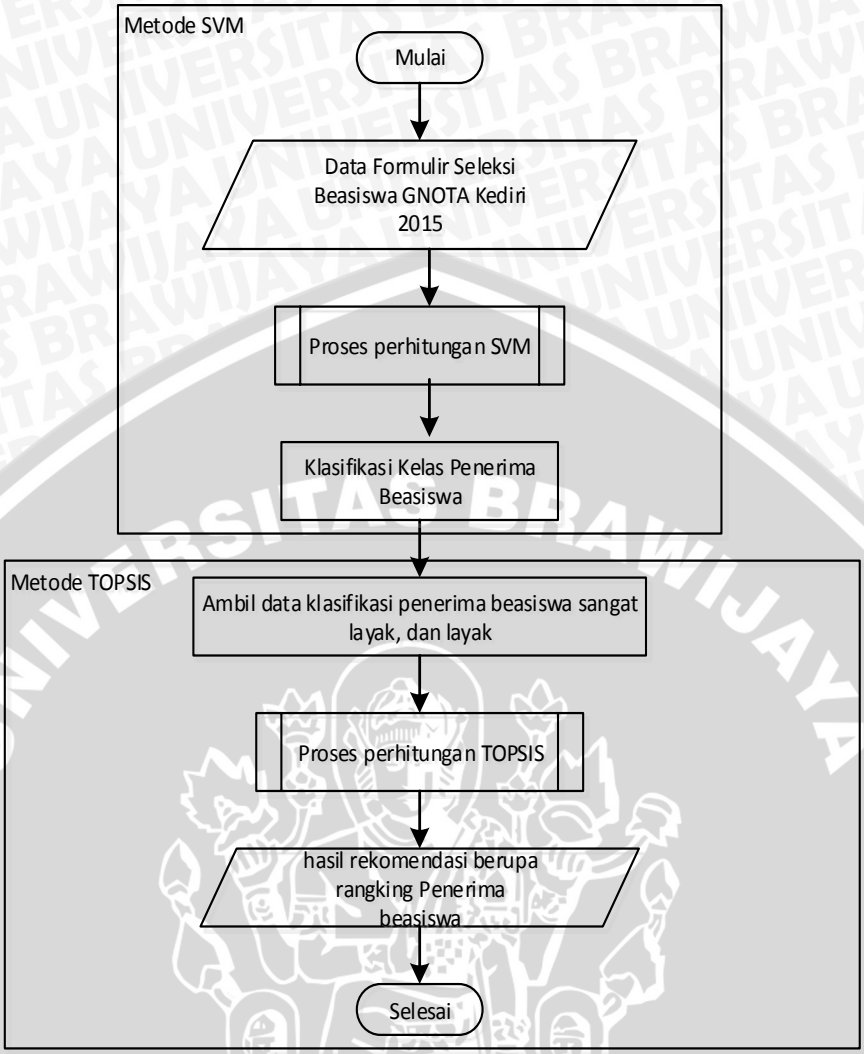
Tabel 4.1 menunjukan data yang didapatkan dari hasil wawancara kepada pihak Lembaga GNOTA Kediri, yang mana keseluruhan data sampel sebanyak 111, adapun data lengkap dapat dilihat pada Lampiran B.

Serta dari hasil wawancara juga didapatkan kriteria perhitungan seleksi beasiswa sebanyak 13 kriteria yaitu terdiri dari Penghasilan, Jumlah Tanggungan, Pekerjaan, Daya Listrik, Biaya Listrik, Jumlah Rumah, Status Rumah, Status Dinding, Status Lantai, PBB, Jumlah Motor, Jumalah Mobil, dan Program Beasiswa, yang mana 13 kriteria tersebut merupakan kriteria utama dalam penentuan beasiswa di lembaga GNOTA Kediri

4.2 Perancangan Sistem Pendukung Keputusan

Perancangan SPK berupa alur dari sistem Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri dengan Implementasi Metode SVM dan TOPSIS yang akan diimplementsaikan nantinya kedalam sistem yang mana perancangan ini dapat membantu dalam sistem pendukung keputusan yang akan dibuat nantinya, perancangan sistem pendukung keputusan dapat di tunjukan pada Gambar 4.2, sebagai berikut :

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA



Gambar 4.2 Alur Kerja SPK Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri

Gambar 4.2 menunjukan proses dari seleksi dan rekomendasi penerima beasiswa di lembaga GNOTA Kediri dengan menggunakan metode SVM dan TOPSIS, yang mana proses dimulai dari proses *input* data sesuai dengan formulir 2015. Setelah data dari formulir di *input* maka akan dilakukan perhitungan dengan metode SVM dengan *output* berupa pengklasifikasian penerima beasiswa, kemudian setelah dilakukan proses klasifikasi metode SVM dan menghasilkan kelas sangat layak, kemudian dilakukan perhitungan dengan metode TOPSIS untuk mendapatkan perangkingan dari kelas sangat layak untuk penerima beasiswa GNOTA Kediri.

4.2.1 Subsistem Basis Pengetahuan

Subsistem basis pengetahuan yaitu subsistem yang mendukung keputusan subsistem lainnya. Pada sistem seleksi dan rekomendasi penerima beasiswa di lembaga GNOTA kediri dengan menggunakan metode SVM dan TOPSIS adalah alur

proses seleksi dan kriteria yang digunakan sebagai pengambil keputusan beserta pembobotannya.

Subsistem basis pengetahuan ini mendukung penyimpanan data yang berada dalam subsistem manajemen data yaitu basis data sistem pendukung keputusan di database manajemen MySQL dan manajemen model untuk memproses perhitungan data dari metode SVM dan TOPSIS. Jumlah data sampel yang digunakan nantinya sebanyak 14 data perserta seleksi beasiswa GNOTA Kediri (9 data *training* dan 5 data *testing*). Semua data yang akan diolah tersebut ditunjukkan pada Tabel 4.2.

Tabel 4.2 Data Perserta Beasiswa GNOTA Kediri.

Ai	KRITERIA													KELAS
	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	
A23	5	2	5	3	4	2	1	1	2	4	5	3	2	A
A24	4	1	5	4	4	2	1	1	2	2	4	3	2	A
A25	3	1	4	4	4	2	1	2	2	4	4	3	2	A
A73	4	1	2	4	4	2	1	1	1	3	3	3	1	B
A74	2	1	3	4	3	2	1	1	1	4	4	3	2	B
A75	3	1	2	4	3	2	1	1	1	3	4	3	1	B
A102	2	1	1	3	3	2	1	1	1	4	4	3	1	C
A103	1	1	3	2	2	2	1	1	2	4	4	3	1	C
A104	2	1	4	3	4	2	1	1	2	3	3	3	1	C
A14	4	1	4	4	4	2	1	1	2	2	3	3	2	A
A32	5	1	4	4	2	2	1	1	1	2	4	3	2	A
A64	3	1	4	3	2	2	1	1	1	4	3	3	1	B
A65	4	1	3	4	4	2	1	1	1	4	3	3	1	B
A99	4	1	4	3	3	2	1	1	1	4	3	3	1	C

Tabel 4.2 menunjukan data sample yang akan digunakan dalam perhitungan didalam sistem nantinya, yang mana terdiri dari 13 kriteria yaitu 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,dan C13 = Program Beasiswa. Serta terdiri 3 kelas yaitu A= Sangat Layak, B= Layak, dan C= Tidak Layak.

4.2.2 Subsistem Managemen Data

Sistem seleksi dan rekomendasi penerima beasiswa di lembaga GNOTA kediri dengan menggunakan metode SVM dan TOPSIS ini nantinya menggunakan DBMS (*Data Base Management System*) berupa MySQL server. *Physical* diagram

nantinya bertujan untuk mempermudah dalam pengembangan dan pengolahan database dalam membuat *query* DDL (*Data Definition Language*).

4.2.3 Subsistem Managemen Model

Subsistem managemen model ini digunakan untuk proses perhitungan data dengan menggunakan kedua metode yaitu SVM dan TOPSIS. Adapun diagram alir ditunjukan pada Gambar 4.2, merupakan penggambaran proses dari penggunaan metode SVM dan TOPSIS berdasarkan secara garis besar. Dari beberapa kriteria yang ada pada Tabel 4.1 yang nantinya digunakan sebagai parameter pengambilan keputusan didapatkan dari hasil wawancara terhadap Kepala maupun karyawan dari lembaga GNOTA Kediri.

Sistemnya sendiri nanti yang akan melakukan pengisian pembobotan dilakukan oleh *administrator* sehingga nantinya akan mendapatkan pembobotan nilai sesuai dari hasil wawancara yang didapat. Dan akan dilakukan proses perhitungan dengan kedua metode tersebut, sehingga menghasilkan klasifikasi kelas penerima dan hasil akhir berupa perangkingan dari kelas penerima sangat layak dari penerima beasiswa di lembaga GNOTA Kediri.

4.2.3.1 Perhitungan Metode SVM

Pada perhitungan metode SVM ini, akan menjelaskan langkah-langkah perhitungan dan penentuan klasifikasi kelas dengan metode SVM. Adapun masukan dari parameter yang didapatkan 13 kriteria seperti Tabel 4.1, yaitu C1, C2, C3 ,C4, C5, C6, C7, C8, C9, C10, C11, C12, dan C13.

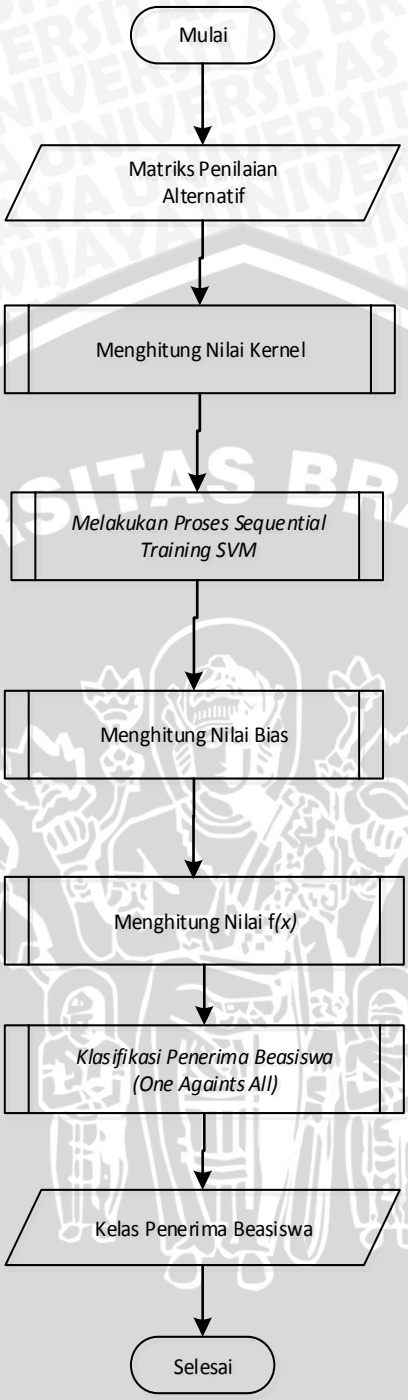
Adapun tahapan dalam manajemen model perhitungan Sistem seleksi dan rekomendasi penerima beasiswa di lembaga GNOTA kediri sebagai berikut:

1. Menentukan matriks penilaian alternatif atau normalisasi.
2. Menghitung matriks kernel.
3. Menghitung proses sequential *training* SVM, dengan langkah-langkah sebagai berikut :
   a. Menghitung nilai *Ei*.
   b. Menghitung nilai $\delta\alpha i$.
   c. Menentukan αi baru.
4. Menghitung nilai bias.
5. Menghitung nilai dari fungsi *f(x).*
6. Klasifikasi Penerima Beasiswa (*one againts all*)

Tahapan proses perhitungan seleksi dan rekomendasi dari langkah-langkah tersebut sehingga dapat dibuat diagram alir yang menjelaskan tahapan proses perhitungan metode SVM.

Adapun tahapan dalam menentukan matriks penilaian alternatif atau normalisasi, menghitung matriks kernel, menghitung matriks Hessian, menghitung iterasi dengan beberapa langkahnya, menentukan *support vector*, menghitung nilai bobot data *training*, menghitung bias dan yang terakhir sampai dengan menghitung nilai *f(x),* akan ditunjukan pada Gambar 4.3.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA



Gambar 4.3 Diagram Alir Perhitungan Menggunakan Metode SVM

Gambar 4.3 menunjukan tahapan proses perhitungan dari metode SVM, dari proses normalisasi sampai penentuan *kelas F(x)* penerima beasiswa dari lembaga GNOTA Kediri.

Langkah 1 : Menentukan Matriks Penilaian Alternatif Level 1

Tabel 4.1 merupakan data yang akan dilakukan *Min-Max Normalization*, yang terdapat 12 data sampel, yang terdiri dari 9 data *traning* dan 3 data *testing*.

Adapun hasil dari matriks penilaian alternatif atau normalisasi pada tabel 4.2. dengan perhitungan menggunakan rumus:

$$v^i(i) = \left(\frac{v(i) - \min(v(i))}{\max(v(i)) - \min(v(i))}\right) x\ (BA - BB) + BB$$

Keterangan :

$v(i)$ = Data yang akan dinormalisasi.

$\min(v(i))$ = Nilai minimum dari keseluruhan data-*i*.

$\max(v(i))$ = Nilai maksimum dari keseluruhan data-*i*.

BA = Batas atas *interval*.

BB = Batas bawah *interval*.

Normalisasi dilakukan untuk mendapatkan nilai antar fitur agar memiliki selisih yang berada pada *interval* yang sama. Untuk contoh perhitungan normalisasi pada data *training* ke A23 dari tabel 4.1 dengan nilai $v(i)$ = 5, $\min(v(i))$ = 1, $\max(v(i))$ = 5, BA = 1, dan BB = 0.1.

$$v^i(i) = \left(\frac{5-1}{5-1}\right) x\ (1 - 0.1) + 0.1$$

$$v^i(i) = 1{,}000$$

Adapun hasil dari semua data sampel yang telah dinormalisasi yang ditunjukan pada Tabel 4.3 untuk data *training* dan Tabel 4.4 untuk data *testing*.

Tabel 4.3 Penilaian Alternatif atau Normalisasi Untuk Data Training Level 1

Ai	C1	C2	C3	C4	C5	....	C11	C12	C13	KELAS
A23	1,000	1,000	1,000	0,550	1,000	....	1,000	1,000	1,000	A
A24	0,775	0,100	1,000	1,000	1,000	....	0,700	1,000	1,000	A
A25	0,550	0,100	0,775	1,000	1,000	....	0,700	1,000	1,000	A
A73	0,775	0,100	0,325	1,000	1,000	....	0,400	1,000	0,100	B
A74	0,325	0,100	0,550	1,000	0,700	....	0,700	1,000	1,000	B
A75	0,550	0,100	0,325	1,000	0,700	....	0,700	1,000	0,100	B
A102	0,325	0,100	0,100	0,550	0,700	....	0,700	1,000	0,100	C
A103	0,100	0,100	0,550	0,100	0,400	....	0,700	1,000	0,100	C
A104	0,325	0,100	0,775	0,550	1,000	....	0,400	1,000	0,100	C

Dari beberapa alternatif dari data sample pada tabel 4.3 yaitu data calon penerima beasiswa lembaga GNOTA Kediri. Data pada tabel 4.3 nantinya sebagai data *training* yang digunakan untuk mencari garis *Hyperplane* untuk memisahkan kelas A (+1), B (-1) dan C (-1).

Tabel 4.4 Penilaian Alternatif atau Normalisasi Untuk data Testing Level 1

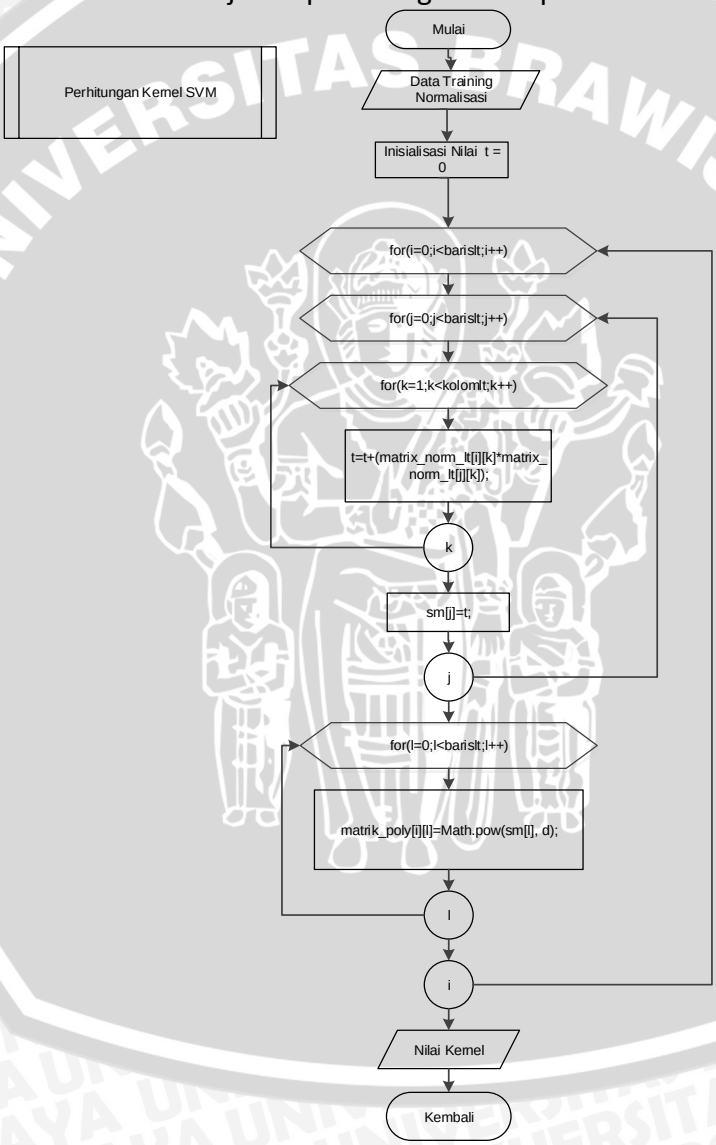
Ai	C1	C2	C3	C4	C5	....	C11	C12	C13	KELAS
A14	0,775	0,100	0,775	1,000	1,000	....	0,400	1,000	1,000	A
A32	1,000	0,100	0,775	1,000	0,400	....	0,700	1,000	1,000	A
A64	0,550	0,100	0,775	0,550	0,400	....	0,400	1,000	0,100	B
A65	0,775	0,100	0,550	1,000	1,000	....	0,400	1,000	0,100	B
A99	0,775	0,100	0,775	0,550	0,700	....	0,400	1,000	0,100	C

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 4.4 merupakan tabel yang telah dilakukan normalisasi pada data *testing*, yang mana data *testing* terdiri dari 2 data untuk kelas sangat layak dan layak, dan 1 data untuk kelas tidak layak pada setiap masing-masing kelas pada data *training*.

Langkah 2: Menghitung Matriks Kernel Level 1

Pada proses perhitungan matriks kernel ini akan dilakukan setelah semua data sampel pada data *training* telah dimasukkan ke dalam sistem seleksi dan rekomendasi. Kemudian data training akan dilakukan perhitungan untuk mencari matriks kernel dengan menggunakan matriks *kernel Polynomial Degree d*, dengan persamaan 2.14. Adapun tahapan-tahapan proses perhitungan awal sampai akhir dari matriks kernel akan ditunjukan pada diagram alir pada Gambar 4.4.



Gambar 4.4 Diagram Alir Perhitungan Matriks Kernel Polynomial Degree *d*

Gambar 4.4 menunjukkan tahapan proses perhitungan matriks kernel, sehingga didapatkan contoh langkah perhitungan matriks kernel *Polynomial Degree d* , dengan beberapa nilai *d* = 2, dari rumus persamaan 2.14.

$K(x, x_i) = (x.x_i)^d$

Nilai dari variabel *x* dan *y* yaitu merupakan dari 2 data *training* yang akan dicari nilai *dot Product* nya dengan menggunakan kernel *Polynomial Degree d.* Untuk baris pertama dan kolom pertama dari Tabel 4.2, kernel (0,0) :

x = A23
y = A23

$$[x\ .\ y]^d = [(X_{C1} \times X_{C1}) + (X_{C2} \times X_{C2}) + (X_{C3} \times X_{C3}) + (X_{C4} \times X_{C4}) + (A23_{C5} \times X_{C5}) + (X_{C6} \times X_{C6}) + (X_{C7} \times X_{C7}) + (X_{C8} \times X_{C8}) + (X_{C9} \times X_{C9}) + (X_{C10} \times X_{C10}) + (X_{C11} \times X_{C11}) + (X_{C12} \times X_{C12}) + (X_{C13} \times X_{C14})]2$$

$$= [(1{,}000 \text{ x } 1{,}000) + (1{,}000 \text{ x } 1{,}000) + (1{,}000 \text{ x } 1{,}000) + (0{,}550 \text{ x } 0{,}550) + (1{,}000 \text{ x } 1{,}000) + (0{,}100 \text{ x } 0{,}100) + (0{,}100 \text{ x } 0{,}100) + (0{,}100 \text{ x } 0{,}100) + (0{,}550 \text{ x } 0{,}550) + (1{,}000 \text{ x } 1{,}000) + (1{,}000 \text{ x } 1{,}000) + (1{,}000 \text{ x } 1{,}000) + (1{,}000 \text{ x } 1{,}000)]^2$$

$$= 74{,}56322500$$

Setelah melakukan perhitungan kernel pada A23 utuk baris pertama dan kolom pertama selesai, maka lakukan seterusnya sehingga akan diperoleh nilai matriks pada kernel *Polynomial Degree d* pada Tabel 4.5.

Tabel 4.5 Matriks Kernel Polynomial Degree d Level 1

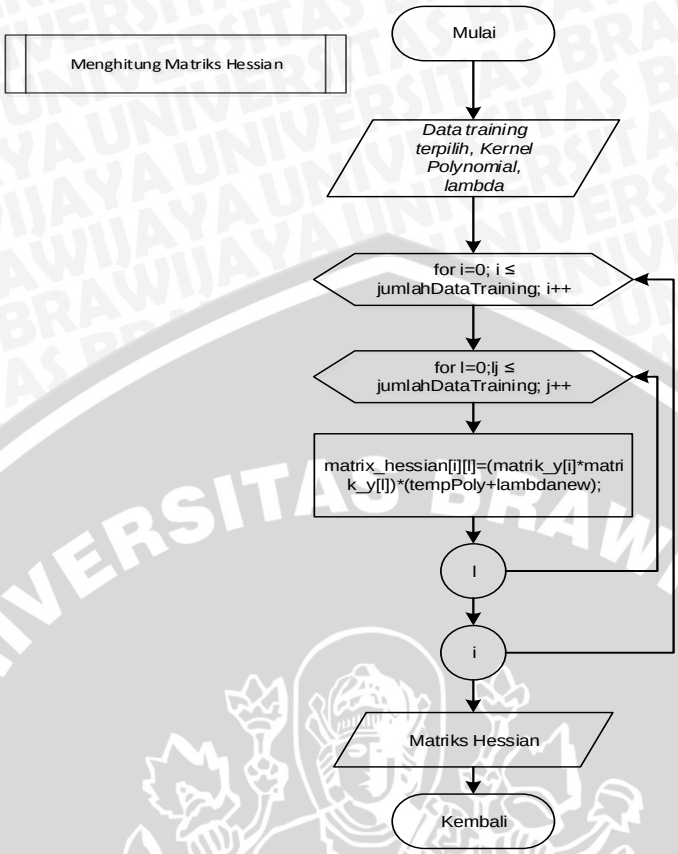
K(x,y)	A23	A24	....	A73	A102	A104
A23	74,56322500	47,02530625	....	25,35122500	19,47015625	25,35122500
A24	47,02530625	43,46929727	....	21,90825039	13,59304727	20,97067539
A25	49,73775625	41,97420156	....	23,99162852	17,94581406	23,01000977
A73	25,35122500	21,90825039	....	19,50326406	12,77615664	14,50657656
A74	36,12010000	30,10579727	....	18,15292539	14,75040039	15,65685977
A75	23,13610000	19,50326406	....	16,49882852	12,25875156	12,64246914
A102	19,47015625	13,59304727	....	12,77615664	11,95862852	10,69698789
A103	18,81390625	11,97160000	....	8,07270156	9,10530625	10,53813906
A104	25,35122500	20,97067539	....	14,50657656	10,69698789	16,09012656

Tabel 4.5 menunjukan hasil dari perkalian matriks kernel dari persamaan 2.14 yang didapatkan dari normalisasi data *training.*

Langkah 3 : Menghitung Matriks Hessian Level 1

Perhitungan pada matriks Hessian dilakukan dengan cara apabila matriks kernel kernel *Polynomial Degree d* sudah selesai, perhitungan untuk mencari nilai pada matriks Hessian ini akan menggunakan kernel kernel *Polynomial Degree 2,* adapun persamaan untuk menghitung matriks Hessian pada persamaan 2.18. tahapan perhitungan matriks Hessian ditunjukkan dengan diagram alir pada Gambar 4.5.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA



Gambar 4.5 Diagram Alir Perhitungan Matriks Hessian

Perhitungan matriks Hessian ini ada beberapa nilai yang dapat diasumsikan yaitu λ = 0.5 (variabel skalar), variabel y_i adalah kelas data ke-*i*, variabel y_j adalah kelas data ke-j, dan $K(x_i, y_j)$ merupakan nilai kernel ke-*i,j*. Dengan rumus dari persamaan 2.18.

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

Dari persamaan tersebut, maka untuk perhitungan matriks Hessian pada A23 sebagai berikut :

Untuk kolom ke-1 baris ke-1:

$$y_i y_j (K(x_i, y_j) + \lambda^2) = 1 \text{ x } 1 \text{ x } (74,56322500 + 0,5^2) = 74,81322500$$

Untuk kolom ke-1 baris ke-2:

$$y_i y_j (K(x_i, y_j) + \lambda^2) = 1 \text{ x } 1 \text{ x } (47,02530625 + 0,5^2) = 47,27530625$$

Untuk kolom ke-1 baris ke-3:

$$y_i y_j (K(x_i, y_j) + \lambda^2) = 1 \text{ x } 1 \text{ x } (49,73775625 + 0,5^2) = 49,98775625$$

Dari beberapa contoh perhitungan matriks Hessian diatas maka didapatkan nilai dari matriks Hessian pada Tabel 4.6.

Tabel 4.6 Matriks Hessian Level 1

Dij	A23	A24	....	A73	A102	A104
A23	74,81322500	47,27530625	....	-25,60122500	-19,72015625	-25,60122500
A24	47,27530625	43,71929727	....	-22,15825039	-13,84304727	-21,22067539
A25	49,98775625	42,22420156	....	-24,24162852	-18,19581406	-23,26000977
A73	-25,60122500	-22,15825039	....	19,75326406	13,02615664	14,75657656

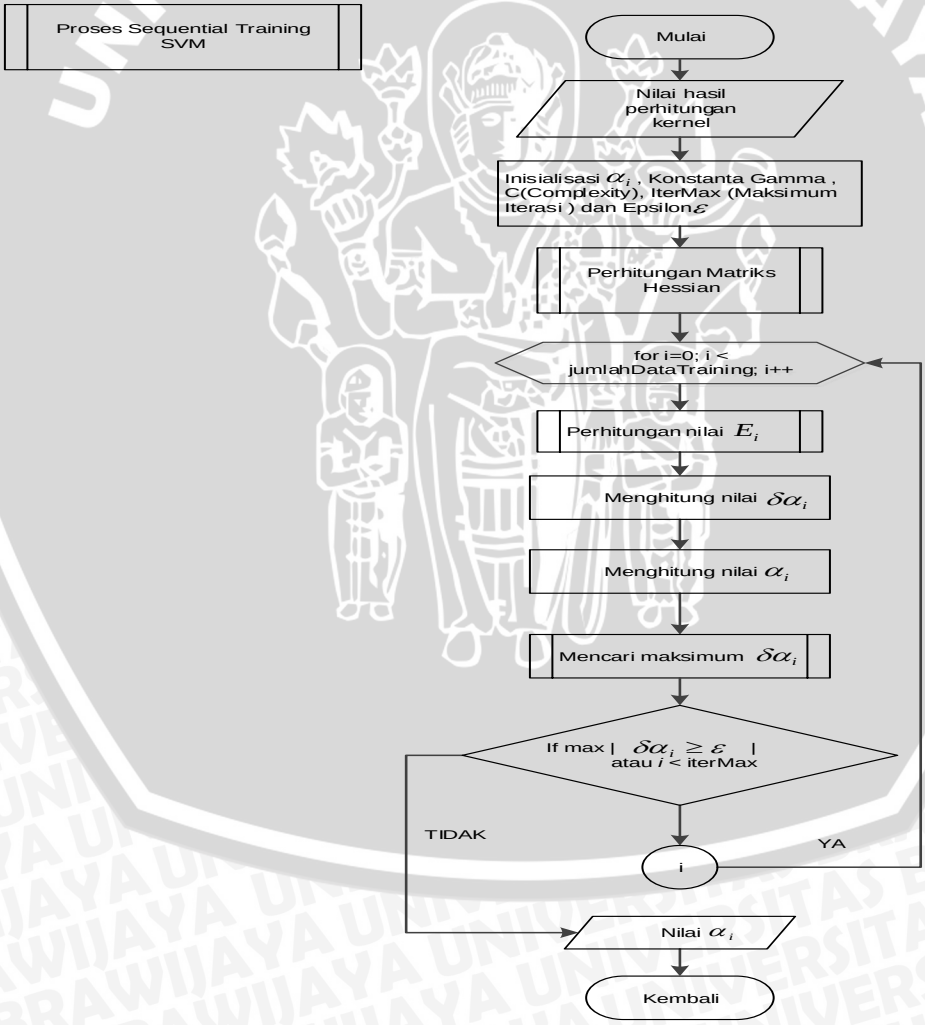
A74	-36,37010000	-30,35579727	....	18,40292539	15,00040039	15,90685977
A75	-23,38610000	-19,75326406	....	16,74882852	12,50875156	12,89246914
A102	-19,72015625	-13,84304727	....	13,02615664	12,20862852	10,94698789
A103	-19,06390625	-12,22160000	....	8,32270156	9,35530625	10,78813906
A104	-25,60122500	-21,22067539	....	14,75657656	10,94698789	16,34012656

Tabel 4.6 merupakan hasil dari seluruh peritungan matriks Hessian, yang mana perhitungannya didapatkan dari perhitungan kernel selesai.

Langkah 4: Melakukan *Sequential Training*

Sequential training akan dilakukan 3 tahap perhitungan yang akan diulang sebanyak nilai $max(|\delta\alpha i|) < \varepsilon$ atau bisa juga diulang sebanyak jumlah *n* kali (*n* merupakan jumlah iterasi maksimum atau iterMAX). Proses pertama untuk melakukan proses iterasi yaitu melakukan perhitungan untuk mencari nilai *Ei* setiap data *training*. Kemudian proses selanjutnya atau proses kedua yaitu melakukan perhitungan terhadap $\delta\alpha i$, proses terakhir atau proses ketiga adalah melakukan pencarian terhadap nilai αi.

Tahapan melakukan *Sequential training* dapat dilihat pada diagram alir yang ditunjukkan pada Gambar 4.6.

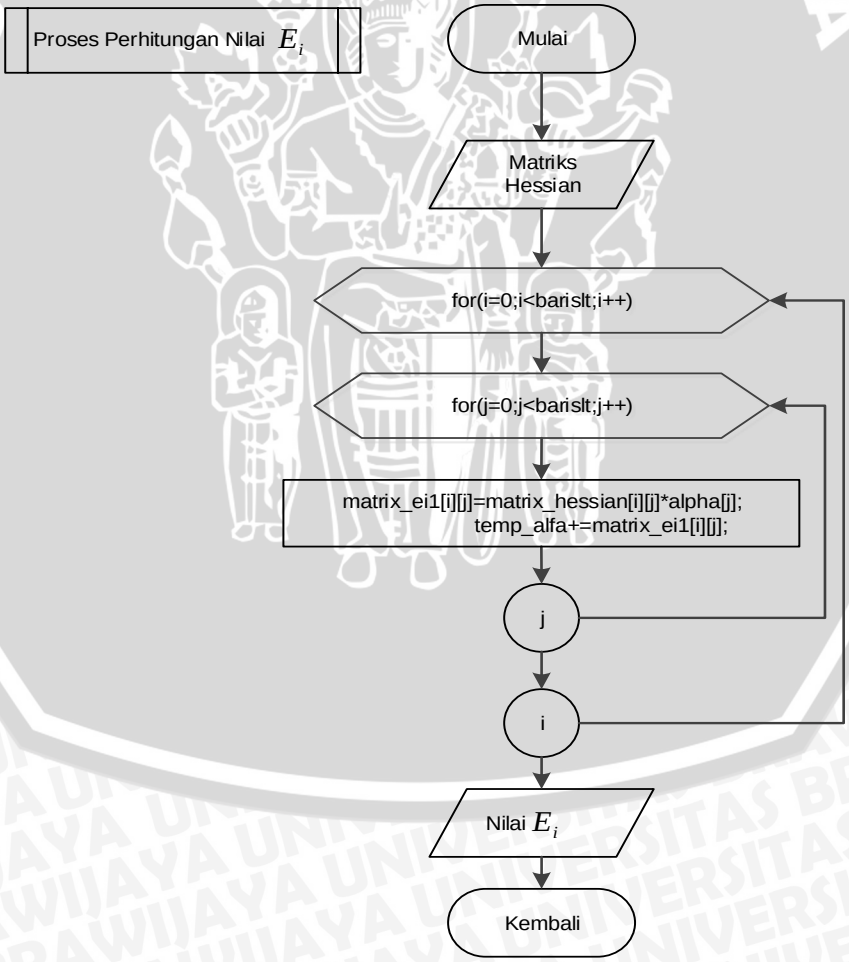


Gambar 4.6 Diagram Alir Proses *Sequential Training SVM*

Itermax dan epsilon yang memiliki nilai variabel pada masing-masingnya akan mempengaruhi iterasi yang nantinya akan dilakukan, jika nilai maksimum dari delta alpha lebih besar dari nilai variabel epsilon maka iterasi akan berhenti. Diasumsikan nilai iteraksi maksimum (*iterMAX*) = 2, epsilon (ε) = 0.001, dan nilai α = 0 (asumsi nilai alpha awal). Sebelum melakukan iterasi, nilai alpha baru akan dihitung dengan beberapa tahap seperti pada proses iterasi selama 1 kali, sehingga dapat menghasilkan nilai Ei = 0, $\delta\alpha i$ = 0,000133666, dan alpha baru αi = 0,000133666. Dari hasil nilai delta aplha baru nilai $max(|\delta\alpha i|) > \varepsilon$ maka dilakukan proses iterasi, Setelah mendapatkan nilai alpha baru maka nilai tersebut digunakan untuk perhitungan alpha pada proses iterasi pertama, adapun langkah-langkah iterasi pertama iterasi berikutnya sebagai berikut :

a. Menghitung nilai *Ei*

Menghitung nilai *Ei* dengan cara melakukan penjumlahan dari hasil perkalian matriks Hessian dengan alpha ke-*j*, yang mana proses tersebut merupakan proses iterasi awal dari metode *Sequential Training* SVM. Langkah perhitungan nilai *Ei* dapat dilihat pada diagram alir pada Gambar 4.7.



Gambar 4.7 Diagram Alir Menghitung Nilai Ei

Gambar 4.7 merupakan tahapan menghitung nilai *Ei* dengan diagram alir, dengan demikian, didapatkan tahapan dari langkah-langkah perhitungan nilai *Ei* dengan nilai variabel α_j adalah alpha ke-*j* dengan hasil perhitungan sebelumnya yaitu 0,000133666 dan D_{ij} adalah nilai matriks Hessian baris ke-*i* dan kolom ke-*j*, dengan demikian didapatkan perhitungan nilai Ei sebagai berikut :

$$Ei = \sum_{j=1}^{l} \alpha_j D_{ij} = (0{,}000133666 \times 74.8123) + (0{,}000133666 \times 47{,}2753)$$
$$+ (0{,}000133666 \times 49{,}9877) + (0{,}000133666 \times -25{,}6012)$$
$$+ (0{,}000133666 \times -36{,}3701) + (0{,}000133666 \times -23{,}3861)$$
$$+ (0{,}000133666 \times -19{,}7201) + (0{,}000133666 \times -19{,}0639)$$
$$+ (0{,}000133666 \times -25{,}6012)$$

$$= 0{,}00298524$$

Dari hasil perhitungan tersebut maka didapatkan nilai *Ei* pada masing-masing iterasi ke 2 pada level 1 dan 2 pada Tabel 4.7.

Tabel 4.7 Hasil Perhitungan Nilai *Ei* Level 1

Data Training	Iterasi 2
	Level 1
A23	0,00298524
A24	0,00182671
A25	0,00034349
A73	0,00254091
A74	0,00087263
A75	0,00254930
A102	0,00284538
A101	0,00174907
A104	0,00154374

Tabel 4.7 menunjukan hasil dari perhitungan nilai *Ei* pada iterasi ke-2 pada masing-masing level 1.

b. Menghitung nilai delta alpha

Menghitung nilai delta alpha dengan cara melakukan perkalian antar nilai gamma dengan hasil dari 1 dikurang dengan nilai *Ei* yang ke-*i* dan dibandingkan dengan nilai alpha ke-*i* yang akan dicari nilai maksimumnya. Setelah nilai maksimumnya didapatkan akan dibandingkan dengan nilai variabel *C* yang dikurangkan dengan nilai alpha ke-*i*. adapun perhitungan delta alpha pada persamaan 2.20 sebagai berikut:

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

Dari persamaan 2.20 maka ada beberapa nilai yang sudah didapatkan seperti nilai *C* = 1, *Ei* = 0,00298524 ,α merupakan nilai alpha ke-*j* hasil dari perhitungan sebelumnya yaitu 0,000133666, dan nilai γ= 0,000133666 didapatkan dari persamaan 2.22. berikut perhitungan delta alpha untuk data *training* A23.

$$\delta\alpha_j = min\{max[\gamma(1-E_i), -\alpha_j], C-\alpha_j\}$$
$$min\{max[0{,}000133666 \times (1-0{,}00298524), -0{,}000133666], 1 - 0{,}000133666\} = 0{,}00013327$$

Apabila telah melakukan perhitungan pada data *training* A23, maka lakukan perhitungan selanjutnya terhadap semua data *training*. Sehingga didapatkan nilai delta alpha dengan iterasi sebanyak 2 kali untuk 9 data *training*, ditunjukkan pada Tabel 4.8.

Tabel 4.8 Hasil Menghitung Nilai Delta Alpha Level 1

δαi	2
A23	0,00013327
A24	0,00013342
A25	0,00013362
A73	0,00013333
A74	0,00013355
A75	0,00013333
A102	0,00013329
A103	0,00013343
A104	0,00013346

Tabel 4.8 menunjukkan hasil dari menghitung nilai delta alpha pada iterasi yang dilakukan yang dilakukan sebanyak dua kali untuk medapatkan nilai delta alphanya, sehingga nantinya Tabel 4.8 akan digunakan untuk menentukan nilai alpha dari delta alpha i.

c. Menghitung nilai alpha

Menghitung nilai alpha baru dengan cara menambahkan nilai delta alpha ke-*i* dengan nilai alpha ke-*i* sebelumnya, pada perhitungan nilai alpha maka persamaan yang digunakan persamaan 2.21. Sehingga perhitungan nilai alpha sebagai berikut :

$$\alpha_j = \alpha_j + \delta\alpha_j$$

$$\alpha_j + \delta\alpha_j = 0{,}000133666 + 0{,}00013327 = 0{,}00026693$$

Dari hasil perhitungan diatas maka nilai α_j merupakan nilai alpha data ke-j hasil dari perhitungan sebelumnya dan $\delta\alpha_j$ merupakan nilai delta alpha data ke -j. Sehingga menghasilkan nilai 0,00026693, hasil perhitungan nilai alpha tersebut

merupakan perhitungan alpha di iterasi sebanyak 2 kali untuk 9 data *training.* Sehingga didapatkan nilai alpha pada Tabel 4.9

Tabel 4.9 Hasil Menghitung Nilai Alpha Level 1

α_j	2
A23	0,00026693
A24	0,00026709
A25	0,00026729
A73	0,00026699
A74	0,00026722
A75	0,00026699
A102	0,00026695
A103	0,00026710
A104	0,00026713

Tabel 4.10 merupakan hasil dari nilai alpha dari perhitungan persamaan 2.21, yang akan digunakan untuk perhitungan selanjutnya yaitu menentukan *support vector.*

Langkah 5 : Menentukan *Support Vector* Level 1

Support vector merupakan perhitungan yang ditentukan dari nilai αi pada iterasi terakhir, jika data training yang memiliki nilai αi lebih dari nol bisa dijadikan *support vector*. Pada perhitungan nantinya akan menggunakan nilai αi baru yang merupakan hasil perhitungan dari iterasi terakhir atau dalam perhitungan ini iterasi ke-2. Nilai αi yang dihasilkan dari perhitungan iterasi ke-2 menunjukkan 9 data *training* yang memiliki nilai αi lebih dari nol, sehingga semua data *training* tersebut merupakan *support vector*, yang mana nilai αi dapat ditunjukkan pada Tabel 4.10.

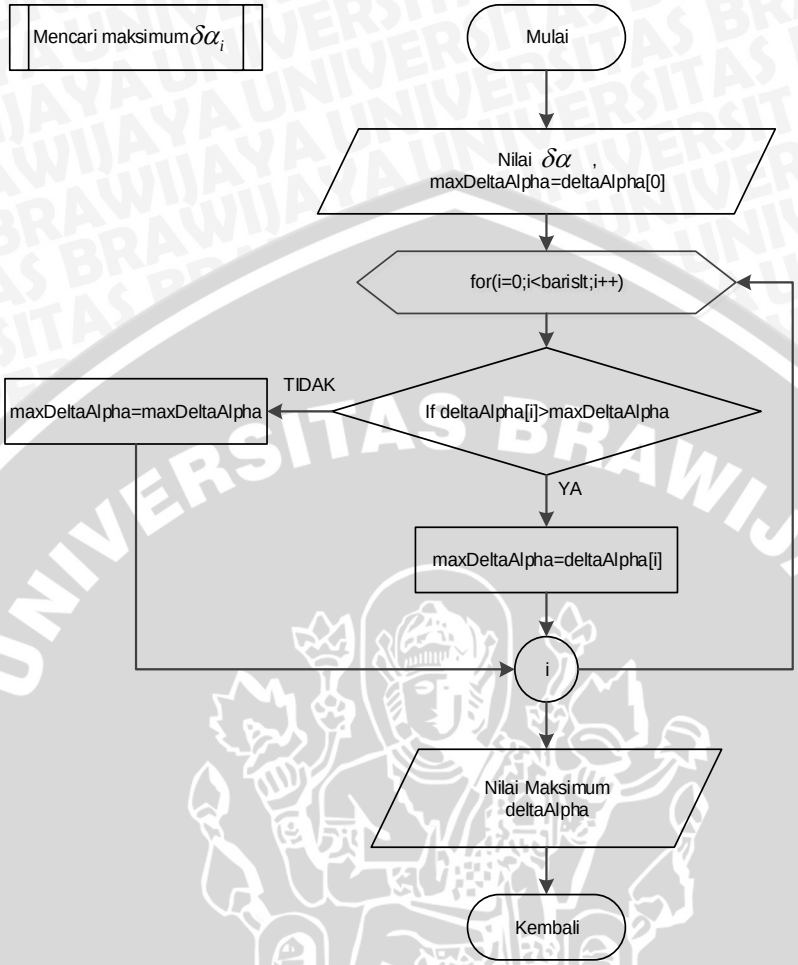
Tabel 4.10 Nilai α Pada Masing-masing Data Training Dari Iterasi ke-2 Level 1

Data Training SV	αi
A23	0,00026693
A24	0,00026709
A25	0,00026729
A73	0,00026699
A74	0,00026722
A75	0,00026699
A102	0,00026695
A103	0,00026710
A104	0,00026713

Tabel 4.10 merupakan hasil dari nilai alpha, yang akan digunakan pada perhitungan selanjutnya.

Adapun untuk memenuhi syarat untuk berhentinya iterasi akan dijelaskan pada pencarian nilai maksimum dari nilai *deltaalpha,* yang akan dijelaskan pada Gambar 4.8 sebagai berikut.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA



Gambar 4.8 Diagram Alir Maksimum Nilai Delta Alpha

Gambar 4.8 menjelaskan langkah-langkah mencari nilai dari maksimum delta alpha.

Langkah 6: Menghitung Nilai Bobot Data *Training* Level 1

Nilai bobot data *training* terdapat 2 bagian, yaitu terdiri dari bagian pertama bobot data *training* di-*dot product* dengan menggunakan nilai data *training* alpha tertinggi kepada kelas positif ($W.x^+$) dan bagian kedua bobot data *training* di-*dot product* dengan menggunakan nilai data *training* alpha tertinggi pada kelas negatif ($W.x^-$). Perhitungan bobot kelas negatif maupun bobot kelas negatif menggunakan perhitungan kernel dari data *training* yang memiliki kelas positif dan perhitungan kernel dari data *training* yang memiliki kelas negatif untuk masing-masing bobot.

Perhitungan pada setiap bobot bertujuan agar setiap kelas yaitu kelas positif dan negatif untuk tahap perhitungan selanjutnya. Nilai alpha yang akan digunakan merupakan nilai alpha yang dihasilkan pada perhitungan iterasi yang dilakukan sebanyak dua kali atau iterasi ke-2.

Adapun perhitungan untuk menentukan kelas positif dan negatif, yang dilakukan pada data *training* A23 dengan nilai $K(x^+, x_i)$ sebagai berikut :

$$wx_i^+ = \sum_{i=1}^{n} \alpha_i\, y_i K(x^+, x_i) = 0{,}00026693 \text{ x } 1 \text{ x } 49{,}73775625 = 0{,}013276668$$

Perhitungan diatas menunjukan perhitungan bobot kelas positif dengan nilai variabel α_i adalah nilai alpha data ke-i, variabel y_i merupakan kelas dari data ke-*i*, dan perhitungan $K(x^+, x_i)$ adalah nilai kernel dari data kelas positif ke-*i* yang memiliki nilai alpha terbesar yang ada pada kelas positif. Kemudian lakukan perhitungan seterusnya untuk bobot kelas positif dengan perhitungan yang sama sehingga hasil dari perhitungan bobot kelas positif dari semua data *training* dapat dilihat pada Tabel 4.11 yang merupakan hasil dari perhitungan bobot kelas positif sebagai berikut :

Tabel 4.11 Nilai Bobot Kelas Positif Data *Training* Level 1

Data Training	$w.x_i^+$
A23	0,013276668
A24	0,011210816
A25	0,013202494
A73	-0,006405592
A74	-0,009412468
A75	-0,005858697
A102	-0,004790672
A103	-0,00418589
A104	-0,006146573

Tabel 4.11 merupakan hasil dari perhitungan nilai bobot kelas positif, sehingga akan dilanjutkan untuk menghitung nilai kelas bobot negatif pada data *training* A23 dengan nilai $K(x^-, x_i)$, sehingga perhitungannya sebagai berikut:

$$wx_i^- = \sum_{i=1}^{n} \alpha_i\, y_i K(x^-, x_i) = 0{,}00026693 \text{ x } 1 \text{ x } 36{,}1201 = 0{,}009641661$$

Perhitungan diatas dengan nilai $K(x^-, x_i)$ merupakan nilai kernel dari data kelas negatif ke-*i* yang mempunyai nilai alpha terbesar pada kelas negatif, dan dilakukan perhitungan semua data *training* dan hasilnya pada Tabel 4.13.

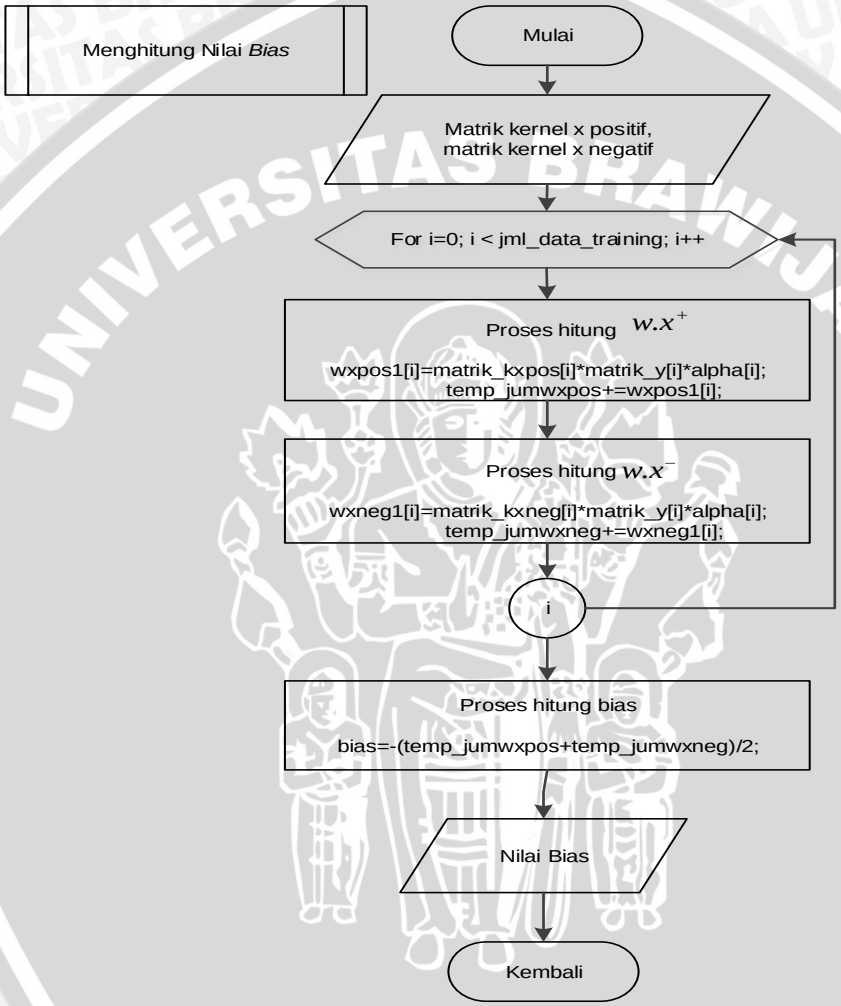
Tabel 4.12 Nilai Bobot Kelas Negatif Data *Training* Level 1

Data Training	$w.x_i^-$
A23	0,009641661
A24	0,008040905
A25	0,00941496
A73	-0,0048467
A74	-0,007902426
A75	-0,00468174
A102	-0,00393765

A103	-0,00308766
A104	-0,004182355

Langkah 7 Menghitung Nilai Bias Level 1

Menghitung nilai bias menggunakan beberapa langkah yang harus dihitung terlebih dahulu, adapun langkah-langkah perhitungannya digambarkan kedalam diagram alir pada Gambar 4.9, nantinya dalam mencari nilai bias membutuhkan jumlah dari kelas bobot positif dan jumlah dari kelas bobot negatif.



Gambar 4.9 Diagram Alir Menghitung Nilai Bias

Gambar 4.9 menunjukkan tahapan perhitungan nilai bias,yang mana nilai bobot data *training* didapatkan dari hasil perkalian alpha, nilai kernel dari data dengan kelas positif dan negatif, selanjutnya nilai bobot yang telah di-*dot product* dengan data pada kelas positif maupun data pada kelas negatif dijumlahkan dan hasil dari perjumlahannya di kalikan dengan nilai -0,5. Nilai bias diasumsikan dalam variabel *b*.

Nilai bias didapatkan dengan cara jumlah bobot kelas positif di tambah dengan jumlah kelas bobot negatif, kemudian hasil jumlah dari bobot kelas positif

dan bobot kelas negatif dikalikan dengan -0,5 (minus setengah), adapun perhitungan nilai bias sebagai berikut:

$$b = -\frac{1}{2}\left(\sum_{i=1}^{n} wx_i^{+} + \sum_{i=1}^{n} wx_i^{-}\right)$$

$$= -\frac{1}{2}(0{,}000890085 + (-0{,}001541007))$$

$$= 0{,}000325461$$

Perhitungan diatas yang mana nilai *b* adalah nilai bias, , $\sum_{i=1}^{n} wx_i^{+}$ merupakan jumlah dari bobot kelas positif, dan $\sum_{i=1}^{n} wx_i^{-}$ merupakan jumlah dari bobot kelas negatif. Sehingga nilai bias menunjukkan adangka 0,000325461.

Langkah 8 : Menghitung Nilai dari Fungsi *f(x)* Level 1

Menghitung nilai dari fungsi *f(x)* dilakukan kepada data *testing* dengan cara melakukan penjumlahan nilai dari total bobot data *testing* dengan nilai bias yang sudah dihitung pada proses sebelumnya, dan hasil akhir nanti berupa tanda dari nilai yang dihasilkan oleh fungsi *f(x).*

Perhitungan nilai *f(x)* dengan syarat jika pada level satu bernilai positif maka akan masuk kelas +1 dan masuk kelas A atau kelas penerima beasiswa sangat layak, dan jika bernilai negatif maka masuk kelas -1, apabila pada level satu bernilai negatif maka akan dihitung lagi pada level selanjutnya, jika pada level dua bernilai positif maka akan masuk kelas +1 dan masuk kelas B atau penerima beasiswa layak, jika pada level dua bernilai negatif maka akan masuk kelas -1 dan masuk dalam kelas C atau tidak layak.

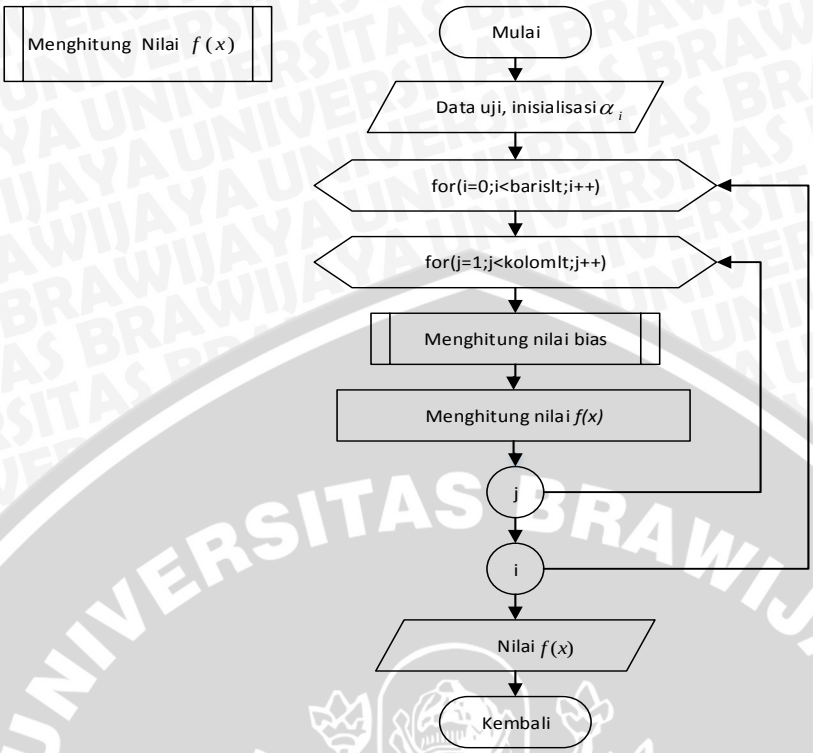
Adapun data *testing* yang digunakan dalam perhitungan nilai *f(x)* sebanyak 3 data *testing* dari 12 data, data *testing* tersebut ditunjukkan pada Tabel 4.13.

Tabel 4.13 Penilaian Alternatif atau Normalisasi Untuk Data Testing Level 1

Ai	C1	C2	C3	C4	C5	....	C11	C12	C13	KELAS
A14	0,775	0,100	0,775	1,000	1,000	....	0,400	1,000	1,000	A
A32	1,000	0,100	0,775	1,000	0,400	....	0,700	1,000	1,000	A
A64	0,550	0,100	0,775	0,550	0,400	....	0,400	1,000	0,100	B
A65	0,775	0,100	0,550	1,000	1,000	....	0,400	1,000	0,100	B
A99	0,775	0,100	0,775	0,550	0,700	....	0,400	1,000	0,100	C

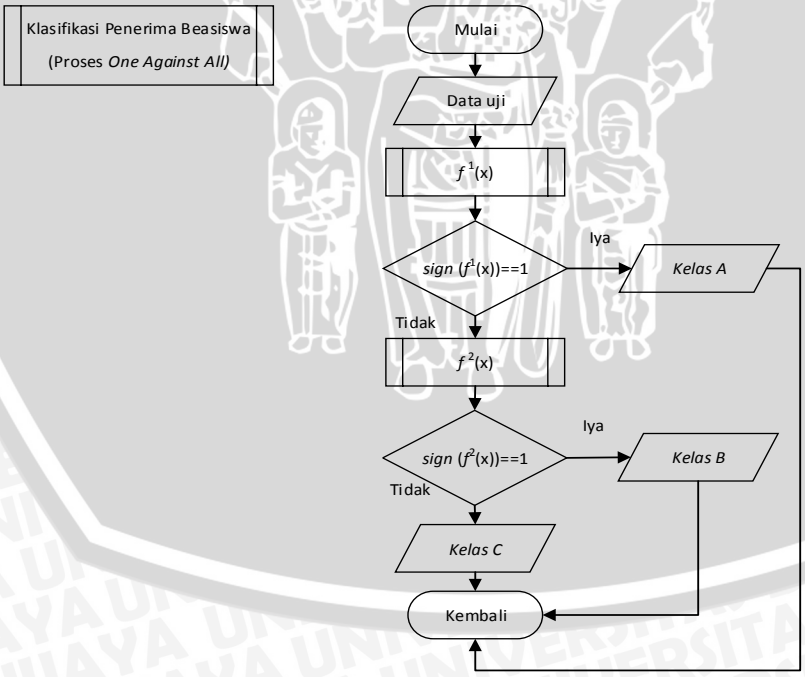
Sumber : [Wawancara]

Tabel 4.13 merupakan data *testing* yang akan digunakan dalam perhitungan nilai *f(x)*, adapun tahapan-tahapan perhitungan nilai *f(x)* digambarkan dalam diagram alir pada Gambar 4.10 sebagai berikut :



Gambar 4.10 Diagram Alir Menghitung Nilai dari Fungsi *f(x)*

Gambar 4.10 merupakan tahapan-tahapan perhitungan nilai *f(x)* yang digambarkan kedalam diagram alir.



Gambar 4.11 Proses *One-Against-All*

Gambar 4.11 merupakan proses penentuan kelas dari *one-against-all,* untuk menentukan kelas A, B dan C.

Nilai bobot yang ada pada *testing* didapatkan dari perhitungan nilai alpha baru hasil dari proses iterasi pada proses sebelumnya yang dikalikan dengan kelas data *testing* dari pakar, selanjutnya lakukan proses perkalian dengan fungsi kernel dari data *testing,* sehingga didapatkan nilai *f(x)* pada persamaan 2.12.

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

a. Mencari Nilai *f(x)* data uji 1 kelas A level 1

Perhitungan kernel $K(x, x_i)$ dilakukan pada data *testing* A14 dengan kernel *Polynomial Degree 2.* Dari persamaan kernel tersebut maka berikut perhitungan kernel yang dilakukan pada data *testing* A14, sebagai berikut :

$$\begin{aligned} K(x, x_i) = {} & [(0{,}775 \text{ x } 1{,}000) + (0{,}100 \text{ x } 0{,}100) + (0{,}775 \text{ x } 1{,}000) \\ & + (1{,}000 \text{ x } 0{,}550) + (1{,}000 \text{ x } 1{,}000) \\ & + (0{,}100 \text{ x } 0{,}100) + (0{,}100 \text{ x } 0{,}100) \\ & + (0{,}100 \text{ x } 0{,}100) + (0{,}550 \text{ x } 0{,}550) \\ & + (0{,}400 \text{ x } 1{,}000) + (0{,}400 \text{ x } 1{,}000) \\ & + (1{,}000 \text{ x } 1{,}000) + (1{,}000 \text{ x } 1{,}000)]^2 \end{aligned}$$

= 40,10055625

Perhitungan kernel untuk data *testing* seperti diatas, dan lakukan terhadap semua data *testing* sehingga menghasilkan nilai $K(x, x_i)$ pada Tabel 4.14

Tabel 4.14 Nilai Kernel Terhadap Masing-masing Data *Training* Level 1

SV	k(xi,x)
A23	40,10055625
A24	37,92250352
A25	37,14140664
A73	20,13765625
A74	26,55469727
A75	17,08272227
A102	11,93270664
A103	9,773439063
A104	18,361225

Tabel 4.14 merupakan hasil dari perhitungan kernel terhadap masing-masing data *training.* Apabila sudah melakukan perhitungan tersebut maka tahap selanjutnya menghitung jumlah bobot pada persamaan untuk data *testing* A23 sebagai berikut :

$$\alpha_i y_i K(x, x_i) = 0{,}00026693 \text{ x } 1 \text{ x } 40{,}100055625 = 0{,}0107200$$

Dari perhitungan diatas maka didapatkan nilai bobot A23 dengan nilai 0,010704177, dan lakukan perhitungan terhadap data yang lainnya, sehingga menghasilkan hasil perhitungan pada Tabel 4.15

Tabel 4.15 Nilai Bobot Data Terhadap Masing-masing Data *Training* Level 1

ID	αiyiK(xi,x)
A23	0,000107200
A24	0,000101378
A25	0,000099291
A73	-0,000053834
A74	-0,000070989
A75	-0,000045667
A102	-0,000031900
A103	-0,000026127
A104	-0,000049085

Apabila sudah mendapatkan hasil nilai bobot maka perhitungan selanjutnya menghitung nilai *f(x)* dengan persamaan 2.12 sebagai berikut:

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

$$f(x) = sign((0{,}000030267) + 0{,}000325461\ )$$

$$f(x) = sign(\ 0{,}000033553) = \ 1$$

Variabel $\sum_{i=1}^{n} \alpha_i\, y_i K(x, x_i)$ merupakan jumlah dari Tabel 4.15 yang merupakan jumlah nilai bobot dari semua data yang bernilai 0,000030267 sehingga nilai *f(x)* bernilai $sign(\sum_{i=1}^{n} \alpha_i\, y_i K(x, x_i) + b) = 1$ yang artinya termasuk kelas +1 yang masuk kelas penerima kelas sangat layak, dan tidak dilanjutkan perhitungannya pada level selanjutnya.

b. Mencari Nilai *f(x)* data uji 2 kelas A level 1

Perhitungan kernel $K(x, x_i)$ dilakukan pada data *testing* A32 dengan kernel *Polynomial Degree 2.* Dari persamaan kernel tersebut maka berikut perhitungan kernel yang dilakukan pada data *testing* A32, sebagai berikut :

$$\begin{aligned} K(x, x_i) = \ & [(1{,}000 \text{ x } 1{,}000) + (0{,}100 \text{ x } 0{,}100) + (0{,}775 \text{ x } 1{,}000) \\ & + (1{,}000 \text{ x } 0{,}550) + (0{,}400 \text{ x } 1{,}000) \ + (0{,}100 \text{ x } 0{,}100) \\ & + (0{,}100 \text{ x } 0{,}100) + (0{,}100 \text{ x } 0{,}100) + (0{,}100 \text{ x } 0{,}550) \\ & + (0{,}400 \text{ x } 1{,}000) + (0{,}700 \text{ x } 1{,}000) + (1{,}000 \text{ x } 1{,}000) \\ & + (1{,}000 \text{ x } 1{,}000)]^2 \\ & = 36{,}1201 \end{aligned}$$

Perhitungan kernel untuk data *testing* seperti diatas, dan lakukan terhadap semua data *testing* sehingga menghasilkan nilai $K(x, x_i)$ pada Tabel 4.16.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 4.16 Nilai Kernel Terhadap Masing-masing Data *Training* Level 1

SV	k(xi,x)
A23	36,1201
A24	32,433025
A25	31,14337539
A73	17,11373477
A74	24,71332656
A75	16,01500352
A102	10,70925625
A103	8,244076563
A104	13,18143789

Tabel 4.16 merupakan hasil dari perhitungan kernel terhadap masing-masing data *training.* Apabila sudah melakukan perhitungan tersebut maka tahap selanjutnya menghitung jumlah bobot pada persamaan untuk data *testing* A23 sebagai berikut :

$$\alpha_i y_i K(x, x_i) = 0{,}00026693 \text{ x } 1 \text{ x } 36{,}1201 = 0{,}000096559$$

Dari perhitungan diatas maka didapatkan nilai bobot A23 dengan nilai 0,000096559, dan lakukan perhitungan terhadap data yang lainnya, sehingga menghasilkan hasil perhitungan pada Tabel 4.17

Tabel 4.17 Nilai Bobot Data Terhadap Masing-masing Data *Training* Level 1

ID	αiyiK(xi,x)
A23	0,000096559
A24	0,000086703
A25	0,000083256
A73	-0,000045750
A74	-0,000066066
A75	-0,000042813
A102	-0,000028629
A103	-0,000022039
A104	-0,000035238

Apabila sudah mendapatkan hasil nilai bobot maka perhitungan selanjutnya menghitung nilai *f(x)* dengan persamaan 2.12 sebagai berikut:

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

$$f(x) = sign((0{,}000025984) + 0{,}000325461\ )$$

$$f(x) = sign(\ 0{,}000029270) = 1$$

Variabel $\sum_{i=1}^{n} \alpha_i\, y_i K(x, x_i)$ merupakan jumlah dari Tabel 4.17 yang merupakan jumlah nilai bobot dari semua data yang bernilai 0,000029270 sehingga nilai *f(x)* bernilai $sign(\sum_{i=1}^{n} \alpha_i\, y_i K(x, x_i) + b) = 1$ yang artinya termasuk kelas +1 yang masuk kelas penerima kelas sangat layak, dan tidak dilanjutkan perhitungannya pada level selanjutnya.

c. Mencari Nilai *f(x)* data uji 1 kelas B level 1

Perhitungan kernel $K(x, x_i)$ dilakukan pada data *testing* A64 dengan kernel *Polynomial Degree 2.* Dari persamaan kernel tersebut maka berikut perhitungan kernel yang dilakukan pada data *testing* A64, sebagai berikut :

$$\begin{aligned} K(x, x_i) = {} & [(0{,}550 \text{ x } 1{,}000) + (0{,}100 \text{ x } 0{,}100) + (0{,}775 \text{ x } 1{,}000) \\ & + (0{,}550 \text{ x } 0{,}550) + (0{,}400 \text{ x } 1{,}000) + (0{,}100 \text{ x } 0{,}100) \\ & + (0{,}100 \text{ x } 0{,}100) + (0{,}100 \text{ x } 0{,}100) + (0{,}100 \text{ x } 0{,}550) \\ & + (1{,}000 \text{ x } 1{,}000) + (0{,}400 \text{ x } 1{,}000) + (1{,}000 \text{ x } 1{,}000) \\ & + (0{,}100 \text{ x } 1{,}000)]^2 \\ & = 22{,}207656 \end{aligned}$$

Perhitungan kernel untuk data *testing* seperti diatas, dan lakukan terhadap semua data *testing* sehingga menghasilkan nilai $K(x, x_i)$ pada Tabel 4.18.

Tabel 4.18 Nilai Kernel Terhadap Masing-masing Data *Training* Level 1

SV	k(xi,x)
A23	22,20765625
A24	16,21068906
A25	19,12422227
A73	12,58919102
A74	14,938225
A75	11,72634414
A102	10,10445156
A103	9,494101563
A104	11,88094727

Tabel 4.18 merupakan hasil dari perhitungan kernel terhadap masing-masing data *training.* Apabila sudah melakukan perhitungan tersebut maka tahap selanjutnya menghitung jumlah bobot pada persamaan untuk data *testing* A64 sebagai berikut :

$$\alpha_i y_i K(x, x_i) = 0{,}00026693 \text{ x } 1 \text{ x } 22{,}20765625 = 0{,}000059367$$

Dari perhitungan diatas maka didapatkan nilai bobot A23 dengan nilai 0,000059367, dan lakukan perhitungan terhadap data yang lainnya, sehingga menghasilkan hasil perhitungan pada Tabel 4.19.

Tabel 4.19 Nilai Bobot Data Terhadap Masing-masing Data *Training* Level 1

ID	αiyiK(xi,x)

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

A23	0,000059367
A24	0,000043336
A25	0,000051125
A73	-0,000033655
A74	-0,000039935
A75	-0,000031348
A102	-0,000027012
A103	-0,000025381
A104	-0,000031761

Apabila sudah mendapatkan hasil nilai bobot maka perhitungan selanjutnya menghitung nilai *f(x)* dengan persamaan 2.12 sebagai berikut:

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

$$f(x) = sign((-0{,}000035263) + 0{,}000325461\ )$$

$$f(x) = sign(-0{,}000031976) = -1$$

Variabel $\sum_{i=1}^{n} \alpha_i\, y_i K(x, x_i)$ merupakan jumlah dari Tabel 4.19 yang merupakan jumlah nilai bobot dari semua data yang bernilai $-0{,}000035263$ sehingga nilai *f(x)* bernilai $sign(\sum_{i=1}^{n} \alpha_i\, y_i K(x, x_i) + b) = 1$ yang artinya termasuk kelas -1 yang masuk kelas penerima kelas diantara layak dan tidak layak, maka dilanjutkan perhitungannya pada level selanjutnya untuk mencari kelas layak.

d. Mencari Nilai *f(x)* data uji 2 kelas B level 1

Perhitungan kernel $K(x, x_i)$ dilakukan pada data *testing* A65 dengan kernel *Polynomial Degree 2.* Dari persamaan kernel tersebut maka berikut perhitungan kernel yang dilakukan pada data *testing* A65, sebagai berikut :

$$\begin{aligned} K(x, x_i) = {} & [(0{,}775 \text{ x } 1{,}000) + (0{,}100 \text{ x } 0{,}100) + (0{,}550 \text{ x } 1{,}000) \\ & + (1{,}000 \text{ x } 0{,}550) + (1{,}000 \text{ x } 1{,}000) + (0{,}100 \text{ x } 0{,}100) \\ & + (0{,}100 \text{ x } 0{,}100) + (0{,}100 \text{ x } 0{,}100) + (0{,}100 \text{ x } 0{,}550) \\ & + (1{,}000 \text{ x } 1{,}000) + (0{,}400 \text{ x } 1{,}000) + (1{,}000 \text{ x } 1{,}000) \\ & + (0{,}100 \text{ x } 1{,}000)]^2 \\ & = 30{,}9136 \end{aligned}$$

Perhitungan kernel untuk data *testing* seperti diatas, dan lakukan terhadap semua data *testing* sehingga menghasilkan nilai $K(x, x_i)$ pada Tabel 4.20.

Tabel 4.20 Nilai Kernel Terhadap Masing-masing Data *Training* Level 1

SV	k(xi,x)

A23	30,9136
A24	25,2569066
A25	28,8637563
A73	22,0841254
A74	21,9433691
A75	18,879025
A102	15,1856348
A103	10,660225
A104	17,5822973

Tabel 4.20 merupakan hasil dari perhitungan kernel terhadap masing-masing data *training.* Apabila sudah melakukan perhitungan tersebut maka tahap selanjutnya menghitung jumlah bobot pada persamaan untuk data *testing* A64 sebagai berikut :

$$\alpha_i y_i K(x, x_i) = 0{,}00026693 \text{ x } 1 \text{ x } 30{,}9136 = 0{,}000082641$$

Dari perhitungan diatas maka didapatkan nilai bobot A23 dengan nilai 0,000082641 , dan lakukan perhitungan terhadap data yang lainnya, sehingga menghasilkan hasil perhitungan pada Tabel 4.21.

Tabel 4.21 Nilai Bobot Data Terhadap Masing-masing Data *Training* Level 1

ID	αiyiK(xi,x)
A23	0,000082641
A24	0,000067519
A25	0,000077162
A73	-0,000059037
A74	-0,000058661
A75	-0,000050469
A102	-0,000040596
A103	-0,000028498
A104	-0,000047003

Apabila sudah mendapatkan hasil nilai bobot maka perhitungan selanjutnya menghitung nilai *f(x)* dengan persamaan 2.12 sebagai berikut:

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

$$f(x) = sign((-0{,}000056942) + 0{,}000325461\ )$$

$$f(x) = sign(-0{,}000053656) = -1$$

Variabel $\sum_{i=1}^{n} \alpha_i\, y_i K(x, x_i)$ merupakan jumlah dari Tabel 4.21 yang merupakan jumlah nilai bobot dari semua data yang bernilai $-0{,}000056942$ sehingga nilai *f(x)* bernilai $sign(\sum_{i=1}^{n} \alpha_i\, y_i K(x, x_i) + b) = -1$ yang artinya termasuk kelas -1 yang masuk kelas penerima kelas diantara layak dan tidak layak,

maka dilanjutkan perhitungannya pada level selanjutnya untuk mencari kelas layak.

e. Mencari Nilai *f(x)* data uji 1 kelas C level 1

Perhitungan kernel $K(x, x_i)$ dilakukan pada data *testing* A99 dengan kernel *Polynomial Degree 2.* Dari persamaan kernel tersebut maka berikut perhitungan kernel yang dilakukan pada data *testing* A99, sebagai berikut :

$$\begin{aligned} K(x, x_i) = \ & [(0{,}775 \text{ x } 1{,}000) + (0{,}100 \text{ x } 0{,}100) + (0{,}775 \text{ x } 1{,}000) \\ & + (0{,}5500 \text{ x } 0{,}550) + (0{,}700 \text{ x } 1{,}000) + (0{,}100 \text{ x } 0{,}100) \\ & + (0{,}100 \text{ x } 0{,}100) + (0{,}100 \text{ x } 0{,}100) + (0{,}100 \text{ x } 0{,}550) \\ & + (1{,}000 \text{ x } 1{,}000) + (0{,}400 \text{ x } 1{,}000) + (1{,}000 \text{ x } 1{,}000) \\ & + (0{,}100 \text{ x } 1{,}000)]^2 \\ & = 27{,}431406 \end{aligned}$$

Perhitungan kernel untuk data *testing* seperti diatas, dan lakukan terhadap semua data *testing* sehingga menghasilkan nilai $K(x, x_i)$ pada Tabel 4.22.

Tabel 4.22 Nilai Kernel Terhadap Masing-masing Data *Training* Level 1

SV	k(xi,x)
A23	27,4314063
A24	20,2556254
A25	23,0100098
A73	16,1805063
A74	17,206941
A75	14,1235035
A102	11,9845785
A103	10,3925641
A104	14,5924

Tabel 4.22 merupakan hasil dari perhitungan kernel terhadap masing-masing data *training.* Apabila sudah melakukan perhitungan tersebut maka tahap selanjutnya menghitung jumlah bobot pada persamaan untuk data *testing* A64 sebagai berikut :

$$\alpha_i y_i K(x, x_i) = 0{,}00026693 \text{ x } 1 \text{ x } 27{,}431406 = 0{,}000073332$$

Dari perhitungan diatas maka didapatkan nilai bobot A23 dengan nilai 0,000073332 , dan lakukan perhitungan terhadap data yang lainnya, sehingga menghasilkan hasil perhitungan pada Tabel 4.23.

Tabel 4.23 Nilai Bobot Data Terhadap Masing-masing Data *Training* Level 1

ID	αiyiK(xi,x)
A23	0,000073332

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

A24	0,000054149
A25	0,000061513
A73	-0,000043255
A74	-0,000046000
A75	-0,000037756
A102	-0,000032038
A103	-0,000027782
A104	-0,000039010

Apabila sudah mendapatkan hasil nilai bobot maka perhitungan selanjutnya menghitung nilai *f*(*x*) dengan persamaan 2.12 sebagai berikut:

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

$$f(x) = sign((-0{,}000036847) + 0{,}000325461\ )$$

$$f(x) = sign(-0{,}000033561) = -1$$

Variabel $\sum_{i=1}^{n} \alpha_i\, y_i K(x, x_i)$ merupakan jumlah dari Tabel 4.21 yang merupakan jumlah nilai bobot dari semua data yang bernilai $-0{,}000036847$ sehingga nilai *f(x)* bernilai $sign(\sum_{i=1}^{n} \alpha_i\, y_i K(x, x_i) + b) = -1$ yang artinya termasuk kelas -1 yang masuk kelas penerima kelas diantara layak dan tidak layak, maka dilanjutkan perhitungannya pada level selanjutnya untuk mencari kelas layak.

Sehingga untuk perhitungan level 2, akan lianjutkan kelas layak dan tidak layak untuk data *training*, dan untuk data *testing* nya menggunakan *id* yang masih bernilai -1, yaitu A64, A65 dan A99.

Langkah 9 : Menentukan Matriks Penilaian Alternatif Level 2

Matriks penilaian alternatif ini merupakan normalisasi dari data *training* maupun data *testing,* dari perhitungan level 1 sebelumnya, namun disini data yang digunakan untuk data *training* hanya menggunakan kelas layak dan tidak layak sedangkan untuk data *testing* menggunakan data dengan *id* A64, A65 dan A99. Adapun data *training* dan data *testing* ditunjukkan pada Tabel 4.24 dan Tabel 4.25.

Tabel 4.24 Penilaian Alternatif atau Normalisasi Data Training Level 2

Ai	C1	C2	C3	C4	C5	....	C11	C12	C13	KELAS
A73	0,775	0,100	0,325	1,000	1,000	....	0,400	1,000	0,100	B
A74	0,325	0,100	0,550	1,000	0,700	....	0,700	1,000	1,000	B
A75	0,550	0,100	0,325	1,000	0,700	....	0,700	1,000	0,100	B
A102	0,325	0,100	0,100	0,550	0,700	....	0,700	1,000	0,100	C
A103	0,100	0,100	0,550	0,100	0,400	....	0,700	1,000	0,100	C
A104	0,325	0,100	0,775	0,550	1,000	....	0,400	1,000	0,100	C

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 4.25 Penilaian Alternatif atau Normalisasi Data Testing Level 2

Ai	C1	C2	C3	C4	C5	....	C11	C12	C13	KELAS
A14	0,775	0,100	0,775	1,000	1,000	....	0,400	1,000	1,000	A
A32	1,000	0,100	0,775	1,000	0,400	....	0,700	1,000	1,000	A
A64	0,550	0,100	0,775	0,550	0,400	....	0,400	1,000	0,100	B
A65	0,775	0,100	0,550	1,000	1,000	....	0,400	1,000	0,100	B
A99	0,775	0,100	0,775	0,550	0,700	....	0,400	1,000	0,100	C

Tabel 4.24 dan Tabel 4.25 merupakan normalisasi dari data *training* dan data *testing* yang sudah dihitung pada perhitungan awal, dan digunakan untuk perhitungan kernel pada tahap selanjutnya.

Langkah 10 : Menghitung Matriks Kernel Level 2

Pada perhitungan matriks kernel level 2 ini akan dilakukan setelah semua data sampel pada data *training* telah dimasukkan kedalam sistem seleksi dan rekomendasi. Kemudian data *training* akan dilakukan perhitungan untuk mencari metriks kernel dengan menggunakan matriks *kernel Polynomial Degree d,* sama pada level 1 sebelumnya perhitungan matriks *kernel Polynomial Degree d* ini menggunakan persamaan 2.14.

$$K(x, x_i) = (x.x_i)^d$$

Nilai dari variabel *x* dan *y* yaitu merupakan dari 2 data *training* yang akan dicari nilai *dot Product* nya dengan menggunakan *kernel Polynomial Degree d*. Untuk baris pertama dan kolom pertama dati Tabel 4.26, kernel(0,0), sebagai berikut :

x = A73
y = A73

$$[x\ .\ y]^d = [(X_{C1} \times X_{C1}) + (X_{C2} \times X_{C2}) + (X_{C3} \times X_{C3}) + (X_{C4} \times X_{C4}) + (A23_{C5} \times X_{C5}) + (X_{C6} \times X_{C6}) + (X_{C7} \times X_{C7}) + (X_{C8} \times X_{C8}) + (X_{C9} \times X_{C9}) + (X_{C10} \times X_{C10}) + (X_{C11} \times X_{C11}) + (X_{C12} \times X_{C12}) + (X_{C13} \times X_{C14})]2$$

$$= [(0{,}775 \text{ x } 0{,}775) + (0{,}100 \text{ x } 0{,}100) + (0{,}325 \text{ x } 0{,}325) + (1{,}000 \text{ x } 1{,}000) + (1{,}000 \text{ x } 1{,}000) + (0{,}100 \text{ x } 0{,}100) + (0{,}100 \text{ x } 0{,}100) + (0{,}100 \text{ x } 0{,}100) + (0{,}100 \text{ x } 0{,}1000) + (0{,}700 \text{ x } 0{,}700) + (0{,}400 \text{ x } 0{,}400) + (1{,}000 \text{ x } 1{,}000) + (0{,}100 \text{ x } 0{,}100)]^2$$

$$= 19{,}50326406$$

Setelah melakukan perhitungan kernel pada A73 untuk baris pertama dan kolom pertama selesai, maka lakukan perhitungan seterusnya sehingga memperoleh nilai matriks pada *kernel Polynomial Degree d* pada Tabel 4.26.

Tabel 4.26 Matriks Kernel Polynomial Degree *d* Level 2

K(x,y)	A73	A74	....	A102	A103	A104
A73	19,50326406	18,15292539	....	12,77615664	8,07270156	14,50657656
A74	18,15292539	29,57320352	....	14,75040039	11,56000000	15,65685977
A75	16,49882852	17,53515625	....	12,25875156	8,46082656	12,64246914
A102	12,77615664	14,75040039	....	11,95862852	9,10530625	10,69698789
A103	8,07270156	11,56000000	....	9,10530625	11,05562500	10,53813906
A104	14,50657656	15,65685977	....	10,69698789	10,53813906	16,09012656

Tabel 4.26 menunjukkan hasil dari perkalian matriks kernel dari persamaan 2.14 yang didapatkan dari normalisasi data *training*.

Langkah 11 : Menghitung Matriks Hessian Level 2

Perhitungan pada matriks Hessian dilakukan dengan cara apabila matriks kernel kernel *Polynomial Degree d* sudah selesai, perhitungan untuk mencari nilai pada matriks Hessian ini akan menggunakan kernel kernel *Polynomial Degree 2,* adapun persamaan untuk menghitung matriks Hessian pada persamaan 2.18.

Perhitungan matriks Hessian ini ada beberapa nilai yang dapat diasumsikan yaitu λ = 0.5 (variabel skalar), variabel y_i adalah kelas data ke-*i*, variabel y_j adalah kelas data ke-j, dan $K(x_i, y_j)$ merupakan nilai kernel ke-*i,j*. Dengan rumus dari persamaan 2.18.

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

Dari persamaan tersebut, maka untuk perhitungan matriks Hessian pada A73 sebagai berikut :

Untuk kolom ke-1 baris ke-1:

$$y_i y_j (K(x_i, y_j) + \lambda^2) = 1 \text{ x } 1 \text{ x } (19{,}50326406 + 0{,}5^2) = 19{,}75326406$$

Untuk kolom ke-1 baris ke-2:

$$y_i y_j (K(x_i, y_j) + \lambda^2) = 1 \text{ x } 1 \text{ x } (18{,}15292539 + 0{,}5^2) = 18{,}40292539$$

Untuk kolom ke-1 baris ke-3:

$$y_i y_j (K(x_i, y_j) + \lambda^2) = 1 \text{ x } 1 \text{ x } (16{,}49882852 + 0{,}5^2) = 16{,}74882852$$

Dari beberapa contoh perhitungan matriks Hessian diatas makan didapatkan nilai dari matriks Hessian pada Tabel 4.27.

Tabel 4.27 Matriks Hessian Level 2

Dij	A73	A74	....	A102	A103	A104
A73	19,75326406	18,40292539	....	-13,02615664	-8,32270156	-14,75657656
A74	18,40292539	29,82320352	....	-15,00040039	-11,81000000	-15,90685977
A75	16,74882852	17,78515625	....	-12,50875156	-8,71082656	-12,89246914
A102	-13,02615664	-15,00040039	....	12,20862852	9,35530625	10,94698789

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

A103	-8,32270156	-11,81000000	**....**	9,35530625	11,30562500	10,78813906
A104	-14,75657656	-15,90685977	**....**	10,94698789	10,78813906	16,34012656

Tabel 4.27 merupakan hasil dari seluruh perhitungan matriks Hessian yang mana perhitungannya didapatkan dari perhitungan kernel selesai.

Langkah 11: Melakukan iterasi Level 2

Proses iterasi akan dilakukan 3 tahap perhitungan yang akan diulang sebanyak nilai *max(δαi)* < *ε* atau bisa juga diulang sebanyak jumlah *n* kali (*n* merupakan jumlah iterasi maksimum atau iterMAX). Proses pertama untuk melakukan proses iterasi yaitu melakukan perhitungan untuk mencari nilai *Ei* setiap data *training*. Kemudian proses selanjutnya atau proses kedua yaitu melakukan perhitungan terhadap *δαi*, proses terakhir atau proses ketiga adalah melakukan pencarian terhadap nilai *αi*.

Itermax dan epsilon yang memiliki nilai variabel pada masing-masingnya akan mempengaruhi iterasi yang nantinya akan dilakukan, jika nilai maksimum dari delta alpha lebih besar dari nilai variabel epsilon maka iterasi akan berhenti. Diasumsikan nilai iteraksi maksimum (*iterMAX*) = 2, epsilon (*ε*) = 0.001, dan nilai *α* = 0 (asumsi nilai alpha awal). Sebelum melakukan iterasi, nilai alpha baru akan dihitung dengan beberapa tahap seperti pada proses iterasi selama 1 kali, sehingga dapat menghasilkan nilai *Ei* = 0, *δαi* = 0,000335309, dan alpha baru αi = 0,000335309. Dari hasil nilai delta aplha baru nilai *max(|δαi|)* > *ε* maka dilakukan proses iterasi, Setelah mendapatkan nilai alpha baru maka nilai tersebut digunakan untuk perhitungan alpha pada proses iterasi pertama, adapun langkah-langkah iterasi pertama iterasi berikutnya sebagai berikut :

a. Menghitung nilai *Ei*

Menghitung nilai *Ei* dengan cara melakukan penjumlahan dari hasil perkalian matriks Hessian dengan alpha ke-*j*, yang mana proses tersebut merupakan proses iterasi awal dari metode *Sequential Training* SVM.

Tahapan dari langkah-langkah perhitungan nilai *Ei* dengan nilai Variabel α_j adalah alpha ke-*j* dengan hasil perhitungan sebelumnya yaitu 0,000335309 dan D_{ij} adalah nilai matriks Hessian baris ke-*i* dan kolom ke-*j*, dengan demikian didapatkan perhitungan nilai Ei sebagai berikut :

$$Ei = \sum_{j=1}^{l} \alpha_j D_{ij}$$

$$= (0{,}000335309 \times 19{,}75326) + (0{,}000335309 \times 18{,}40292) + (0{,}000335309 \times 16{,}74882) + (0{,}000335309 \times -13{,}02615) + (0{,}000335309 \times -8{,}32270) + (0{,}000335309 \times -14{,}75657)$$

$$= 0{,}00630368$$

Dari hasil perhitungan tersebut maka didapatkan nilai *Ei* pada masing-masing iterasi ke 2 pada level 2 pada Tabel 4.28.

Tabel 4.28 Hasil Perhitungan *Ei* Level 2

Data Training	Iterasi 2
	Level 1
A73	0,00630368
A74	0,00781071
A75	0,00542556
A102	-0,00269065
A101	0,00087366
A104	-0,00183771

Tabel 4.28 menunjukan hasil dari perhitungan nilai *Ei* pada iterasi ke-2 pada masing-masing level 2.

b. Menghitung nilai delta alpha

Menghitung nilai delta alpha dengan cara melakukan perkalian antar nilai gamma dengan hasil dari 1 dikurang dengan nilai *Ei* yang ke-*i* dan dibandingkan dengan nilai alpha ke-*i* yang akan dicari nilai maksimumnya. Setelah nilai maksimumnya didapatkan akan dibandingkan dengan nilai variabel *C* yang dikurangkan dengan nilai alpha ke-*i*.

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

Persamaan 2.20 maka ada beebrapa nilai yang sudah didapatkan seperti nilai *C* = 1, *Ei* = 0,00630368, α merupakan nilai alpha ke-*j* hasil dari perhitungan sebelumnya yaitu 0,000335309, dan nilai γ= 0,000335309 didapatkan dari persamaan 2.22. berikut perhitungan delta alpha untuk data *training* A73.

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

$$= min\{max[0{,}000335309 \times (1-0{,}00630368), -0{,}000335309], 1 - 0{,}000335309\}$$

$$= 0{,}000333196$$

Apabila telah melakukan perhitungan pada data *training* A73, maka lakukan perhitungan selanjutnya terhadap semua data *training*. Sehingga didapatkan nilai delta alpha dengan iterasi sebanyak 2 kali untuk 6 data *training*, ditunjukkan pada Tabel 4.29.

Tabel 4.29 Hasil Menghitung Nilai Delta Alpha Level 2

δαi	2
A73	0,000333196
A74	0,00033269
A75	0,00033349
A102	0,000336212
A103	0,000335016

A104	0,000335926

Tabel 4.29 menunjukkan hasil dari menghitung nilai delta alpha pada iterasi yang dilakukan yang dilakukan sebanyak dua kali untuk medapatkan nilai delta alphanya, sehingga nantinya Tabel 4.29 akan digunakan untuk menentukan nilai alpha dari delta alpha i.

c. Menghitung nilai alpha

Menghitung nilai alpha baru dengan cara menambahkan nilai delta alpha ke-*i* dengan nilai alpha ke-*i* sebelumnya.

perhitungan nilai alpha maka persamaan yang digunakan persamaan 2.21. sehingga perhitungan nilai alpha sebagai berikut :

$$\alpha_j = \alpha_j + \delta\alpha_j$$

$$\alpha_j + \delta\alpha_j = 0{,}000335309 + 0{,}000333196 = 0{,}000668505$$

Dari hasil perhitungan diatas maka nilai α_j merupakan nilai alpha data ke-j hasil dari perhitungan sebelumnya dan $\delta\alpha_j$ merupakan nilai delta alpha data ke -j. Sehingga menghasilkan nilai 0,000668505, hasil perhitungan nilai alpha tersebut merupakan perhitungan alpha di iterasi sebanyak 2 kali untuk 6 data *training.* Sehingga didapatkan nilai alpha pada Tabel 4.30

Tabel 4.30 Hasil Menghitung Nilai Alpha Level 2

α_j	**2**
A73	0,000668505
A74	0,000668
A75	0,0006688
A102	0,000671521
A103	0,000670326
A104	0,000671235

Tabel 4.30 merupakan hasil dari nilai alpha dari perhitungan persamaan 2.21, yang akan digunakan untuk perhitungan selanjutnya yaitu menentukan *support vector.*

Langkah 12 : Menentukan *Support Vector* Level 2

Support vector merupakan perhitungan yang ditentukan dari nilai αi pada iterasi terakhir, jika data training yang memiliki nilai αi lebih dari nol bisa dijadikan *support vector*. Pada perhitungan nantinya akan menggunakan nilai αi baru yang merupakan hasil perhitungan dari iterasi terakhir atau dalam perhitungan ini iterasi ke-2. Nilai αi yang dihasilkan dari perhitungan iterasi ke-2 menunjukkan 6 data *training* yang memiliki nilai αi lebih dari nol, sehingga semua data *training*

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

tersebut merupakan *support vector*, yang mana nilai αi dapat ditunjukkan pada Tabel 4.31.

Tabel 4.31 Nilai α Pada Masing-Masing Data Training Hasil Dari Iterasi ke-2 Level 2

Data Training	αi
SV	
A73	0,000668505
A74	0,000668
A75	0,0006688
A102	0,000671521
A103	0,000670326
A104	0,000671235

Langkah 13: Menghitung Nilai Bobot Data *Training* Level 2

Nilai bobot data *training* terdapat 2 bagian, yaitu terdiri dari bagian pertama bobot data *training* di-*dot product* dengan menggunakan nilai data *training* alpha tertinggi kepada kelas positif ($W.x^+$) dan bagian kedua bobot data *training* di-*dot product* dengan menggunakan nilai data *training* alpha tertinggi pada kelas negatif ($W.x^-$). Perhitungan bobot kelas negatif maupun bobot kelas negatif menggunakan perhitungan kernel dari data *training* yang memiliki kelas positif dan perhitungan kernel dari data *training* yang memiliki kelas negatif untuk masing-masing bobot.

Perhitungan pada setiap bobot bertujuan agar setiap kelas yaitu kelas positif dan negatif untuk tahap perhitungan selanjutnya. Nilai alpha yang akan digunakan merupakan nilai alpha yang dihasilkan pada perhitungan iterasi yang dilakukan sebanyak dua kali atau iterasi ke-2.

Adapun perhitungan untuk menentukan kelas positif dan negatif, yang dilakukan pada data *training* A73 dengan nilai $K(x^+, x_i)$ sebagai berikut :

$$wx_i^+ = \sum_{i=1}^{n} \alpha_i\, y_i K(x^+, x_i) = 0{,}000668505 \text{ x } 1 \text{ x } 16{,}498828 = 0{,}011029551$$

Perhitungan diatas menunjukan perhitungan bobot kelas positif dengan nilai variabel α_i adalah nilai alpha data ke-i, variabel y_i merupakan kelas dari data ke-*i*, dan perhitungan $K(x^+, x_i)$ adalah nilai kernel dari data kelas positif ke-*i* yang memiliki nilai alpha terbesar yang ada pada kelas positif. Kemudian lakukan perhitungan seterusnya untuk bobot kelas positif dengan perhitungan yang sama sehingga hasil dari perhitungan bobot kelas positif dari semua data *training* dapat dilihat pada Tabel 4.32 yang merupakan hasil dari perhitungan bobot kelas positif sebagai berikut :

Tabel 4.32 Nilai Kelas Positif Data *Training* Level 2

Data Training	$w.x_i^+$
A73	0,011029551
A74	0,01171348
A75	0,010372297

A102	-0,008232009
A103	-0,005671511
A104	-0,008486067

Tabel 4.32 merupakan hasil dari perhitungan nilai bobot kelas positif, sehingga akan dilanjutkan untuk menghitung nilai kelas bobot negatif pada data *training* A73 dengan nilai $K(x^-, x_i)$, sehingga perhitungannya sebagai berikut:

$$x_i^- = \sum_{i=1}^{n} \alpha_i \, y_i K(x^-, x_i) = 0{,}000668505 \text{ x } 1 \text{ x } 12{,}77615664 = 0{,}008540926$$

Perhitungan diatas dengan nilai $K(x^-, x_i)$ merupakan nilai kernel dari data kelas negatif ke-*i* yang mempunyai nilai alpha terbesar pada kelas negatif, dan dilakukan perhitungan semua data *training* dan hasilnya pada Tabel 4.33.

Tabel 4.33 Nilai Bobot Kelas Negatif Data *Training* Level 2

Data Training	$w.x_i^-$
A73	0,008540926
A74	0,009853264
A75	0,008198647
A102	-0,00803047
A103	-0,006103522
A104	-0,007180192

Langkah 14 Menghitung Nilai Bias Level 2

Menghitung nilai bias menggunakan beberapa langkah yang harus dihitung terlebih dahulu, nantinya dalam mencari nilai bias membutuhkan jumlah dari kelas bobot positif dan jumlah dari kelas bobot negatif.

nilai bobot data *training* didapatkan dari hasil perkalian alpha, nilai kernel dari data dengan kelas positif dan negatif, selanjutnya nilai bobot yang telah di-*dot product* dengan data pada kelas positif maupun data pada kelas negatif dijumlahkan dan hasil dari perjumlahannya di kalikan dengan nilai -0,5. Nilai bias diasumsikan dalam variabel *b*.

Nilai bias didapatkan dengan cara jumlah bobot kelas positif di tambah dengan jumlah kelas bobot negatif, kemudian hasil jumlah dari bobot kelas positif dan bobot kelas negatif dikalikan dengan -0,5 (minus setengah), adapun perhitungan nilai bias sebagai berikut:

$$\begin{aligned} b &= -\tfrac{1}{2}\left(\textstyle\sum_{i=1}^{n} wx_i^+ + \sum_{i=1}^{n} wx_i^-\right) \\ &= -\tfrac{1}{2}(0{,}010725742 + (-0{,}005278653)) \\ &= -0{,}008002197 \end{aligned}$$

Perhitungan diatas yang mana nilai *b* adalah nilai bias, , $\sum_{i=1}^{n} wx_i^+$ merupakan jumlah dari bobot kelas positif, dan $\sum_{i=1}^{n} wx_i^-$ merupakan jumlah dari bobot kelas negatif. Sehingga nilai bias menunjukkan angka $-0{,}008002197$.

Langkah 15 : Menghitung Nilai dari Fungsi *f(x)* Level 2

Menghitung nilai dari fungsi *f(x)* dilakukan kepada data *testing* dengan cara melakukan penjumlahan nilai dari total bobot data *testing* dengan nilai bias yang sudah dihitung pada proses sebelumnya, dan hasil akhir nanti berupa tanda dari nilai yang dihasilkan oleh fungsi *f(x).*

Perhitungan nilai *f(x)* dengan syarat jika pada level satu bernilai positif maka akan masuk kelas +1 dan masuk kelas B atau kelas penerima beasiswa layak, dan jika bernilai negatif maka masuk kelas -1 atau kelas tidak layak, apabila pada level

Adapun data *testing* yang digunakan dalam perhitungan nilai *f(x)* sebanyak 3 data *testing,* Yng terdiri dari 2 data testing kelas layak dan 1 data *testing* kelas tidak layak, data *testing* tersebut ditunjukkan pada Tabel 4.34.

Tabel 4.34 Penilaian Alternatif atau Normalisasi Untuk Data Testing Level 2

Ai	C1	C2	C3	C4	C5	....	C11	C12	C13	KELAS
A64	0,550	0,100	0,775	0,550	0,400	....	0,400	1,000	0,100	B
A65	0,775	0,100	0,550	1,000	1,000	....	0,400	1,000	0,100	B
A99	0,775	0,100	0,775	0,550	0,700	....	0,400	1,000	0,100	C

Tabel 4.34 merupakan data *testing* yang akan digunakan dalam perhitungan nilai *f(x).*Adapun perhitungan nilai *f(x),* dengan 3 data uji, yaitu 2 data uji untuk kelas layak dan 1 data uji untuk kelas tidak layak, sebagai berikut :

a. Mencari Nilai *f(x)* data uji 1 kelas B level 2

Perhitungan kernel $K(x, x_i)$ dilakukan pada data *testing* A64 dengan kernel *Polynomial Degree 2.* Dari persamaan kernel tersebut maka berikut perhitungan kernel yang dilakukan pada data *testing* A64, sebagai berikut :

$$\begin{aligned} K(x, x_i) = \; & [(0{,}775 \text{ x } 0{,}550) + (0{,}100 \text{ x } 0{,}100) + (0{,}325 \text{ x } 0{,}775) \\ & + (1{,}000 \text{ x } 0{,}550) + (1{,}000 \text{ x } 0{,}400) \\ & + (0{,}100 \text{ x } 0{,}100) + (0{,}100 \text{ x } 0{,}100) \\ & + (0{,}100 \text{ x } 0{,}100) + (0{,}100 \text{ x } 0{,}100) \\ & + (0{,}700 \text{ x } 1{,}000) + (0{,}400 \text{ x } 0{,}400) \\ & + (1{,}000 \text{ x } 1{,}000) + (0{,}100 \text{ x } 0{,}100)]^2 \end{aligned}$$

= 12,58919102

Perhitungan kernel untuk data *testing* seperti diatas, dan lakukan terhadap semua data *testing* sehingga menghasilkan nilai $K(x, x_i)$ pada Tabel 4.35

Tabel 4.35 Nilai Kernel Terhadap Masing-masing Data *Training* Level 2

SV	k(xi,x)

A73	12,58919102
A74	14,938225
A75	11,72634414
A102	10,10445156
A103	9,494101563
A104	11,88094727

Tabel 4.35 merupakan hasil dari perhitungan kernel terhadap masing-masing data *training.* Apabila sudah melakukan perhitungan tersebut maka tahap selanjutnya menghitung jumlah bobot pada persamaan untuk data *testing* A64 sebagai berikut :

$$\alpha_i y_i K(x, x_i) = 0{,}000668505 \text{ x } 1 \text{ x } 12{,}58919102 = 0{,}008415938$$

Dari perhitungan diatas maka didapatkan nilai bobot A73 dengan nilai 0,008415938, dan lakukan perhitungan terhadap data yang lainnya, sehingga menghasilkan hasil perhitungan pada Tabel 4.36.

Tabel 4.36 Nilai Bobot Data Terhadap Masing-masing Data *Training* Level 2

ID	αiyiK(xi,x)
A73	0,008415938
A74	0,009978731
A75	0,007842573
A102	-0,006785351
A103	-0,006364141
A104	-0,007974907

Apabila sudah mendapatkan hasil nilai bobot maka perhitungan selanjutnya menghitung nilai *f(x)* dengan persamaan 2.12 sebagai berikut:

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

$$f(x) = sign((0{,}005112843) + (-0{,}008002197)\,)$$

$$f(x) = sign(-0{,}002889355) = -1$$

Variabel $\sum_{i=1}^{n} \alpha_i\, y_i K(x, x_i)$ merupakan jumlah dari Tabel 4.36 yang merupakan jumlah nilai bobot dari semua data yang bernilai $-0{,}002889355$ sehingga nilai *f(x)* bernilai $sign(\sum_{i=1}^{n} \alpha_i\, y_i K(x, x_i) + b) = -1$ kelas penerima tidak layak.

b. Mencari Nilai *f(x)* data uji 2 kelas B level 2

Perhitungan kernel $K(x, x_i)$ dilakukan pada data *testing* A65 dengan kernel *Polynomial Degree 2.* Dari persamaan kernel tersebut maka berikut perhitungan kernel yang dilakukan pada data *testing* A65, sebagai berikut :

$$K(x, x_i) = [(0{,}775 \text{ x } 0775) + (0{,}100 \text{ x } 0{,}100) + (0{,}325 \text{ x } 0{,}550) + (1{,}000 \text{ x } 1{,}000) + (1{,}000 \text{ x } 1{,}000) + (0{,}100 \text{ x } 0{,}100) + (0{,}100 \text{ x } 0{,}100) + (0{,}100 \text{ x } 0{,}100) + (0{,}100 \text{ x } 0{,}100) + (0{,}700 \text{ x } 1{,}000) + (0{,}400 \text{ x } 0{,}400) + (1{,}000 \text{ x } 1{,}000) + (0{,}100 \text{ x } 0{,}100)]^2$$

= 22,08412539

Perhitungan kernel untuk data *testing* seperti diatas, dan lakukan terhadap semua data *testing* sehingga menghasilkan nilai $K(x, x_i)$ pada Tabel 4.37

Tabel 4.37 Nilai Kernel Terhadap Masing-masing Data *Training* Level 2

SV	k(xi,x)
A73	22,08412539
A74	21,94336914
A75	18,879025
A102	15,18563477
A103	10,660225
A104	17,58229727

Tabel 4.37 merupakan hasil dari perhitungan kernel terhadap masing-masing data *training.* Apabila sudah melakukan perhitungan tersebut maka tahap selanjutnya menghitung jumlah bobot pada persamaan untuk data *testing* A65 sebagai berikut :

$$\alpha_i y_i K(x, x_i) = 0{,}000668505 \text{ x } 1 \text{ x } 22{,}08412539 = 0{,}01476335$$

Dari perhitungan diatas maka didapatkan nilai bobot A73 dengan nilai 0,01476335, dan lakukan perhitungan terhadap data yang lainnya, sehingga menghasilkan hasil perhitungan pada Tabel 4.38.

Tabel 4.38 Nilai Bobot Data Terhadap Masing-masing Data *Training* Level 2

ID	αiyiK(xi,x)
A73	0,01476335
A74	0,014658165
A75	0,012626283
A102	-0,010197472
A103	-0,007145824
A104	-0,011801853

Apabila sudah mendapatkan hasil nilai bobot maka perhitungan selanjutnya menghitung nilai *f(x)* dengan persamaan 2.12 sebagai berikut:

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

$$f(x) = sign((0{,}01290265) + (-0{,}008002197)\,)$$

$$f(x) = sign(0{,}00490045) = 1$$

Variabel $\sum_{i=1}^{n} \alpha_i\, y_i K(x, x_i)$ merupakan jumlah dari Tabel 4.38 yang merupakan jumlah nilai bobot dari semua data yang bernilai 0,00490045 sehingga nilai *f(x)* bernilai $sign(\sum_{i=1}^{n} \alpha_i\, y_i K(x, x_i) + b) = -1$ kelas penerima tidak layak.

c. Mencari Nilai *f*(x) data uji 1 kelas C level 2

Perhitungan kernel $K(x, x_i)$ dilakukan pada data *testing* A99 dengan kernel *Polynomial Degree 2.* Dari persamaan kernel tersebut maka berikut perhitungan kernel yang dilakukan pada data *testing* A99, sebagai berikut :

$$\begin{aligned} K(x, x_i) = \; & [(0{,}775 \text{ x } 0775) + (0{,}100 \text{ x } 0{,}100) + (0{,}325 \text{ x } 0{,}775) \\ & + (1{,}000 \text{ x } 0{,}550) + (1{,}000 \text{ x } 0{,}700) + (0{,}100 \text{ x } 0{,}100) \\ & + (0{,}100 \text{ x } 0{,}100) + (0{,}100 \text{ x } 0{,}100) + (0{,}100 \text{ x } 0{,}100) \\ & + (0{,}700 \text{ x } 1{,}000) + (0{,}400 \text{ x } 0{,}400) + (1{,}000 \text{ x } 1{,}000) \\ & + (0{,}100 \text{ x } 0{,}100)]^2 \end{aligned}$$

= 16,18050625

Perhitungan kernel untuk data *testing* seperti diatas, dan lakukan terhadap semua data *testing* sehingga menghasilkan nilai $K(x, x_i)$ pada Tabel 4.39

Tabel 4.39 Nilai Kernel Terhadap Masing-masing Data *Training* Level 2

SV	k(xi,x)
A73	16,18050625
A74	17,20694102
A75	14,12350352
A102	11,98457852
A103	10,39256406
A104	14,5924

Tabel 4.39 merupakan hasil dari perhitungan kernel terhadap masing-masing data *training.* Apabila sudah melakukan perhitungan tersebut maka tahap selanjutnya menghitung jumlah bobot pada persamaan untuk data *testing* A99 sebagai berikut :

$$\alpha_i y_i K(x, x_i) = 0{,}000668505 \text{ x } 1 \text{ x } 16{,}18050625 = 0{,}010816751$$

Dari perhitungan diatas maka didapatkan nilai bobot A73 dengan nilai 0,010816751 , dan lakukan perhitungan terhadap data yang lainnya, sehingga menghasilkan hasil perhitungan pada Tabel 4.40.

Tabel 4.40 Nilai Bobot Data Terhadap Masing-masing Data *Training* Level 2

ID	αiyiK(xi,x)
A73	0,010816751
A74	0,011494233
A75	0,009445792
A102	-0,008047896
A103	-0,006966404
A104	-0,009794929

Apabila sudah mendapatkan hasil nilai bobot maka perhitungan selanjutnya menghitung nilai *f(x)* untuk mendapatkan nilai *f(x)* dengan menggunakan persamaan 2.12 sebagai berikut:

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

$$f(x) = sign((0{,}006947547) + (-0{,}008002197))$$

$$f(x) = sign(-0{,}0010547) = -1$$

Variabel $\sum_{i=1}^{n} \alpha_i\, y_i K(x, x_i)$ merupakan jumlah dari Tabel 4.38 yang merupakan jumlah nilai bobot dari semua data yang bernilai $-0{,}0010547$ sehingga nilai *f(x)* bernilai $sign(\sum_{i=1}^{n} \alpha_i\, y_i K(x, x_i) + b) = -1$ kelas penerima tidak layak.

Sehingga menghasilkan beberapa perhitungan *f(X),* yang bisa dilihat pada Tabel 4.41 untuk penerima kelas sangat layak, layak dan tidak layak.

Tabel 4.41 Data Hasil Perhitungan *f(X)* dan Klasifikasi

ID	f (x) level 1	f(x) level 2	HASIL KLASIFIKASI	ACTUAL CLASS
A14	0,00003355347	-	1 DILEVEL 1 = A	A
A32	0,00002926982	-	1 DILEVEL 1= A	A
A64	-0,00003197642	-0,00511948150	-1 DILEVEL 2 = C	B
A65	-0,00005365596	-0,02203417690	-1 DILEVEL 2 = C	B
A99	-0,00003356100	-0,00328477736	-1 DILEVEL 2 = C	C

Tabel 4.41 merupakan hasil dari perhitungan nilai *f(x)* untuk masing-masing level 1 dan level 2.

4.2.3.2 Perhitungan dengan Metode TOPSIS

Perhitungan dengan metode TOPSIS ini menjelaskan cara perhitungan untuk perangkingan dengan menggunakan bobot yang telah ditentukan, perhitungan dengan metode TOPSIS dilakukan setelah tahap klasifikasi dengan mteode SVM atau metode klasifikasi sudah didapatkan kelas yang diinginkan, yaitu kelas kelas bernilai positif +1 pada perhitungan dilevel satu, yaitu kelas sangat layak dan layak bagi penerima beasiswa GNOTA Kediri.

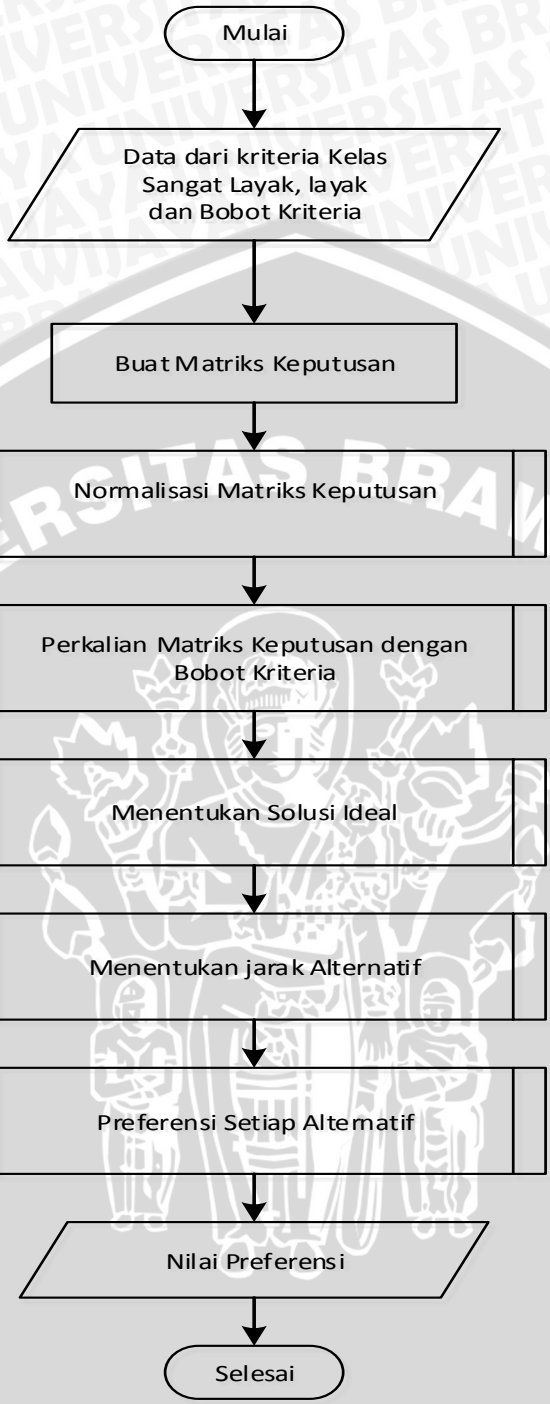
Penerima beasiswa yang mana nantinya apabila kuota yang disediakan sebanyak 10, namun kelas penerima sangat layak hanya menempati urutan sampai 6, kemudian sisa dari kuota akan disisi oleh penerima beasiswa dengan kelas layak sampai kuota terisi penuh sebanyak 10 calon mahasiswa penerima beasiswa GNOTA kediri tahun 2015.

Pembobotan nantinya yang ada pada metode topsis ini berasal dari pakar yaitu hasil wawancara langsung yang didapatkan oleh GNOTA Kediri. Adapun Untuk subsistem manajemen model pada sistem seleksi dan rekomendasi penerima beasiswa dengan perhitungan metode TOPSIS sebagai berikut:

1. Menentukan bobot penilai.
2. Membuat normalisasi matriks.
3. Membuat pembobotan pada matriks yang telah dinormalisasi.
4. Menentukan nilai solusi ideal positif dan solusi ideal negatif.
5. Menghitung *separation measure*. *Separation measure* ini merupakan pengukuran jarak dari suatu alternative ke solusi ideal positif dan solusi ideal negatif.
6. Menghitung nilai preferensi untuk setiap alternatif.

Manajemen model sistem seleksi dan rekomendasi penerima beasiswa dari perhitungan metode TOPSIS diatas merupakan tahapan-tahapan yang akan dilakukan untuk perhitungan TOPSIS, yaitu mulai dari menentukan bobot penilai, normalisasi matriks serta sampai dengan tahap akhir menghitung nilai prefensi untuk setiap nilai alternatif, yang mana nilai prefensi tersebut akan menentukan pengaruh peringkat penerima beasiswa pada GNOTA kediri nantinya.

Adapun contoh perhitungan metode TOPSIS ini dilakukan pada tahap proses akhir, yang mana akan menghasilkan perangkingan pada kelas sangat layak dan kelas layak penerima beasiswa GNOTA Kediri tahun 2015, dan tahapan berupa diagram alir ditunjukkan pada Gambar 4.12 sebagai berikut:



Gambar 4.12 Diagram Alir Perhitungan Metode TOPSIS

Gambar 4.12 merupakan diagram alir dari langkah-langkah perhitungan metode TOPSIS yang menjelaskan bahwa data pada saat perhitungan SVM yang bernilai 1+ pada level satu akan digunakan datanya untuk perhitungan TOPSIS, kemudian bobot pada masing-masing kriteria akan diinputkan sesuai bobot yang ditentukan oleh pakar, bobot dari pakar pada masing-masing kriteria berbeda-beda dengan jumlah keseluruhan dari bobot yaitu 1.

Bobot diinputkan akan dibuat perhitungan matriks keputusan, matriks keputusan tersebut dilakukan proses normalisasi. hasil dari matriks keputusan yang dinormalisasi akan dilakukan proses perkalian dengan bobot nilai yang sudah ditentukan oleh pakar.

Hasil dari proses perkalian tersebut yaitu matriks yang menentukan solusi ideal positif dan solusi ideal negatif, untuk mencari solusi ideal positif dengan cara mencari nilai maksimum dari setiap kriteria sedangkan untuk mencari solusi ideal negatif dengan cara mencari nilai minimum dari setiap kriterianya.

Apabila sudah mencari solusi ideal positif maupun solusi ideal negatif hasilnya akan digunakan untuk menentukan nilai preferensinya, nilai prefensi tersebut merupakan proses akhir untuk memberikan hasil dari perangkingan.

Langkah1 Menentukan bobot penilaian

Menentukan bobot penilaian disini datanya didapatkan dari pakar yang dilakukan dengan wawancara langsung kepada pihak GNOTA Kediri sendiri, dan untuk data yang digunakan merupakan data dari perhitungan dengan metode SVM yang memiliki nilai +1 pada level 1 dan nilai +1 pada level 2 yang ada pada data uji, sehingga akan dilanjutkan di metode TOPSIS yang akan dicari bobot penilaiannya, data tersebut dapat ditunjukkan pada Tabel 4.42.

Tabel 4.42 Data Testing Hasil Perhitungan dengan Metode SVM

Ai	C1	C2	C3	C4	C5	....	C11	C12	C13	KELAS
A14	0,775	0,100	0,775	1,000	1,000	....	0,400	1,000	1,000	A
A32	1,000	0,100	0,775	1,000	0,400	....	0,700	1,000	1,000	A
A65	0,775	0,100	0,550	1,000	1,000	....	0,400	1,000	0,100	B

Bobot yang diberikan atau didapatkan oleh pakar dari Lembaga GNOTA Kediri ditunjukkan pada Tabel 4.43.

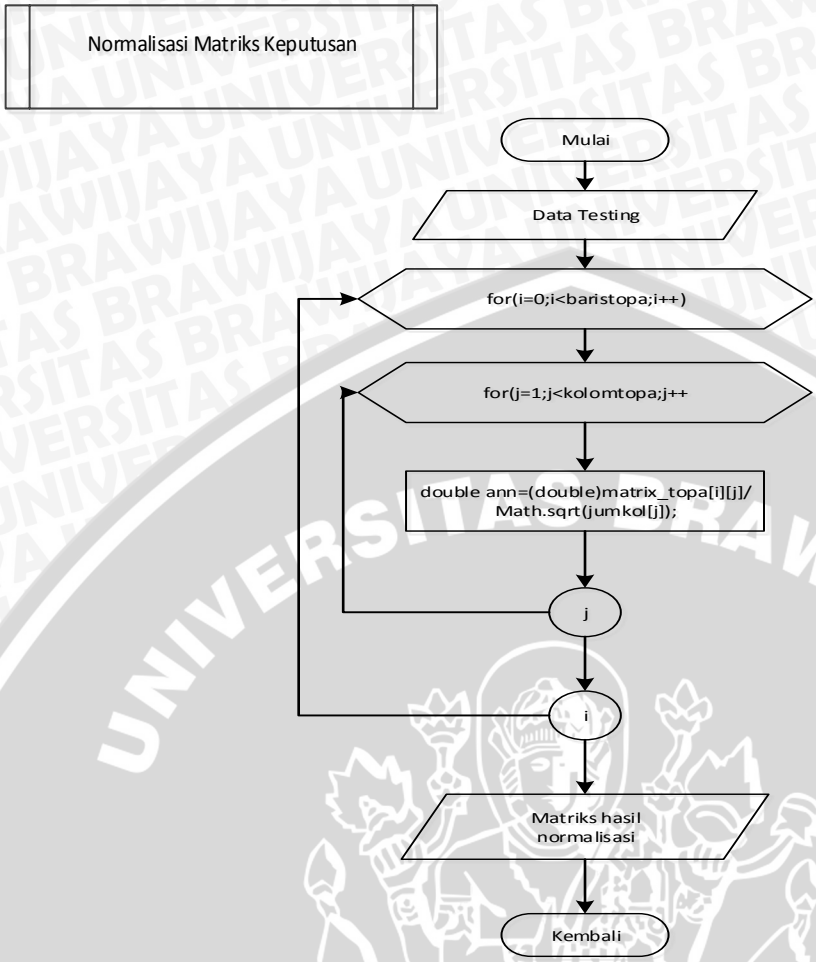
Tabel 4.43 Bobot Kriteria

W	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	JML
BOBOT	0,15	0,15	0,05	0,1	0,1	0,05	0,05	0,05	0,05	0,1	0,05	0,05	0,05	1

Langkah 2 Membuat normalisasi matriks

Membuat normalisasi martiks atau variabel *R* merupakan matriks keputusan yang ternormalisasi, yang tahapan-tahapan perhitungan membuat normalisasi martiks digambar dalam bentuk diagram alir, ditunjukkan pada Gambar 4.13.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA



Gambar 4.13 Diagram Alir Membuat Normalisasi Matriks

Gambar 4.13 merupakan tahapan perhitungan normalisasi matriks yang digambar dalam diagram alir, sehingga adapun perhitungan pada persamaan 2.24.

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^{m} x_{ij^2}}}$$

$$r_{11} = \frac{0{,}0775}{\sqrt{(0{,}775^2) + (1{,}000^2)}}$$

$$r_{11} = 0{,}6126$$

Hasil perhitungan diatas merupakan nilai dari normalisasi matriks dan lakukan pada semua data untuk kelas A, dan lakukan juga pada kelas B sehingga mengasilkan data matriks ternormalisasi pada Tabel 4.44.

Tabel 4.44 Matriks Normalisasi

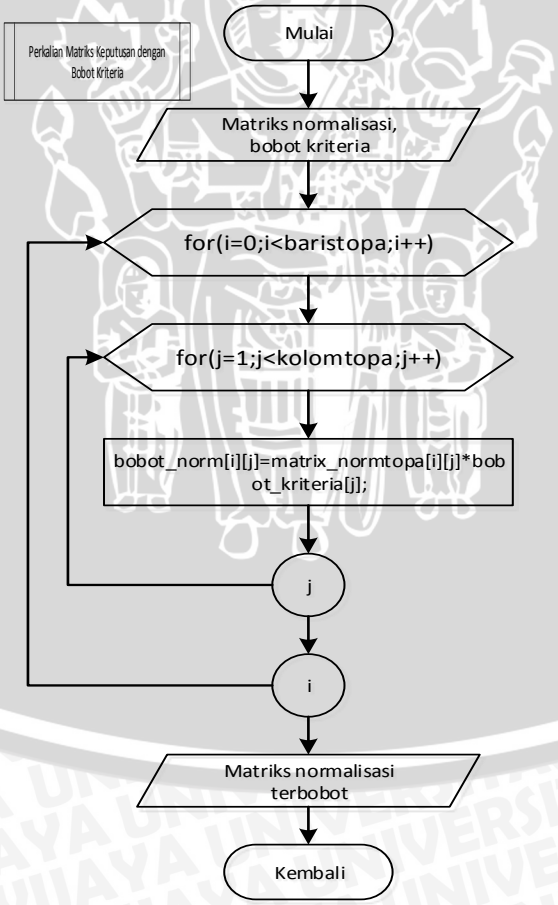
r_{ij}	C1	C2	C3	C4	C5	....	C11	C12	C13
KELAS A									
A14	0,6126	0,7071	0,7071	0,7071	0,9285	....	0,4961	0,7071	0,7071

A32	0,7904	0,7071	0,7071	0,7071	0,3714	**....**	0,8682	0,7071	0,7071
KELAS B									
A65	1,0000	1,0000	1,0000	1,0000	1,0000	**....**	1,0000	1,0000	1,0000

Tabel 4.44 merupakan hasil dari perhitungan matriks normalisasi dari semua data yang termasuk dalam kelas sangat layak dan kelas layak dari kelas A maupun kelas B, yang mana kelas layak hanya memiliki satu data saja yang dilakukan proses TOPSIS, sehingga akan dilanjutkan untuk perhitungan selanjutnya.

Langkah 3 Membuat pembobotan pada matriks yang telah dinormalisasi

Membuat pembobotan pada matriks yang telah dinormalisasi merupakan tahapan yang dilakukan setelah mencari nilai matriks normalisasi r_{ij}, langkah selanjutnya mengkalikan dengan bobot-bobot yaitu (w_j) untuk menghasilkan matriks yang telah dinormalisasikan yang dikalikan dengan bobot yang didapatkan dalam wawancara langsung oleh pakar. Sehingga adapun tahapan-tahapan dari proses awal samapai proses akhir dalam perhitungan pembobotan pada matriks yang telah dinormalisasi akan dijelaskan dalam bentuk diagram alir, berikut diagram alir ditunjukkan pada Gambar 4.14.



Gambar 4.14 Diagram Alir Perhitungan pembobotan pada matriks yang telah dinormalisasi

Gambar 4.14 merupakan tahapan-tahapan perhitungan pembobotan pada matriks yang telah dinormalisasi, sehingga dapat dilakukan perhitungan dengan persamaan 2.28, sebagai berikut:

$V = R_{ij}W_i$
$V = R_{11}W_1$
$V = 0{,}6126 \times 0{,}15$
$V = 0{,}0919$

Perhitungan diatas merupakan perhitungan variabel *R* yang merupakan matriks normalisasi A23 pada kriteria pertama (C1), dan variabel *W* merupakan bobot untuk krtiteria pertama (C1), sehingga menghasilkan perhitungan pada Tabel 4.45, sebagai berikut:

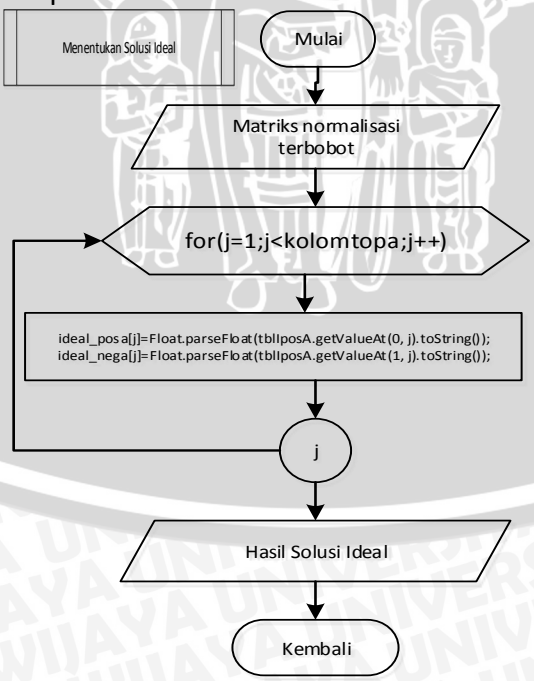
Tabel 4.45 Matriks pembobotan pada matriks yang telah dinormalisasi

V	C1	C2	C3	C4	C5	....	C11	C12	C13
KELAS A									
A14	0,0919	0,1061	0,0354	0,0707	0,0928	....	0,0248	0,0354	0,0354
A32	0,1186	0,1061	0,0354	0,0707	0,0371	....	0,0434	0,0354	0,0354
KELAS B									
A65	0,1500	0,1500	0,0500	0,1000	0,1000	....	0,0500	0,0500	0,0500

Tabel 4.45 merupakan hasil dari matriks semua perhitungan pembobotan pada matriks yang telah dinormalisasi.

Langkah 4 Menentukan nilai solusi ideal positif dan solusi ideal negatif

Menentukan Nilai solusi ideal positif dan negatif yang mana masing-masing solusi ideal dinotasikan dengan A^+dan A^-, adapun tahapan untuk menentukan solusi ideal ditunjukkan pada Gambar 4.15.



Gambar 4.15 Diagram Alir Menentukan nilai solusi ideal positif dan solusi ideal negatif

Gambar 4.15 merupakan tahapan perhitungan solusi ideal positif dan solusi ideal negatif, yang mana pada perhitungan solusi ideal positif menggunakan persamaan 2.29 dan 2.30 untuk persamaan solusi ideal negatif, dan berikut perhitungan untuk masing-masing solusi ideal:

$$A^{+} = \{(\max v_{ij} \,|j \in J), (\min v_{ij} \,|j \in J'), i = 1,2,3, \dots, m\}$$
$$= \{v_1^{+}, v_2^{+}, \dots, v_n^{+}\}$$

$$A^{+} = \{\max(0.0919, 0.1186)\}$$
$$= 0.1186$$

Perhitungan A^{+} diatas menggunakan contoh dari maksimum dari semua kolom kriteria pertama atau C1.

$$A^{-} = \{(\min v_{ij} \,|j \in J), (\max v_{ij} \,|j \in J'), i = 1,2,3, \dots, m\}$$
$$= \{v_1^{-}, v_2^{-}, \dots, v_n^{-}\}$$

$$A^{-} = \{\min(0.0919, 0.1186)\}$$
$$= 0.0919$$

Perhitungan A^{-} diatas menggunakan contoh dari min dari semua kriteria pertama atau C1. Sehingga dari perhitungan kedua solusi tersebut dilakukan juga terhadap semua kriteria yaitu sebanyak 13 kriteria, dan hasil perhitungannya untuk masing-masing solusi ideal ditunjukkan pada Tabel 4.46 dan Tabel 4.47.

Tabel 4.46 Solusi Ideal Positif

V	C1	C2	C3	C4	C5	....	C11	C12	C13
KELAS A									
A23	0,0919	0,1061	0,0354	0,0707	0,0928	....	0,0248	0,0354	0,0354
A24	0,1186	0,1061	0,0354	0,0707	0,0371	....	0,0434	0,0354	0,0354
A^{+}	**0,1186**	**0,1061**	**0,0354**	**0,0707**	**0,0928**		**0,0434**	**0,0354**	**0,0354**
KELAS B									
A65	0,1500	0,1500	0,0500	0,1000	0,1000	....	0,0500	0,0500	0,0500
A^{+}	**0,1500**	**0,1500**	**0,0500**	**0,1000**	**0,1000**	....	**0,0500**	**0,0500**	**0,0500**

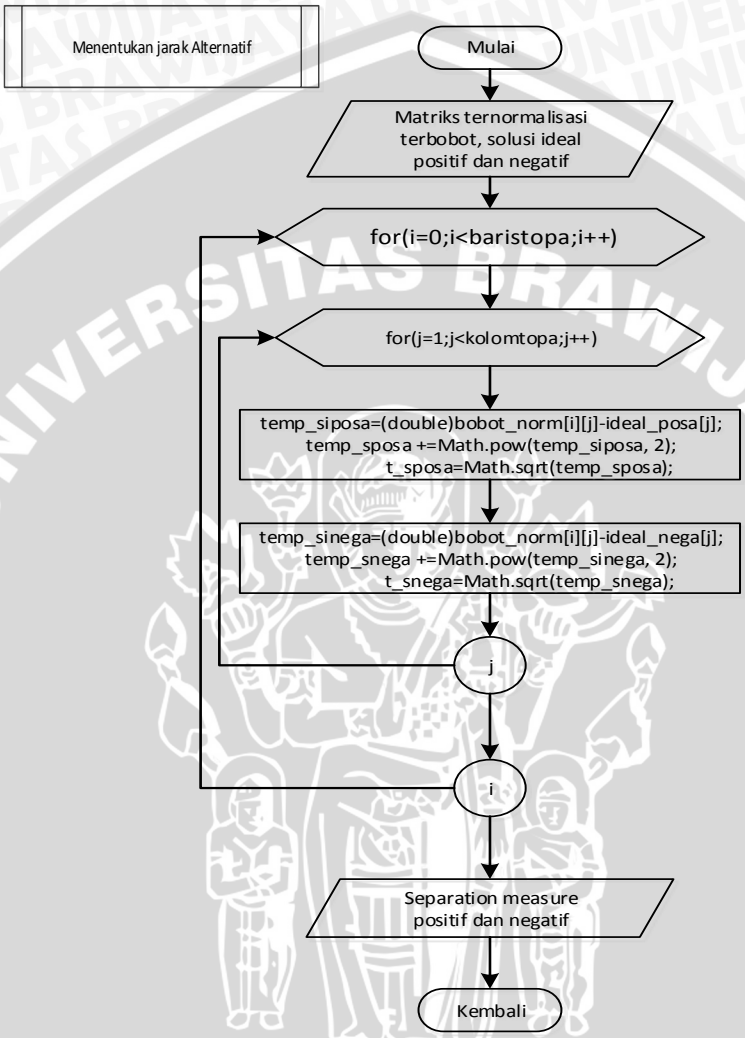
Tabel 4.47 Solusi Ideal Negatif

V	C1	C2	C3	C4	C5	....	C11	C12	C13
KELAS A									
A23	0,0919	0,1061	0,0354	0,0707	0,0928	....	0,0248	0,0354	0,0354
A24	0,1186	0,1061	0,0354	0,0707	0,0371	....	0,0434	0,0354	0,0354
A^{-}	**0,0919**	**0,1061**	**0,0354**	**0,0707**	**0,0371**		**0,0248**	**0,0354**	**0,0354**
KELAS B									
A65	0,1500	0,1500	0,0500	0,1000	0,1000	....	0,0500	0,0500	0,0500
A^{-}	**0,1500**	**0,1500**	**0,0500**	**0,1000**	**0,1000**	....	**0,0500**	**0,0500**	**0,0500**

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Langkah 5 Menghitung nilai *separation measure*

Menghitung nilai *separation measure* bertujuan untuk pengukuran jarak dari alternatif ke solusi ideal positif maupun solusi ideal negatif, adapun langkah-langkah perhitungan separation measure dijelaskan dalam diagram alir, yang ditunjukkan pada Gambar 4.16 sebagai berikut:



Gambar 4.16 Diagram Alir Menghitung Nilai Separation Measure

Gambar 4.16 merupakan langkah-langkah perhitungan *separation measure* dari langkah awal sampai mendapatkan nilai *separation measure,* adapun persamaan untuk menghitung *separation measure* positif dan negatif pada contoh data A23 menggunakan persamaan 2.31 dan persamaan 2.32, berikut perhitungan *separation measure* positif*:*

$$s_i^+ = \sqrt{\sum_{i=1}^{n} (v_{ij} - v_i^+)^2}$$

$$s_i^+ = \sqrt{\begin{array}{c}(0{,}0919-0{,}1186)^2+\ (0{,}1061-0{,}1061)^2+(0{,}0354-0{,}0354)^2\\ +(0{,}0707-0{,}0707)^2+(0{,}0928-0{,}0928)^2+(0{,}0354-0{,}0354)^2\\ +(0{,}0354-0{,}0354)^2+(0{,}0354-0{,}0354)^2+(0{,}0492-0{,}0492)^2\\ +(0{,}0707-0{,}0707)^2+(0{,}0248-0{,}0434)^2+(0{,}0354-0{,}0354)^2\\ +(0{,}0354-0{,}0354)^2\end{array}}$$

$s_i^+ = 0{,}0325$

Perhitungan *separation measure* positif didapatkan dengan hasil dari jumlah matriks pembobotan ternormalisasi dikurang dengan solusi ideal positif kemudian dilakukan penguadratan, sehingga lakukan juga perhitungan *separation measure* terhadap alternatif lainnya sehingga menghasilkan *separation measure* seperti pada Tabel 4.48 berikut:

Tabel 4.48 Separation Measure Positif

NAMA	s_i^+
KELAS A	
A23	0,0325
A24	0,0687
KELAS B	
A65	0,0000

Setelah perhitungan *separation measure* positif, selanjutnya menghitung *separation measure* negatif, berikut perhitungannya:

$$s_i^- = \sqrt{\sum_{i=1}^{n}(v_{ij}-\ v_i^-)^2}$$

$$s_i^- = \sqrt{\begin{array}{c}(0{,}0919-0{,}0919)^2+\ (0{,}1061-0{,}1061)^2+(0{,}0354-0{,}0354)^2\\ +(0{,}0707-0{,}0707)^2+(0{,}0928-0{,}0371)^2+(0{,}0354-0{,}0354)^2\\ +(0{,}0354-0{,}0354)^2+(0{,}0354-0{,}0354)^2+(0{,}0492-0{,}0089)^2\\ +(0{,}0707-0{,}0707)^2+(0{,}0248-0{,}0434)^2+(0{,}0354-0{,}0354)^2\\ +(0{,}0354-0{,}0354)^2\end{array}}$$

$s_i^- = 0{,}06873$

Perhitungan *separation measure* negatif didapatkan dengan hasil dari jumlah matriks pembobotan ternormalisasi dikurang dengan solusi ideal negatif kemudian dilakukan penguadratan, sehingga lakukan juga perhitungan *separation measure* terhadap alternatif lainnya sehingga menghasilkan *separation measure* seperti pada Tabel 4.49 berikut:

Tabel 4.49 Separation Measure Negatif

NAMA	s_i^-
KELAS A	
A23	0,06873
A24	0,03252
KELAS B	

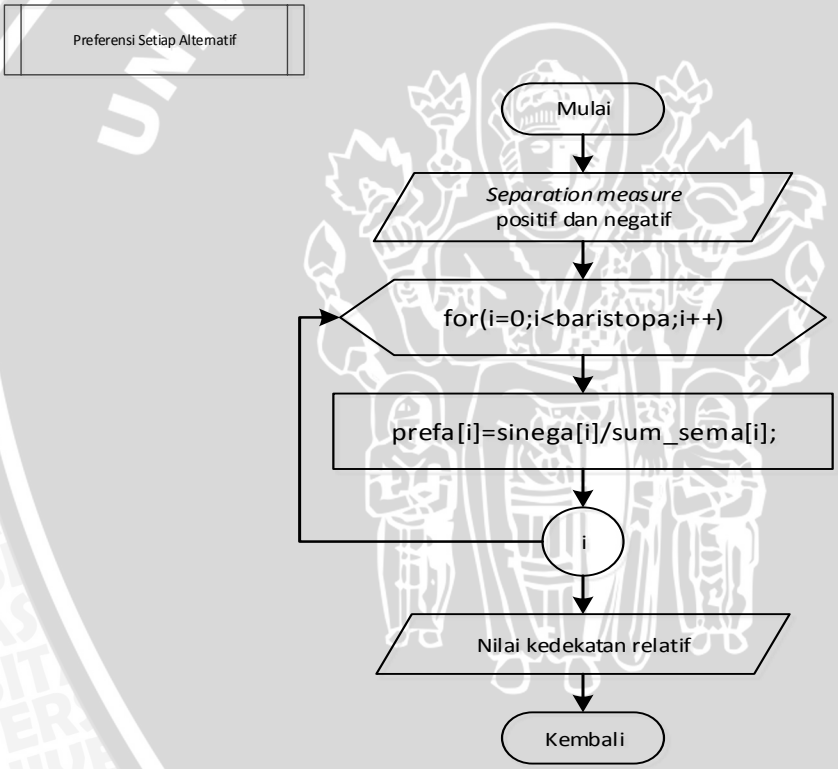
A65	0,0000

Tabel 4.48 dan Tabel 4.49 merupakan hasil dari perhitungan *separation measure* positif dan *separation measure* negatif, yang mana perhitungan *separation measure* akan dilanjutkan pada langkah atau perhitungan selanjutnya.

Langkah 6 Menghitung Nilai Preferensi

Menghitung nilai preferensi pada metode TOPSIS merupakan langkah terakhir yang menghasilkan peringkat dari semua data yang sangat layak, untuk menentukan rangking-rangking pada setiap alternatif nantinya maka perlu menghitung nilai preferensi pada setiap alternatif.

Tahapan-tahapan menghitung nilai preferensi akan dijelaskan dalam bentuk diagram alir, yang mana tahapan-tahapan tersebut menghasilkan perangkingan pada keseluruhan yang memiliki nilai preferensi yang tertinggi, adapun diagram alir untuk tahapan-tahapan perhitungan preferensi ditunjukkan pada Gambar 4.17 sebagai berikut:



Gambar 4.17 Diagram Alir Menghitung Nilai Preferensi

Gambar 4.17 merupakan tahapan-tahapan untuk menghitung nilai preferensi berupa diagram alir, sehingga untuk menghitung nilai preferensi tersebut dapat menggunakan persamaan 2.33, dan berikut perhitunganya pada alternatif A23.

$$C_{A23}^{+} = \frac{S_{A23}^{-}}{S_{A23}^{-} + S_{A23}^{+}}$$

$$C_{A23}^{+} = \frac{0,6873}{0,0325 + 0,6873}$$

$$C_{A23}^{+} = 0,6788$$

Perhitungan nilai preferensi diatas didapatkan dari hasil *separation measure* negatif dibagi dengan hasil jumlah dari *separation measure* negatif dan *separation measure* positif, dan lakukan perhitungan nilai preferensi terhadap semua alternatif, sehingga menghasilkan nilai preferensi seperti pada Tabel 4.50.

Tabel 4.50 Nilai Preferensi

NAMA	PREFERENSI	RANKING
KELAS A		
A23	0,6788	1
A24	0,3212	2
KELAS B		
A65	0,0000	1

Tabel 4.49 merupakan hasil dari perhitungan nilai preferensi terhadap semua data atau alternatif, yang mana dari nilai $\boldsymbol{C_i^+}$ atau jarak kedekatan alternatif terhadap solusi ideal maka diperoleh nilai $\boldsymbol{C_{A23}^+}$ untuk kelas A memiliki peringkat 1 dari nilai preferensi tertinggi sedangkan untuk kelas B yang memiliki nilai preferensi tertinggi yaitu $\boldsymbol{C_{A65}^+}$, sehingga menjadi peringkat pertama untuk kelas B.

Jadi untuk tampilan akhir jika jumlah kuota penerima beasiswa sebanyak 5 orang maka yang menerima beasiswa terdiri dari semua kelas A dan B, dikarenakan total dari jumlah data A hanya sebanyak dua buah data, dan masih tersedia tiga buah data, maka kelas B akan mendapatkan beasiswa juga, sehingga masih ada untuk data lain yang mana jika data mengalami pertambahan dan kuota masih ada.

4.2.3.3 Perhitungan Evaluasi

Perhitungan evaluasi untuk mengukur performansi dan mengukur tingkat keberhasilan untuk membangdingkan hasil dari proses implementasi yangtelah dilakukan degan beberapa aturan yang telah dibuat untuk proses evaluasi untuk menentukan akurasi dari sistem dengan metode SVM.

Dari proses evaluasi atau pengujian menggunakan salah satu model evaluasi yaitu *confusion matriks* yang mana dari hasil perhitungan nilai *f(x),* maka didapatkan tabel *confusion matriks* nya seperti yang ditunjukkan pada Tabel 4.51.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 4.51 Model Evaluasi *Confusion* Matriks

ACTUAL CLASS	*PREDICTED CLASS*			JUMLAH
	Sangat Layak	Layak	Tidak Layak	
Sangat Layak	2	0	0	2
Layak	0	1	1	2
Tidak Layak	0	0	1	1

Adapun untuk menghitung nilai dari akurasi sistem yang dibuat menggunakan persamaan 2.34 , yaitu sebagai berikut.

$$Classification\ Accuration = \frac{TP + TN}{TP + FP + FN + TN}$$

Dari persamaan diatas maka nilai akurasi yang didapatkan yaitu :

a. TP = (C1,1)+(C2,2)+(C3,3) = 2+1+1 = 4

b. TN = (C1,1 + C2,2)+(C1,1 + C3,3)+(C2,2 + C3,3) = (2+1) + (2+1) + (1+1) = 8

c. FP = (C2,1 + C3,1)+(C1,2+C3,2)+(C1,3 + C2,3) = (0+0) +(0+0) +(0+2) = 2

d. FN = (C1,2 + C1,3)+(C2,1 + C2,3)+(C3,1+ C3,2) = (0+0)+(0+2)+(0+0) = 2

$$Classification\ Accuration = \frac{4 + 8}{4 + 2 + 2 + 8}$$

$$Classification\ Accuration = \frac{12}{16}$$

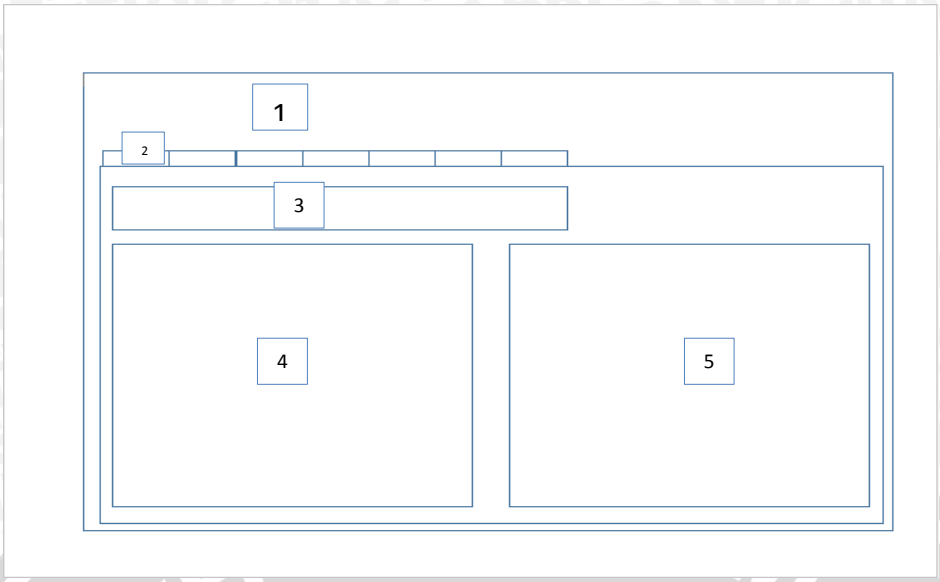
$$Classification\ Accuration = 86\%$$

Sehingga dari perhitungan tersebut didapatkan nilai akurasi pada sistem dengan menggunakan *confusion matriks* sebesar 86%.

4.2.4 Subsistem Antarmuka

Subsistem antarmuka atau *interface* disini bertujuan untuk mempermudah setiap *user* dapat berkomunikasi dan menggunakan sistem ini dengan baik maupun cara berinteraksinya, pada perancangan antarmuka ini akan menjelaskan *interface* yang akan dibuat nantinya seperti tombol menu dan adapun perancangan *interface* terdiri dari halaman *login* dan halaman utama dari sistem Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri dengan Implementasi Metode SVM dan TOPSIS, perancangan antarmuka untuk halaman *login* dapat dilihat pada Gambar 4.18.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA



Gambar 4.18 Perancangan Antarmuka Sistem

Gambar 4.18 merupakan perancangan antarmuka dari sistem yang akan dibuat nantinya yang mana sistem nantinya diimplementasikan pada NetBeans versi 7.3.

Keterangan :

1 = Halaman antarmuka sistem seleksi dan beasiswa dengan metode SVM dan TOPSIS.

2 = Tab menu sistem yang terdiri dari load data, normalisasi, poly n hessian, iterasi level 1, mencari nilai *f(x),* level 2, 2.normalisasi, 2.poly n hessian, iterasi level 2, hasil perhitungan *f(x)*, dan TOPSIS.

3 = Pemilihan proses perhitungan yang terdiri dari manual, random dan load data.

4 = Tabel dari data latih.

5 = Tabel dari data uji.

4.2.5 Subsistem Skenario Pengujian

Subsistem skenario pengujian menjelaskan beberapa pengujian yang akan dilakukan terhadap sistem Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri dengan Implementasi Metode SVM dan TOPSIS, pengujian yang akan dilakukan nantinya terdiri dari beberapa pengujian, adapun pengujian terhadap sistem yaitu:

1. Pengujian untuk mengetahui perbandingan jumlah data *training* dan data *testing* terhadap hasil akurasi.
2. Pengujian untuk mengetahui waktu eksekusi pada sistem selama proses perhitungan.
3. Pengujian terhadap tingkat akurasi dari pengaruh parameter *lambda*.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

4. Pengujian untuk mengetahuipengaruh nilai konstanta *gamma* terhadap nilai akurasi.
5. Pengujian tingkat akurasi terhadap pengaruh nilai *epsilon*.
6. Pengujian tingkat akurasi terhadap pengaruh nilai ietrasi maksimum.
7. Pengujian untuk mengetahui pengaruh nilai *C* terhadap akurasi.

Skenario diatas terdiri dari enam skenario pengujian yang nantinya akan digunakan unutk mengevaluasi hasil perhitungan sistem Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri dengan Implementasi Metode SVM dan TOPSIS.

4.2.5.1 Pengujian untuk mengetahui perbandingan jumlah data *training* dan data *testing* terhadap hasil akurasi.

Pengujian untuk mengetahui perbandingan jumlah data *training* dan data *testing* terhadap hasil akurasi, sehingga diketahui apakah banyaknya data *training* maupun banyaknya data *testing* dalam perbandingan dapat mempengaruhi terhadap nilai akurasi, adapun rancangan untuk mengetahui perbandingan jumlah data *training* dan data *testing* terhadap hasil akurasi ditunjukkanTabel 4.52.

Tabel 4.52 Skenario Pengujian Rasio Perbandingan

Rasio (%)	Percobaan Ke-*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						

Tabel 4.52 merupakan rasio perbandingan data yang akan digunakan pada saat proses pengujian.

4.2.5.2 Pengujian untuk mengetahui waktu eksekusi pada sistem selama proses perhitungan

Pengujian untuk mengetahui waktu eksekusi pada sistem selama proses perhitungan untuk mengetahui waktu eksekusi terhadap perbandingan data *training* dan data *testing*, sehingga akan mengetahui berapa lama sistem memproses dari proses seleksi tersebut. Adapun skenario pengujian waktu terdapat pada Tabel 4.53.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 4.53 Skenario Pengujian Waktu Eksekusi

Rasio (%)	Percobaan Ke-*i*					Rata-rata (detik)
	1	2	3	4	5	
90 : 10						
80 : 20						
70 : 30						
60 : 40						
50 : 50						
40 : 60						
30 : 70						
20 : 80						
10 : 90						

Tabel 4.53 merupakan skenario pengujian untuk mengetahui waktu proses sistem dalam melakukan seleksi.

4.2.5.3 Pengujian terhadap tingkat akurasi dari pengaruh nilai parameter *lambda*

Pengujian terhadap tingkat akurasi dari pengaruh nilai parameter *lambda* yang terbaik untuk mengetahui tingkat akurasi yang dihasilkan. Nilai *lambda* yang akan diuji dijelaskan pada Tabel 4.54.

Tabel 4.54 Skenario Pengujian Parameter Lambda

Lambda	Percobaan Ke-*i*					Rata-rata
	1	2	3	4	5	
0.1						
0.3						
0.5						
1						
5						
7						
10						
20						
30						
40						

Tabel 4.54 merupakan nilai lambda yang akan diuji nantinya dan pengujian dilakukan sebanyak lima kali.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

4.2.5.4 Pengujian untuk mengetahui pengaruh nilai konstanta *gamma* terhadap nilai akurasi.

Pengujian pengaruh terhadap tingkat akurasi dari pengaruh nilai konstanta gamma yang terbaik untuk mengetahui tingkat akurasi yang dihasilkan. Nilai konstanta gamma yang akan diuji dijelaskan pada Tabel 4.55.

Tabel 4.55 Skenario Pengujian Nilai Konstanta Gamma

Konstanta Gamma	Percobaan Ke-*i*					Rata-rata
	1	2	3	4	5	
0,00001						
0,0001						
0,00005						
0,001						
0,005						
0,01						
0,05						
0,1						
0,5						
1						

Tabel 4.55 merupakan skenario pengujian nilai konstanta gamma untuk mengetahui tingkat akurasi.

4.2.5.5 Pengujian tingkat akurasi terhadap nilai *epsilon*

Pengujian pengaruh terhadap tingkat akurasi dari pengaruh nilai *epsilon* yang terbaik untuk mengetahui tingkat akurasi yang dihasilkan. Nilai *epsilon* yang akan diuji dijelaskan pada Tabel 4.56.

Tabel 4.56 Skenario Pengujian nilai *epsilon*

Epsilon	Percobaan Ke-*i*					Rata-rata
	1	2	3	4	5	
0,000000001						
0,00000001						
0,0000001						
0,000001						
0,00001						
0,0001						
0,001						
0,01						
0,1						
1						

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 4.56 merupakan skenario pengujian untuk mengetahui pengaruh nilai *epsilon* terhadap akurasi.

4.2.5.6 Pengujian tingkat akurasi terhadap nilai iterasi maksimum

Pengujian terhadap tingkat akurasi dari pengaruh pengaruh nilai iterasi maksimum yang terbaik untuk mengetahui tingkat akurasi yang dihasilkan. Nilai nilai iterasi maksimum yang akan diuji dijelaskan pada Tabel 4.57.

Tabel 4.57 Pengujian Tingkat iterasi maksimum

Iterasi	Percobaan Ke-*i*					Rata-rata
	1	2	3	4	5	
100						
500						
1000						
2000						
3000						
5000						
10000						
25000						
50000						
100000						

Tabel 4.57 merupakan jumlah nilai iterasi maksimum yang akan diuji nantinya dan pengujian dilakukan sebanyak lima kali.

4.2.5.7 Pengujian untuk mengetahui pengaruh nilai *C* terhadap akurasi

Pengujian pengaruh terhadap tingkat akurasi dari pengaruh nilai *C* yang terbaik untuk mengetahui tingkat akurasi yang dihasilkan. Nilai *C* yang akan diuji dijelaskan pada Tabel 4.58.

Tabel 4.58 Pengujian untuk mengetahui pengaruh nilai *C*

C	Percobaan Ke-*i*					Rata-rata
	1	2	3	4	5	
1						
10						
20						
30						
40						
50						
60						
70						

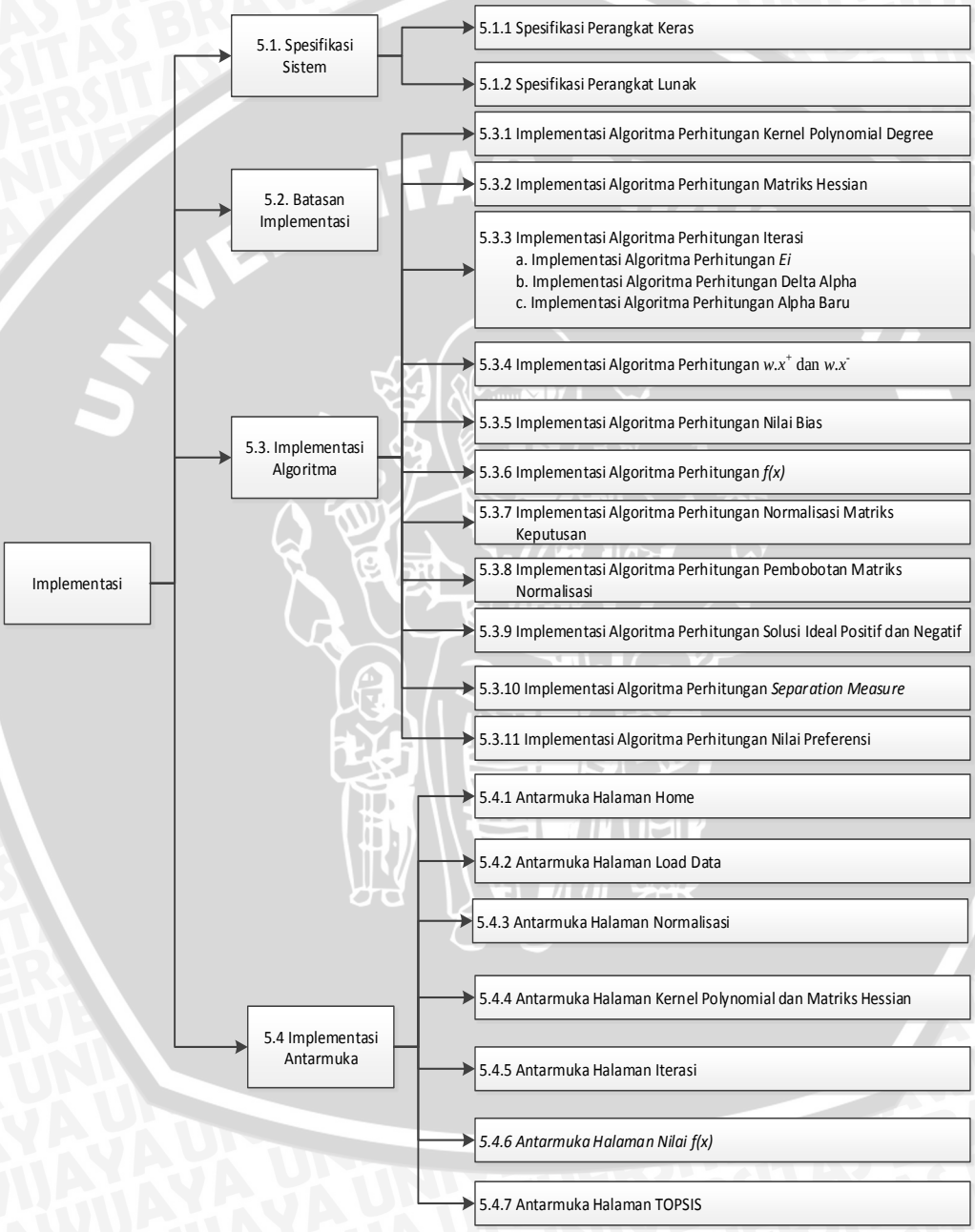
repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

80						
90						

Tabel 4.58 merupakan skenario dari pengujian nilai C, untuk mengetahui pengaruh nilai C terhadap pengaruh akurasi.

BAB 5 IMPLEMENTASI SISTEM

Bab ini akan menguraikan proses-proses dari Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri dengan Implementasi Metode SVM dan TOPSIS yang meliputi spesifikasi sistem, batasan implementasi, implementasi algoritma dan implementasi antarmuka, adapun aliran penulisan bab ini sebagai berikut.



Gambar 5.1 Diagram Implementasi Sistem

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Gambar 5.1 merupakan alur dari implementasi dari sistem Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri dengan Implementasi Metode SVM dan TOPSIS

5.1 Spesifikasi Sistem

Proses implementasi sistem membutuhkan spesifikasi yang sesuai dengan yang dibangun sehingga dapat berfungsi sesuai dengan kebutuhan. Spesifikasi perangkat keras maupun spesifikasi perangkat lunak yang dibutuhkan oleh sistem sebagai yang akan dijabarkan.

5.1.1 Spesifikasi Perangkat Keras

Pengembangan dari Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri dengan Implementasi Metode SVM dan TOPSIS menggunakan perangkat keras sesuai yang dibutuhkan, yang ditunjukkan pada Tabel 5.1.

Tabel 5.1 Spesifikasi Perangkat Keras

Komponen	Spesifikasi
Prosesor	*Intel(R)® Core™ i5-4200U CPU @ 1.60 GHz 2.30 GHz*
Memori	*4096 MB*
Kartu Grafis	*NVIDIA ® GEFORCE 840M*
Hard disk	*1 TB*

Tabel 5.1 merupakan spesifikasi perangkat keras yang dibutuhkan untuk mendukung Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri dengan Implementasi Metode SVM dan TOPSIS.

5.1.2 Spesifikasi Perangkat Lunak

Pengembangan dari Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri dengan Implementasi Metode SVM dan TOPSIS menggunakan perangkat lunak sesuai yang dibutuhkan, yang ditunjukkan pada Tabel 5.2.

Tabel 5.2 Spesifikasi Perangkat Lunak

Nama	Spesifikasi
Sistem operasi	*Microsoft Windows 8.1 Single Language 64-bit*
Bahasa pemrograman	*Java*
Tools Pemrograman	*NetBeans IDE 7.3*
Database	*MySQL*
Tools DBMS	*SQLyog Ultimate v9.02*

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 5.2 merupakan spesifikasi perangkat lunak yang dibutuhkan untuk mendukung Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri dengan Implementasi Metode SVM dan TOPSIS.

5.2 Batasan Implementasi

Beberapa batasan dalam membangun sistem Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri dengan Implementasi Metode SVM dan TOPSIS sebagai berikut:

1. Sistem dibangun aplikasi berbasis *dekstop* dengan menggunakan bahasa *java*.
2. Penyimpanan data dengan menggunakan DBMS MySQL.
3. Metode yang digunakan dalam menyelesaikan masalah adalah metode *Support Vector Machine* (SVM) dan metode TOPSIS.
4. Data masukan yang digunakan dalam sistem adalah data hasil seleksi pendaftaran beasiswa di lembaga GNOTA Kabupaten Kediri tahun 2015.
5. Hasil *Output* dari hasil perhitungan dengan metode SVM adalah menghasilkan klasifikasi penerima beasiswa sedangkan untuk metode TOPSIS *Output*-nya adalah perangkingan penerima beasiswa dari klasifikasi sangat layak dan layak.
6. Kriteria yang digunakan pada penelitian ini sejumlah 13 kriteria.
7. Hasil evaluasi atau akurasi hanya ada pada metode *Support Vector Machine* (SVM) dengan menggunakan evaluasi atau akurasi *Confusion Matriks.*

5.3 Implementasi Algoritma

Sistem Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri dengan Implementasi Metode SVM dan TOPSIS memiliki 13 algoritma yang ada pada sistem seleksi dan rekomendasi, yang mana total keseluruhan ada 13 algoritma, algoritma tersebut dijelaskan pada Gambar 5.1 pada Diagram Implementasi bagian implementasi algoritma, berikut penjelasan dari masing-masing algoritma yang sudah di jelaskan sebelumnya sehingga nanti pada implementasi dapat dicek satu-satu pada Kode Program.

5.3.1 Implementasi Algoritma Perhitungan Kernel *Polynomial Degree d*

Algortima dari perhitungan kernel *Polynomial Degree d* merupakan proses awal perhitungan dengan menggunakan metode SVM yang melibatkan *dot product* antar 2 data *training* pada ruang *vector* yang berdimensi.

Perhitungan Kernel yang digunakan yaitu *Polynomial Degree d* yang nantinya keluaran yang dihasilkan berupa matriks kernel dengan index $n \times n$,dan n yaitu banyaknya jumlah data *training.* Kode program dari algoritma kernel *Polynomial Degree d* dijelaskan pada Kode Program 5.1.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

```
1.  public void PolynomialDegree(){
    System.out.println("---------------POLYNOMIAL DEGREE 2-----
    -------");
2.          d=Float.parseFloat(txtd.getText());
3.          System.out.println("Nilai d :"+d);
4.          for(i=0;i<barislt;i++){
5.              for(j=0;j<barislt;j++){
6.                  t=0;
7.                  for(k=1;k<kolomlt;k++){
    t=t+(matrix_norm_lt[i][k]*matrix_norm_lt[j][k]);
8.                  }
9.                  sm[j]=t;
10.             }
11.             System.out.println("");
12.             for(l=0;l<barislt;l++){
13.                 temp_matrix_polde[l][i]=sm[l];
14.                 BigDecimal bd=new
    BigDecimal(temp_matrix_polde[l][i]);
15.     bd = bd.setScale(decimalPlace,BigDecimal.ROUND_UP);
16.                 temp_matrix_polde[l][i] = bd.floatValue();
17.                 q=(double)temp_matrix_polde[l][i];
18.                 ff=Math.pow(q, d);
19.               matrix_poly[i][l]=ff;
20.              System.out.print(" "+matrix_poly[i][l]+" ");
21.             }
            }
```

Kode Program 5.1 Implementasi Algoritma Perhitungan Kernel *Polynomial Degree d*

Berikut penjelasan dari Kode Program 5.1:

1. Baris 1 sampai baris 6 menjelaskan tentang perulangan untuk data *training* ke-*i* dan ke-*j*.
2. Baris 7 sampai baris 11 menjelaskan operasi perhitungan dari kernel *Polynomial Degree d.*
3. Baris 12 menjelaskan perulangan untuk menggunakan set perhitungan desimal.
4. Baris 13 sampai baris 18 menjelaskan untuk matriks polynomial untuk ditampilkan.
5. Baris 19 sampai baris 21 menjelaskan untuk hasil tampilan dari matriks polynomial.

5.3.2 Implementasi Algoritma Perhitungan Matrik Hessian

Proses perhitungan *matrix* hessian dilakukan untuk mencari nilai hessian yang nanti akan digunakan untuk perhitungan yang ada pada proses algoritma *Sequential Training* SVM. Variabel dari perhitungan matriks hessian didapatkan dari hasil perhitungan fungsi kernel *Polynomial Degree d* dan nilai dari konstanta *lambda*. Kode program dari algoritma perhitungan matriks hessian dijelaskan pada Kode Program 5.2.

```
1.  public void hitungHessian(){
2.     System.out.println("");
3.     System.out.println("---------------HITUNG MATRIKS HESSIAN-
    -----------");
4.          lambda=Float.parseFloat(txtLambda.getText());
5.          System.out.println("Nilai Lambda :"+lambda);
6.          konstanta=Float.parseFloat(txtKonstanta.getText());
7.          System.out.println("Nilai Konstanta : "+konstanta);
8.      for(i=0;i<barislt;i++){
9.  matrik_y[i]=Float.parseFloat(tblDataLatih.getValueAt(i,
    14).toString());
10.             System.out.println(" "+matrik_y[i]+" ");
11.         }
12.         System.out.println(" ");
13.         tempLambda=(double)lambda;
14.         tempLambda=Math.pow(tempLambda,po);
15.         lambdanew=(float)tempLambda;
16.         for(i=0;i<barislt;i++){
17.           for(l=0;l<barislt;l++){
18.             matrix_hessian[i][l]=matrik_y[j]*matrik_y[j];
19.              tempPoly=(float)matrik_poly[i][l];
20. matrix_hessian[i][l]=(matrik_y[i]*matrik_y[l])*(tempPoly+lam
    bdanew);
21.           System.out.print(" "+matrix_hessian[i][l]+" ");
22.             }
23.             System.out.println(" ");
```

Kode Program 5.2 Implementasi Algoritma Perhitungan Matrik *Hessian*

Berikut penjelasan dari Kode Program 5.2:

1. Baris 1 sampai baris 7 menjelaskan inputan beberapa parameter yang akan dilakukan pada proses perhitungan Matrik.
2. Baris 8 sampai baris 15 menjelaskan inisialisasi untuk perhitungan matrik hessian.
3. Baris 16 sampai baris 17 menjelaskan perulangan baris dan kolom matrik hessian.
4. Baris 18 sampai baris 20 menjelaskan operasi perhitungan matrik hessian.
5. Baris 21 sampai baris 23 menampilkan hasil dari perhitungan matrik hessian.

5.3.3 Implementasi Algoritma Perhitungan Iterasi

Proses iterasi yaitu merupakan proses menghitung jumlah iterasi maksimum. Iterasi pertama melakukan perhitungan mencari nilai Ei untuk setiap data *training*, kemudian melakukan perhitungan nilai delta alpha, dan setelah didapatkan nilai delta alpha dilanjutkan proses perhitungan nilai alpha.

a. Implementasi Algoritma Perhitungan Nilai *Ei*

Proses ini yaitu proses yang dilakukan pertama kali pada tahap perhitungan iterasi, nilai *EI* dilakukan dengan menjumlahkan hasil perkalian dari matrik hessian dengan nilai alpha yang ke-*j*. Pada proses tersebut merupakan proses awal iterasi dari *Sequential Training* SVM. Kode program dari algoritma perhitungan matriks *hessian* dijelaskan pada Kode Program 5.3.

```
1.  System.out.println("---------------HITUNG EI LEVEL 2---------
    ---");
2.     }
3.  alphaAwal=Float.parseFloat(txtAlphaAwal.getText());
4.  if(x1==itermax1){
5.  System.out.println("----------------MATRIK EI LEVEL 2--------
    ----");
6.          }
7.          for(i=0;i<barislt2;i++){
8.              for(l=0;l<barislt2;l++){
9.             matrix_ei2[i][l]=matrix_hessian2[i][l]*alpha2[i];
10.                 if(x1==itermax1){
11.                 System.out.print(" "+matrix_ei2[i][l]+" ");
12.                 }
13. }
14.         if(x1==itermax1){
15.            System.out.println("------------------------Nilai
    EI level 2-------------------");
16.         }
17.         for(i=0;i<barislt2;i++){
18.             tempJEi2=0;
19.              for(l=0;l<barislt2;l++){
20.                 tempJEi2=tempJEi2+matrix_ei2[i][l];
21.                 temp_matrix_ei2[i]=tempJEi2;
22.              }
23.              if(x1==itermax1){
24.  System.out.println("Jml Perbaris ei level 2 ke : "+(i+1)+" =
    "+df.format(temp_matrix_ei2[i]));
25.                 index_kolom[0]=temp_nopeserta2[i];
26.                index_kolom[1]=df.format(temp_matrix_ei2[i]);
27.                 mdl.addRow(index_kolom);
28.              }
29.         }
```

Kode Program 5.3 Implementasi Algoritma Perhitungan Nilai Ei

Berikut penjelasan dari Kode Program 5.3:

1. Baris 1 sampai baris 3 menjelaskan pengambilan nilai alpha sebelumnya atau alpha baru.
2. Baris 4 sampai baris 13 menjelaskan operasi perhitungan dari nilai *Ei* dari nilai alpha dijumlahkan dengan nilai dari matrik hessian.
3. Baris 14 sampai baris 22 menjelaskan operasi penjumlahan dari semua nilai perkalian alpha dan nilai dari matrik hessian.
4. Baris 23 sampai baris 29 menejelaskan hasil dari perhitungan *Ei.*

b. Implementasi Algoritma Perhitungan Nilai Delta Alpa (*δαi*)

Proses ini yaitu proses yang kedua pada tahap perhitungan iterasi, nilai delta alpha yaitu perkalian antara gamma dengan hasil dari 1 dikurangi dengan nilai *Ei* yang ke-*i* dan dibandingkan dengan nilai alpha ke-*i* untuk dicari nilai paling maksimum. Hasil dari nilai maksimum nantinya akan dibandingkan dengan hasil dari variabel *C* yang dikurangi nilai alpha ke-*i*. Kode program dari algoritma perhitungan nilai delta alpha dijelaskan pada Kode Program 5.4.

```
1.  for(i=0;i<barislt2;i++){
2.      tempda2=Math.min((Math.max(gamma2*(1-
    temp_matrix_ei2[i]), -1*alpha2[i])), c1-alpha2[i]);
3.          deltaAlpha2[i]=tempda2;
4.              if(x1==itermax1){
5.                iterKe2.setText(String.valueOf(itermax1));
6.                 System.out.println("Delta Alpha ke "+(i+1)+" =
    "+df.format(deltaAlpha2[i]));
7.                          }
```

Kode Program 5.4 Implementasi Algoritma Perhitungan Nilai Delta Alpha

Berikut penjelasan dari Kode Program 5.4:

1. Baris 1 menjelaskan tentang perulangan untuk data *training* ke-*i.*
2. Baris 2 sampai baris 3 menjelaskan operasi perhitungan dari nilai delta alpha.
3. Baris 5 sampai baris 7 menjelaskan hasil dari perhitungan nilai delta alpha.

c. Implementasi Algoritma Perhitungan Nilai Alpha Baru (*αi*)

Proses ini yaitu proses pada tahap ketiga pada proses iterasi, perhitungan nilai delta alpha merupakan proses penambahan nilai dari delta alpha ke-*i* pada proses sebelumnya. Kode program dari algoritma perhitungan alpha baru dijelaskan pada Kode Program 5.5.

```
1.  for(i=0;i<barislt2;i++){
2.  alpha2[i]=alpha2[i]+deltaAlpha2[i];
3.  maxDeltaAlpha2=Math.max(maxDeltaAlpha2, deltaAlpha2[i]);
4.  }
```

Kode Program 5.5 Implementasi Algoritma Perhitungan Nilai Alpha Baru

Berikut penjelasan dari Kode Program 5.5:

1. Baris 1 menjelaskan perulangan untuk baris nilai delta alpha ke –*i.*
2. Baris 2 dampai baris 4 menjelaskan proses operasi perhitungan nilai alpha baru.

5.3.4 Implementasi Algoritma Perhitungan Nilai $W.X^+$ dan $W.X^-$

Data latih atau data *training* memiliki bobot, yang merupakan bobot *training* di-*dot product* dengan nilai data *training* mempunyai nilai alpha yang tertinggi pada kelas positif ($W.X^+$), dan memiliki nilai tertinggi negatif ($W.X^-$).

Proses bobot bernilai positif merupakan perhitungan yang mengunakan kernel dari data *training* yang memiliki kelas positif dengan nilai alpha tertinggi dari kelas negatif, sedangkan proses bobot negatif melakukan perhitungan menggunakan kernel dari data *training* yang memiliki kelas negatif dengan nilai alpha tertinggi dikelas negatif. Kode program dari algoritma perhitungan kelas positif ($W.X^+$), dan kelas negatif ($W.X^-$) dijelaskan pada Kode Program 5.6.

```
1.  for(i=0;i<barislt;i++){
2.   wxpos1[i]=matrik_kxpos[i]*matrik_y[i]*alpha[i];
3.       temp_jumwxpos+=wxpos1[i];
4.   System.out.println("Nilai WX POS : "+wxpos1[i]);
5.   }
6.  for(i=0;i<barislt;i++){
7.   wxneg1[i]=matrik_kxneg[i]*matrik_y[i]*alpha[i];
8.        temp_jumwxneg+=wxneg1[i];
9.    System.out.println("Nilai WX POS : "+wxneg1[i]);
10   }
```

Kode Program 5.6 Implementasi Algoritma Perhitungan ($W.X^+$) dan ($W.X^-$).

Berikut penjelasan dari Kode Program 5.6:

1. Baris 1 menjelaskan tentang perulangan untuk data *training* ke-*i* dan ke-*l* kelas positif (*$W.X^+$)*.
2. Baris 2 sampai baris 3 menjelaskan operasi perhitungan dari kelas positif (*$W.X^+$).*
3. Baris 4 sampai baris 5 menjelaskan tentang hasil perhitungan dari kelas positif (*$W.X^+$).*
4. Baris 6 menjelaskan tentang perulangan untuk data *training* ke-*i* dan ke-*l* kelas negatif (*$W.X^-$).*
5. Baris 7 sampai baris 8 menjelaskan operasi perhitungan dari kelas negatif (*$W.X^-$).*
6. Baris 9 sampai baris 10 menjelaskan tentang hasil perhitungan dari kelas negatif (*$W.X^-$).*

5.3.5 Implementasi Algoritma Perhitungan Nilai Bias (*b*)

Perhitungan nilai bias membutuhkan total jumlah bobot yang didapatkan dari *dot product* dengan yang ada pada data kelas positif dan total jumlah bobot yang di-*dot product* dengan yang ada pada data kelas negatif. Kedua bobot yang didapat yaitu bobot positif (*$W.X^+$)* dan bobot negatif (*$W.X^-$)* kemudian dijumlahkan dan dikalikan dengan minus setengah sehingga menghasilkan nilai bias. Kode program dari algoritma perhitungan nilai bias dijelaskan pada Kode Program 5.7.

```
1.  bias2=-(temp_jumwxpos2+temp_jumwxneg2)/2;
2.  txtBias2.setText(String.valueOf(df.format(bias2)));
3.   }
```

Kode Program 5.7 Implementasi Algoritma Perhitungan Nilai Bias

Berikut penjelasan dari Kode Program 5.7:

1. Baris 1 menjelaskan tentang operasi perhitungan nilai bias
2. Baris 2 sampai baris 3 menjelaskan hasil perhitungan nilai bias.

5.3.6 Implementasi Algoritma Perhitungan Nilai *f(x)*

Perhitungan nilai *f(x)* pada tahap pengujian dengan menggunakan data *testing* dengan parameter nilai bias. Kemudian dari hasil fungsi nilai *f(x)* ini adalah

bernilai positif atau negatif, apabila hasil nilai uji bernilai positif maka data yang diuji masuk dikelas A atau sangat layak, dan apabila hasil bernilai negatif maka akan dilanjutkan pada proses perhitungan dilevel 2 yang nantinya akan menghasilkan nilai *f(x)* untuk menentukan kelas B atau layak dan kelas C atau tidak layak. Kode program dari algoritma perhitungan nilai *f(x)* dijelaskan pada Kode Program 5.8.

```
1.    for(i=0;i<barislt2;i++){
2.                    ott2=0;
3.                    for(j=1;j<kolomlt2;j++){
4.  dates2[itertt2]=Float.parseFloat(tbl_norm_tt_lv2.getValueAt(
    itertt2, j).toString());
5.              ott2=ott2+(dates2[itertt2]*matrix_norm_lt2[i][j]);
6.                        temp_kxtt2[i]=ott2;
7.                    }
8.                    t_kxtt2=(double)temp_kxtt2[i];
9.                    texkxtt2=Math.pow(t_kxtt2, d);
10.                   matrik_kxtt2[i]=(float)texkxtt2;
11.   System.out.print(" K(xi,x) ke : "+matrik_kxtt2[i]+" ");
12.                   System.out.println("");
13.               }
14.              System.out.println("-----------------mencari
    aiyik(xi,x) level 2");
15.               tempaiyi2=0;
16.               for(i=0;i<barislt2;i++){
17. temp_aiyi2[i]=alpha2[i]*matrik_kxtt2[i]*matrik_y2[i];
18.          System.out.println("Nilai AiYi : "+temp_aiyi2[i]);
19.                   tempaiyi2=tempaiyi2+temp_aiyi2[i];
20.  System.out.println("Jumlah AIYI : "+tempaiyi2+" LEVEL 2 ");
21.               tempnilaifx2=tempaiyi2+bias2;
22. System.out.println("NILAI F(X) : "+tempnilaifx2+" LEVEL 2");
23.               abs2=Math.signum(tempnilaifx2);
24.               temp_nilaifx2[itertt2]=abs2;
25. System.out.println("SIGN 2 : "+temp_nilaifx2[itertt2]+" level
    2 ");
26. }
```

Kode Program 5.8 Implementasi Algoritma Perhitungan Nilai *f(x)*

Berikut penjelasan dari Kode Program 5.8:

1. Baris 1 sampai baris 3 menjelaskan perulangan dari pengambilan baris serta kolom data latih dan data uji ke-*i* dan ke-*j*.
2. Baris 4 sampai baris 10 menjelaskan operasi perhitungan perkalian kernel antara data uji dan data latih.
3. Baris 11 sampai baris 13 menjelaskan hasil dari nilai perkalian kernel data uji dan data latih.
4. Baris 14 sampai baris 20 menjelaskan proses perhitungan *aiyik(xi,x)* dan menampilkan hasilnya.
5. Baris 21 sampai baris 24 menjelaskan proses perhitungan nilai *f(x)*.

6. Baris 25 dan baris 26 menjelaskan hasil dari nilai *f(x).*

5.3.7 Implementasi Algoritma Perhitungan Nilai Normalisasi Matrik Keputusan

Data peserta yang akan dilanjutkan pada proses topsis merupakan data *testing* yang bernilai positif, sehingga menjadi sebuah matrik keputusan, dan matriks keputusan tersebut akan dinormalisasi. Kode program dari algoritma perhitungan nilai normalisasi matrik keputusan dijelaskan pada Kode Program 5.9.

```
1.  for(i=0;i<baristopa;i++){
2.    for(j=1;j<kolomtopa;j++){
3.  doubleann=(double)matrix_topa[i][j]/Math.sqrt(jumkol[j]);
4.     matrix_normtopa[i][j]=(float)ann;
5.   System.out.print(" | "+matrix_normtopa[i][j]+" | ");
6.  }
```

Kode Program 5.9 Implementasi Algoritma Perhitungan Nilai Normalisasi Matriks Keputusan

Berikut penjelasan dari Kode Program 5.9:

1. Baris 1 sampai baris 2 menjelaskan pengulangan untuk ambil data baris dan kolom.
2. Baris 3 sampai baris 4 menjelaskan operasi perhitungan normalisasi matriks keputusan.
3. Baris 4 sampai baris 6 menjelaskan hasil normalisasi matriks keputusan.

5.3.8 Implementasi Algoritma Perhitungan Pembobotan Matrik Normalisasi

Perhitungan pembobotan matrik normalisasi merupakan pencarian nilai *V*, nilai *V* merupakan matrik bobot ternormalisasi, yang mana nilai *V* didapatkan dari matrik ternomalisasi dikalikan dengan bobot yang ditentukan. Kode program dari algoritma perhitungan pembobotan matrik normalisasi dijelaskan pada Kode Program 5.10.

```
1.  for(i=0;i<baristopa;i++){
2.      for(j=1;j<kolomtopa;j++){
3.  bobot_norm[i][j]=matrix_normtopa[i][j]*bobot_kriteria[j];
4.    System.out.print(" | "+bobot_norm[i][j]+" | ");
5.   }
```

Kode Program 5.10 Implementasi Algoritma Perhitungan Pembobotan Matriks Normalisasi

Berikut penjelasan dari Kode Program 5.10:

1. Baris 1 sampai baris 2 menjelaskan perulangan untuk ambil data baris dan kolom kelas A.
2. Baris 3 menjelaskan operasi perhitungan pembobotan matriks normalisasi
3. Baris 4 sampai baris 5 menjelaskan hasil pembobotan matriks normalisasi matriks.

5.3.9 Implementasi Algoritma Perhitungan Solusi Ideal Positif dan Negatif

Pada implementasi algoritma perhitungan solusi ideal positif dan negatif digunakan untuk mencari nilai maksimal dan minimal dari setiap kriteria yang ada.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Kode program dari algoritma perhitungan solusi ideal positif dan solusi ideal negatif dijelaskan pada Kode Program 5.11.

```
1.  for(j=1;j<kolomtopa;j++){
2.  ideal_posa[j]=Float.parseFloat(tblIposA.getValueAt(0,
    j).toString());
3.  ideal_nega[j]=Float.parseFloat(tblIposA.getValueAt(1,
    j).toString());
4.  System.out.println("BOBOT IPOS : "+ideal_nega[j]);
5.          }
```

Kode Program 5.11 Implementasi Algoritma Perhitungan Solusi Ideal Positif dan Negatif

Berikut penjelasan dari Kode Program 5.11:

1. Baris 1 menjelaskan perulangan sejumlah banyaknya kolom untuk perhitungan maksimum dan minimum.
2. Baris 2 menjelaskan operasi perhitungan solusi ideal positif.
3. Baris 3 menjelaskan operasi perhitungan solusi ideal negatif.
4. Baris 4 sampai baris 5 menjelaskan hasil perhitungan solusi ideal positif dan negatif.

5.3.10 Implementasi Algoritma Perhitungan *Separation Measure*

Separation measure mendapatkan nilai dari dua hasil akhir, yaitu jarak dari solusi ideal positif dan solusi ideal negatif, *Separation measure* positif didapatkan dari hasil *V* dikurangkan dengan solusi ideal positif kemudian dipangkatkan dan diakar, *Separation measure* negatif didapatkan dari hasil *V* dikurangkan dengan solusi ideal negatif kemudian dipangkatkan dan diakar. Kode program dari algoritma perhitungan *Separation measure* dijelaskan pada Kode Program 5.12.

```
1.  for(i=0;i<baristopa;i++){
2.      for(j=1;j<kolomtopa;j++){
3.  temp_siposa=(double)bobot_norm[i][j]-ideal_posa[j];
4.      temp_sposa +=Math.pow(temp_siposa, 2);
5.      t_sposa=Math.sqrt(temp_sposa);
6.  temp_sinega=(double)bobot_norm[i][j]ideal_nega[j];
7.      temp_snega +=Math.pow(temp_sinega, 2);
8.      t_snega=Math.sqrt(temp_snega);
9.   }
10.      siposa[i]=(float)t_sposa;
11.      sinega[i]=(float)t_snega;
12.      sum_sema[i]=siposa[i]+sinega[i];
13.  System.out.println(" SI sum    "+df.format(sum_sema[i]));
14.      System.out.println("");
15.          }
```

Kode Program 5.12 Implementasi Algoritma Perhitungan *Separation Measure*

Berikut penjelasan dari Kode Program 5.12:

1. Baris 1 sampai baris 2 menjelaskan perulangan sejumlah banyaknya kolom dan baris.
2. Baris 3 sampai baris 5 menjelaskan operasi perhitungan *Separation Measure* positif.

3. Baris 6 sampai baris 9 menjelaskan operasi perhitungan *Separation Measure* negatif.
4. Baris 10 sampai baris 12 menjelaskan operasi perhitungn jumlah dari *Separation Measure* positif dan negatif.
5. Baris 13 sampai baris 15 menjelaskan hasil perhitungan *Separation Measure* positif dan negatif.

5.3.11 Implementasi Algoritma Perhitungan Preferensi

Perhitungan preferensi merupakan proses tahap akhir TOPSIS, yang mana dari nilai preferensi dapat ditentukan hasil perangkingan. Untuk perhitungan preferensi didapatkan dari pembagian dari *Separation Measure* negatif dibagi dengan jumlah *Separation Measure* positif dengan negatif. Setelah itu dilakukan proses perangkingan, diambil dari nilai preferensi tertinggi. Kode program dari algoritma perhitungan preferensi dijelaskan pada Kode Program 5.13.

```
1.  for(i=0;i<baristopa;i++){
2.      prefa[i]=sinega[i]/sum_sema[i];
3.      System.out.println("NILAI PREF : "+prefa[i]);
4.      }
5.  for(i=0;i<baristopa;i++){
6.          tbl_prefa.setValueAt((i+1), i, 2);
7.           }
```

Kode Program 5.13 Implementasi Algoritma Perhitungan Preferensi

Berikut penjelasan dari Kode Program 5.13:

1. Baris 1 menjelaskan perulangan.
2. Baris 2 menjelaskan operasi perhitungan preferensi.
3. Baris 3 sampai baris 4 menjelaskan hasil dari preferensi.
4. Baris 5 menjelaskan perulangan.
5. Baris 6 sampai 7 menjelaskan perangkingan dari nilai preferensi.

5.4 Implementasi Antarmuka

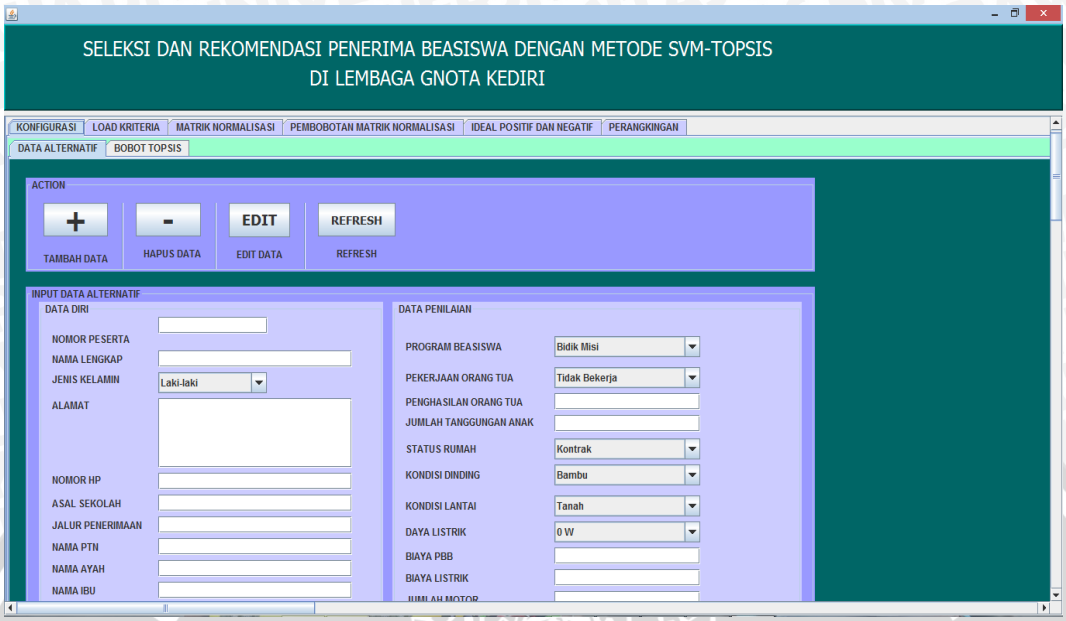
Implementasi antarmuka atau *interface* merupakan penghubung antara *user* dan sistem secara langsung. *Interface* dari sistem Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri dengan Implementasi Metode SVM dan TOPSIS terdiri dari Halaman utama atau *home*, halaman *load data*, halaman normalisasi, halaman kernel *polynomial* dan matrik hessian, halaman iterasi, halaman nilai *f(x),* dan halaman TOPSIS.

5.4.1 Antarmuka Halaman Home

Implementasi halaman *home* merupakan akses untuk *administrator* untuk mengisikan biodata dari pendaftar beasiswa lembaga GNOTA Kabupaten Kediri, yang mana ada beberapa menu sesuai dengan formulir pendaftaran yang telah disebar oleh pihak lembaga GNOTA, sehingga *admin* bisa secara langsung

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

mengisikan data kedalam sistem. implementasi halaman *home* terdapat pada Gambar 5.2.

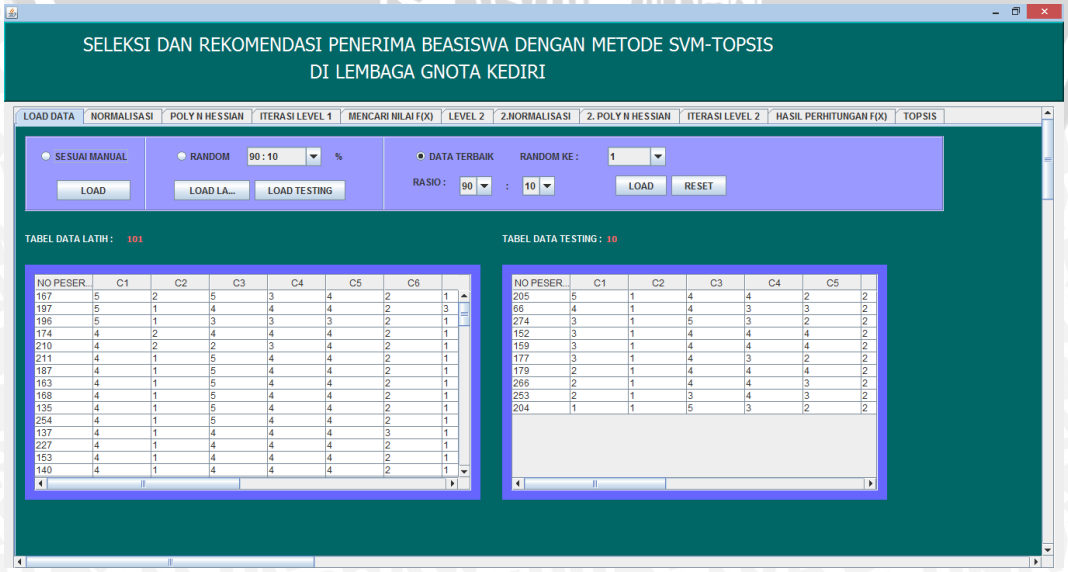


Gambar 5.2 Antarmuka Halaman Home

Gambar 5.2 menjelaskan antarmuka admin untuk mengisi data biodata dari pendaftar beasiwa GNOTA Kediri, yang mana *input* tersebut sama dengan formulir pendaftaran yang disebar oleh pihak GNOTA Kediri.

5.4.2 Antarmuka Halaman Load Data

Implementasi pada halaman antarmuka *load data* merupakan antarmuka untuk mendukung proses perhitungan SVM, yang mana *load data* untuk mengambil data dari *database,* yang mana data yang diambil yaitu data *training* dan data *testing*. Implementasi halaman antarmuka *load* data terdapat pada Gambar 5.3.



Gambar 5.3 Halaman Antarmuka Load Data

Gambar 5.3 menjelaskan antarmuka *load data,* yang memiliki 9 data perbandingan, yaitu antara data *training* dan data *testing*, sehingga masing-masing kelas akan dibandingkan oleh perbandingan data.

5.4.3 Antarmuka Halaman Normalisasi

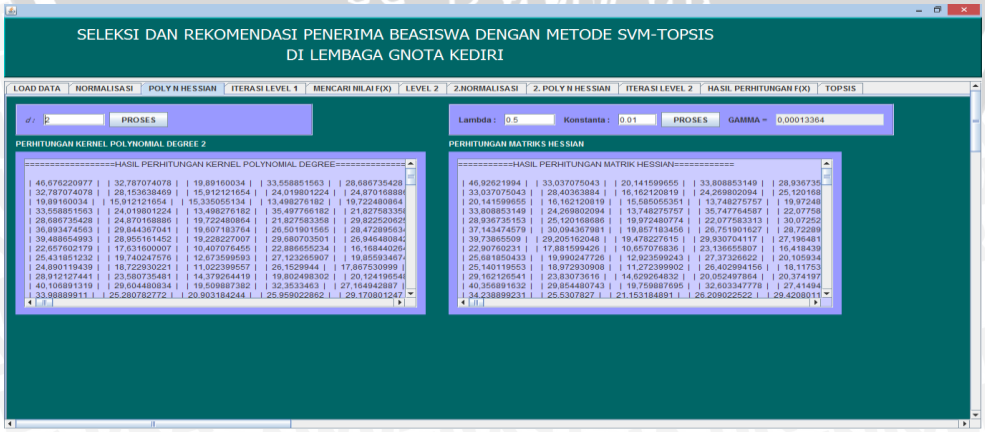
Implementasi pada halaman antarmuka normalisasi merupakan antarmuka untuk melakukan proses normalisasi dari semua data yang dimiliki, yaitu data *training* dan data *testing*. Implementasi halaman antarmuka normalisasi terdapat pada Gambar 5.4.

Gambar 5.4 Halaman Antarmuka Normalisasi

Gambar 5.4 menjelaskan antarmuka dari proses normalisasi perhitungan metode SVM, yang mana proses normalisasi ada pada data *training* dan data *testing.*

5.4.4 Antarmuka Halaman Kernel Polynomial dan Matriks Hessian

Implementasi pada halaman antarmuka Kernel *polynomial* dan matrik hessian merupakan antarmuka untuk perhitungan kernel dan matrik atau mencangkup *prosessing*, yang mana halaman antarmuka ini terdiri dari *level* 1 dan *level* 2. Implementasi halaman antarmuka kernel *polynomial* dan matriks hessian *level 1* dan *level* 2 terdapat pada Gambar 5.5 dan Gambar 5.6.



Gambar 5.5 Halaman Antarmuka Kernel Polynomial da Matriks Hessian Level 1

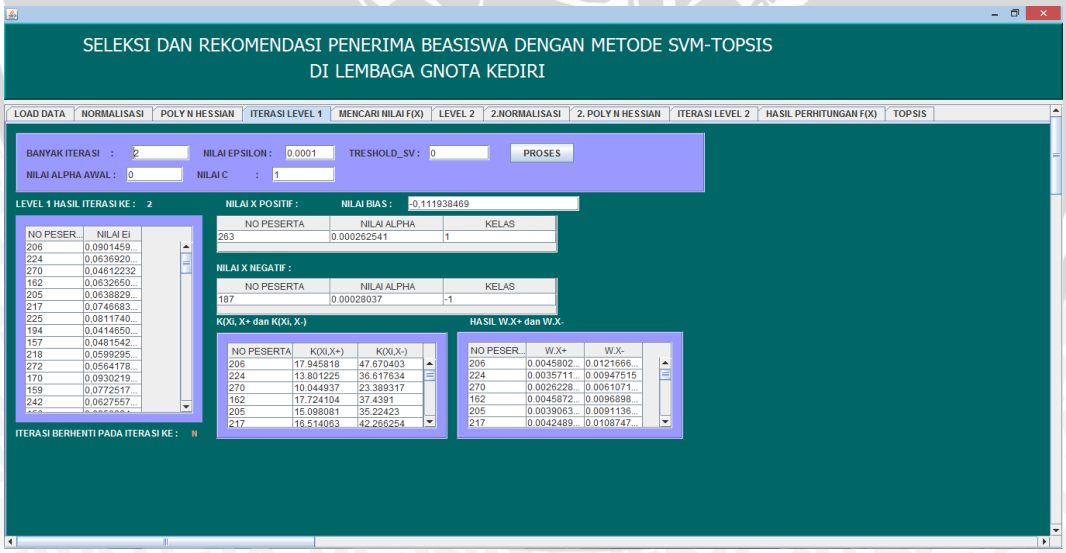
Gambar 5.5 menjelaskan proses perhitungan polynomial pada level 1 dan proses perhitungan matrik hessian level 1 pada metode SVM.

Gambar 5.6 Halaman Antarmuka Kernel Polynomial dan Matrik Hessian Level 2

Gambar 5.6 menjelaskan proses perhitungan polynomial pada level 1 dan proses perhitungan matriks hessian level 1 pada metode SVM.

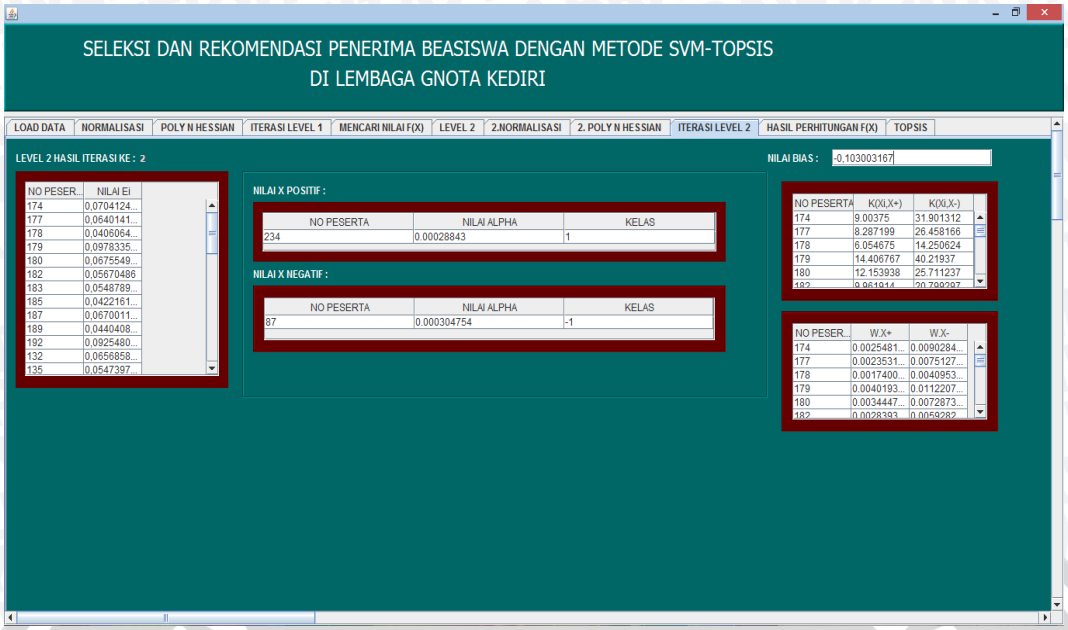
5.4.5 Antarmuka Halaman Iterasi

Implementasi pada halaman antarmuka iterasi merupakan termasuk salah satu proses perhitungan SVM. Implementasi halaman antarmuka iterasi level 1 dan level 2 terdapat pada Gambar 5.7 dan Gambar 5.8.



Gambar 5.7 Halaman Antarmuka Iterasi Level 1

Gambar 5.7 menjelaskan dari perhitungan iterasi pada level satu yang ada pada metode SVM.



Gambar 5.8 Halaman Antarmuka Iterasi Level 2

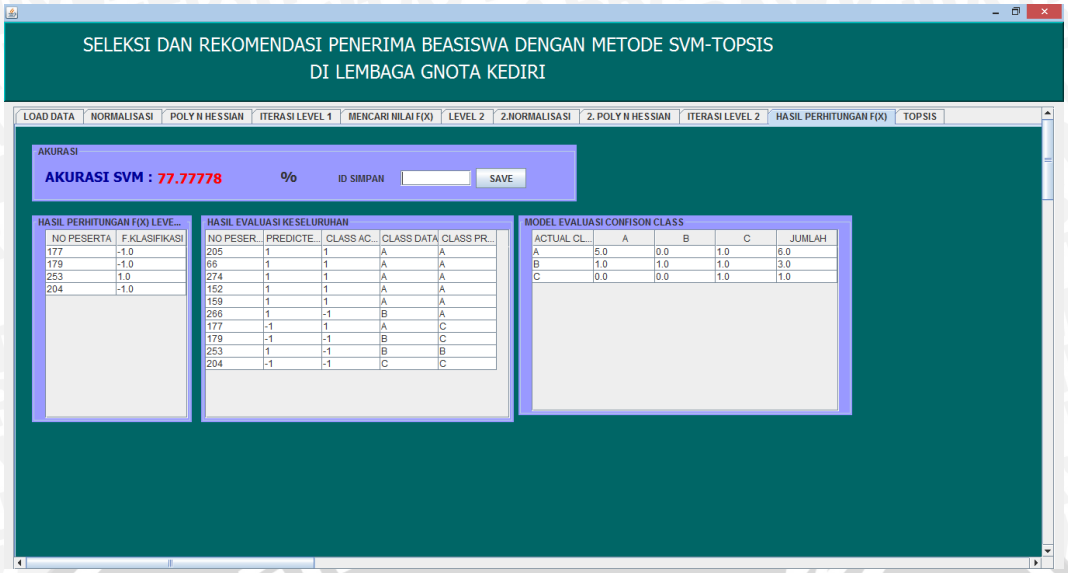
Gambar 5.8 menjelaskan dari perhitungan iterasi pada level dua yang ada pada metode SVM.

5.4.6 Antarmuka Halaman Nilai *f(x)*

Implementasi halaman antarmuka nilai *f(x)* merupakan antarmuka untuk menentukan sebuah kelas dari metode SVM, apakah termasuk kelas sangat layak, layak dan tidak layak. Implementasi halaman antarmuka nilai *f(x)* level 1 dan antarmuka nilai *f(x)* terdapat pada Gambar 5.9 dan Gambar 5.10.

Gambar 5.9 Halaman Antarmuka Nilai *f(x)* Level 1

Gambar 5.9 menjelaskan dari nilai *f(x)* pada level 1 perhitungan metode SVM, yang mana level 1 menjelaskan nilai +1 dan -1, yang bernilai negatif 1 akan dilanjutkan di perhitungan level 2.



Gambar 5.10 Halaman Antarmuka Nilai *f(X)* Level 2

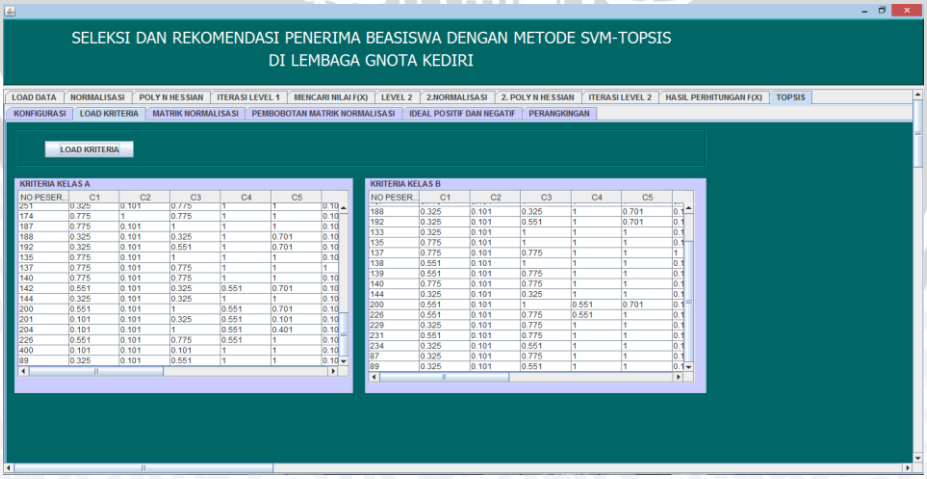
Gambar 5.10 menjelaskan dari nilai *f(x)* pada level 1 perhitungan metode SVM, yang mana level 1 menjelaskan nilai +1 dan -1.

5.4.7 Antarmuka Halaman TOPSIS

Implementasi antarmuka topsis terdiri dari beberapa halaman antarmuka, yaitu terdiri dari *load* kriteria, matriks normalisasi,pembobotan matriks normalisasi, ideal positif dan ideal negatif, dan perangkingan, adapun halaman antarmuka metode TOPSIS sebagai berikut.

a. Antarmuka Halaman Load Kriteria

Antarmuka *load* kriteria pada metode topsis merupakan *load* data dari metode SVM, sehingga dapat dilanjutkan pada *load*. Antarmuka *load* kriteria terdapat pada Gambar 5.11.

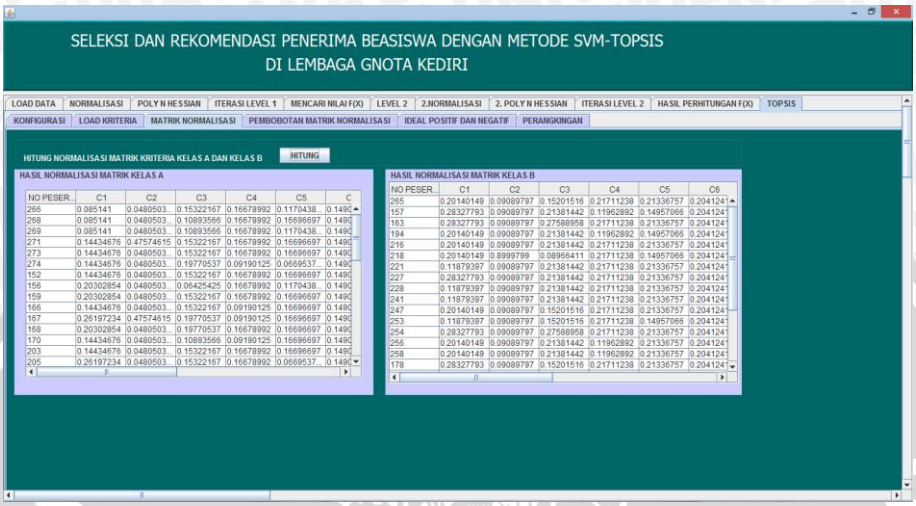


Gambar 5.11 Halaman Antarmuka *Load* Kriteria

Gambar 5.11 menjelaskan antarmuka *load* kriteria pada metode TOPSIS, yang melanjutkan dari metode SVM.

b. Antarmuka Halaman Matrik Normalisasi

Antarmuka matrik normalisasi pada metode TOPSIS merupakan normalisasi terhadap data terpilih pada proses SVM, yang merupakan kelas A yaitu sangat layak dan kelas B layak. Antarmuka matriks normalisasi terdapat pada Gambar 5.12.

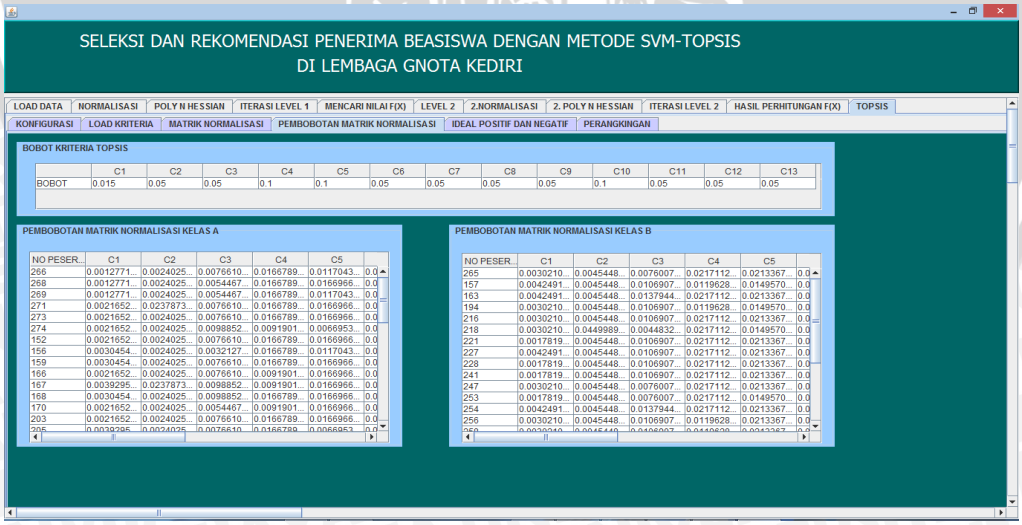


Gambar 5.12 Antarmuka Halaman Matrik Normalisasi

Gambar 5.12 menjelaskan antarmuka matrik normalisasi pada metode TOPSIS, dari kelas A dan B yaitu sangat layak dan layak.

c. Antarmuka Halaman Pembobotan Matriks Normalisasi

Antarmuka halaman pembobotan matriks normalisasi pada metode TOPSIS menampilkan pembobotan dari matriks yang sudah dinormalisasi. Antarmuka pembobotan matriks normalisasi terdapat pada Gambar 5.13.

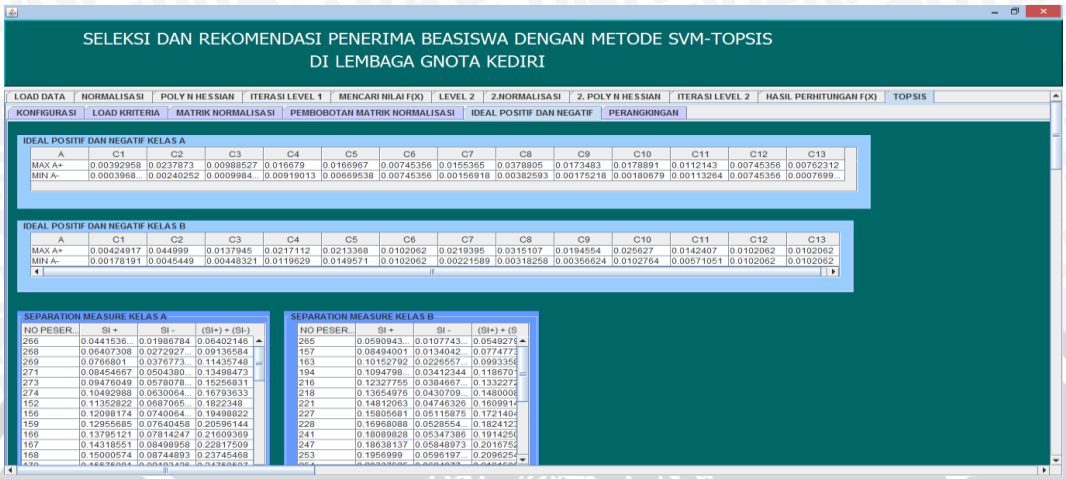


Gambar 5.13 Antarmuka Halaman Pembobotan Matriks Normalisasi

Gambar 5.13 merupakan antarmuka dari halaman pembobotan matriks normalisasi yang ada pada metode TOPSIS.

d. Antarmuka Halaman Ideal Positif dan Ideal Negatif

Antarmuka Halaman ideal positif dan ideal negatif merupakan antarmuka mencari nilai maksimum dan minimum dari matrik ternormalisasi, dan selain ideal positif dan ideal negatif ada juga antarmuka *separation measure*. Antarmuka ideal positif dan ideal negatif terdapat pada Gambar 5.14.



Gambar 5.14 Antarmuka Halaman Ideal Positif dan Ideal Negatif

Gambar 5.14 merupakan antarmuka dari halaman hasil ideal positif dan negatif yang mana akan melakukan pencarian nilai separation measure.

e. Antarmuka Halaman Perangkingan

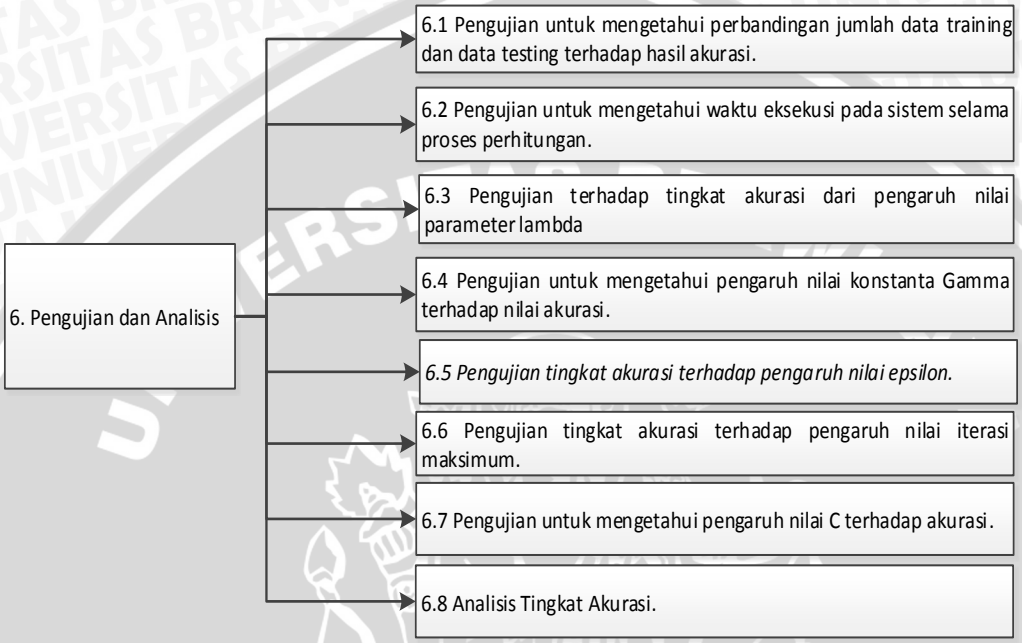
Antarmuka halaman perangkingan merupakan antarmuka untuk menampilkan hasil akhir dari metode TOPSIS, yang mana dari nilai preferensi tertinggi akan diurutkan pada perangkingan masing-masing kelas, kemudian diambil beberapa data untuk memenuhi kuota yang disediakan oleh pihak GNOTA Kediri. Antarmuka perangkingan terdapat pada Gambar 5.15.

Gambar 5.15 Antarmuka Halaman Perangkingan

Gambar 5.15 menunjukkan halaman antarmuka dari halaman perangkingan, yang juga bisa dilakukan pembobotan kuota.

BAB 6 PENGUJIAN DAN ANALISIS

Bab ini akan membahas tentang analisis dan pengujian terhadap sistem yang telah dibuat pada tahap implementasi, yang mana proses pengujian merupakan pengujian yang akan dilakukan 7 pengujian yang mana pengujian tersebut akan dijelaskan dalam diagram proses pengujian dan analisis. Berikut proses pengujian yang akan dijelaskan melalui Gambar 6.1.



Gambar 6.1 Diagram Proses Pengujian dan Analisis

Gambar 6.1 menunjukkan proses pengujian yang mana terdiri dari 6 pengujian terdiri dari rasio data, waktu eksekusi, parameter lambda, parameter konstanta gamma, nilai epsilon, *iterasi* maksimum,parameter *C* dan yang terakhir analisis tingkat akurasi.

6.1 Pengujian Untuk Mengetahui Perbandingan Data *Training* dan Data *Testing* Terhadap Hasil Akurasi.

Pengujian untuk mengetahui perbandingan data *training* dan data *testing* terhadap hasil akurasi untuk mendapatkan hasil akurasi yang terbaik menjelaskan pengujian skenario pengujian dan analisa skenario.

6.1.1 Skenario pengujian Rasio Perbandingan

Pengujian untuk mengetahui perbandingan data *training* dan data *testing* terhadap hasil akurasi untuk mendapatkan hasil akurasi yang terbaik. Data set yang digunakan sebanyak 111 data yang tediri dari 3 kelas yaitu sangat layak, layak, dan tidak layak, pada pengujian perbandingan ini, rasio yang digunakan yaitu 90%:10%, 80%:20%, 70%:30%, 60%:40%, 50%:50%, 40%:60%, 30%:70%, 20%:80%, dan 10%:90%. Nilai parameter *Sequential Training* SVM yang digunakan dalam pengujian ini adalah λ = 0.5,konstanta γ = 0.01, ε = 0.0001, *itermax* = 100,

C = 1 serta nilai *d* = 2 Dan hasil dari pengujian terhadap rasio perbandingan data *training* dan data *testing* dapat dilihat pada Tabel 6.1.

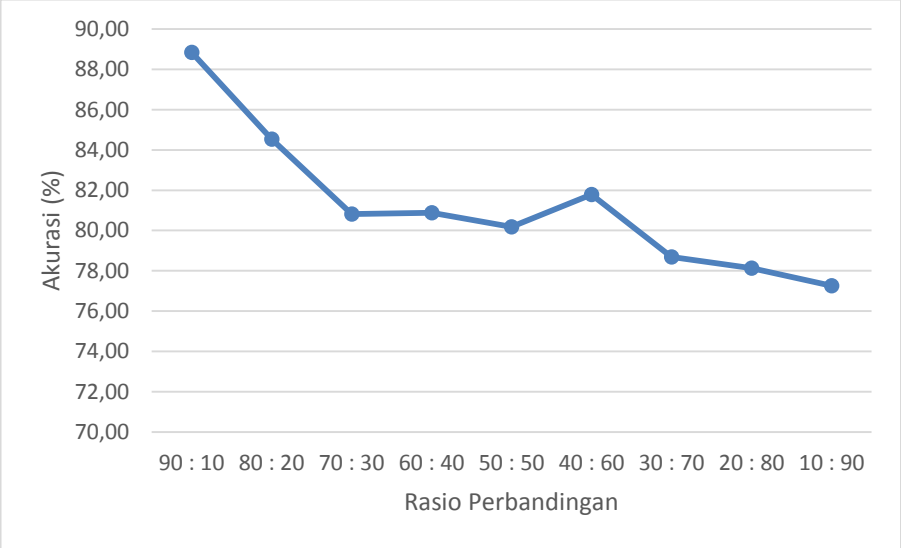
Tabel 6.1 Hasil Pengujian untuk mengetahui perbanndingan data *training* dan data *testing*.

Rasio (%)	Percobaan Ke-*i*					Rata-rata
	1	2	3	4	5	
90 : 10	90,00	82,75	85,71	85,71	100,00	88,83
80 : 20	82,25	78,68	82,25	93,75	85,71	84,53
70 : 30	89,01	74,15	76,74	83,33	80,89	80,82
60 : 40	85,03	78,81	83,33	77,41	79,83	80,88
50 : 50	79,47	83,44	76,55	80,53	80,92	80,18
40 : 60	77,41	88,77	76,24	82,10	84,37	81,78
30 : 70	75,70	75,36	83,33	77,77	81,30	78,69
20 : 80	75,30	76,22	76,51	80,31	82,35	78,14
10 : 90	80,43	77,69	72,76	79,34	76,08	77,26

Tabel 6.1 menunjukkan hasil dari pengujian terhadap rasio perbandingan data *training* dan data *testing* dapat sehingga didapatkan hasil seperti tersebut.

6.1.2 Analisa Pengujian Rasio Perbandingan

Berdasarkan hasil dari grafik pada Gambar 6.2 diketahui bahwa rata-rata tingkat akurasi sebesar 88,83% dan akurasi terbaik sebesar 100% pada rasio 90:10. Sehinggga didapatkan rasio yang digunakan pada pengujian selanjutnya sebesar 90%:10%. Hasil akurasi pengujian rasio perbandingan dijelaskan pada Gambar 6.2.



Gambar 6.2 Grafik Tingkat Akurasi Hasil Pengujian Rasio Perbandingan

Gambar 6.2 menunjukkan hasil tingkat akurasi berupa grafik tersebut dengan hasil tingkat akurasi tertinggi pada rasio perbandingan 90%:10% dengan rata-rata sebesar 88,83%.

Analisis dilakukan pada pengujian rasio perbandingan ini yaitu tingkat akurasi yang berpengaruh terhadap rasio perbandingan, dengan didapatkannya nilai akurasi pada perbandingan data latih yang lebih dominan maka dapat diketahui bahwa semakin tinggi nilai perbandingan pada data latih maka akan semakin tinggi tingkat akurasinya, sehingga pada sebuah sistem apabila data semakin banyak dilatih maka data tersebut akan semakin cerdas dalam mendapatkan tingkat akurasi tertinggi, dan jika sebaliknya dari Gambar 6.2 semakin rendah perbandingan data latih dibanding data uji maka akurasinya semakin rendah.

6.2 Pengujian Untuk Mengetahui Waktu Eksekusi Pada Sistem Selama Proses Perhitungan.

Pengujian untuk mengetahui waktu eksekusi pada sistem selama proses perhitungan untuk mengetahui berapa lama sistem untuk mengeksekusi perhitungan dengan metode SVM dan TOPSIS serta menjelaskan hasil skenario waktu eksekusi sistem dan analisa.

6.2.1 Skenario Pengujian Waktu Eksekusi

Pengujian waktu eksekusi untuk mengetahui berapa lama sistem untuk memproses perhitungan metode SVM dan TOPSIS pada seleksi beasiswa GNOTA kediri, mana pada pengujian waktu eksekusi ini mengambil waktu eksekusi dari pengujian rasio perbandingan yang terdiri dari 90%:10%, 80%:20%, 70%:30%, 60%:40%, 50%:50%, 40%:60%, 30%:70%, 20%:80%, dan 10%:90%, dan beberapa paramter dari metode *Sequential Training* SVM yaitu λ = 0.5,Konstanta γ = 0.01, ε = 0.0001, *itermax* = 100, C = 1, serta nilai d = 2 dan hasil dari pengujian waktu eksekusi yaitu ditunjukkan pada Tabel 6.2.

Tabel 6.2 Pengujian Untuk Mengetahui Waktu Eksekusi Pada Sistem Selama Proses Perhitungan

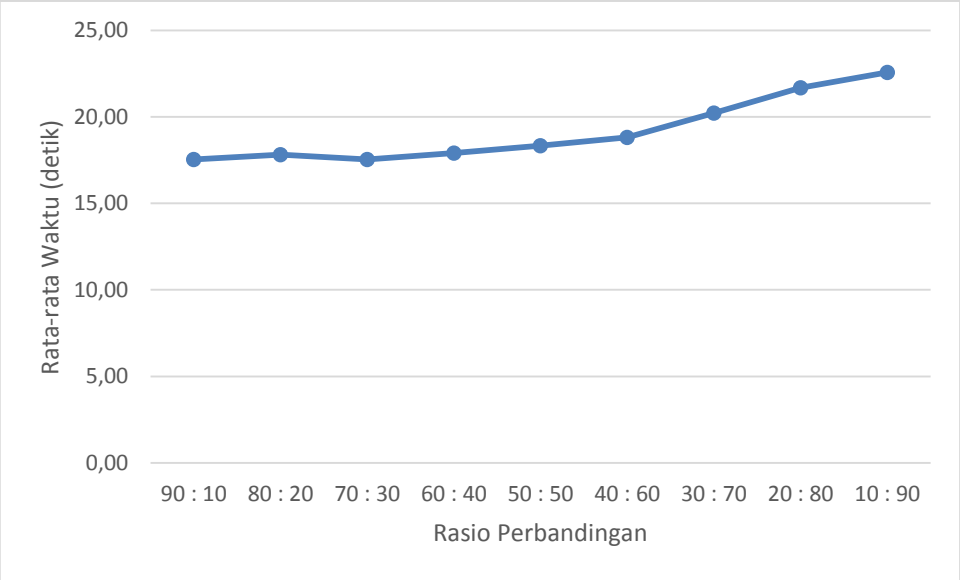
Rasio (%)	Percobaan Ke-*i*					Rata-rata (detik)
	1	2	3	4	5	
90 : 10	13,39	18,70	17,63	18,00	19,99	17,54
80 : 20	17,15	16,59	19,77	16,55	19,01	17,81
70 : 30	19,51	17,22	17,34	16,01	17,66	17,55
60 : 40	18,58	17,74	18,80	17,15	17,24	17,90
50 : 50	17,95	17,65	16,37	22,34	17,41	18,34
40 : 60	19,38	17,91	17,19	17,24	22,38	18,82
30 : 70	20,39	22,86	18,52	20,53	18,85	20,23
20 : 80	19,43	22,64	21,86	21,06	23,40	21,68
10 : 90	21,70	22,54	24,07	22,08	22,48	22,57
RATA-RATA KESELURUHAN						19,16

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Tabel 6.1 menunjukkan hasil dari pengujian untuk mengetahui waktu eksekusi pada sistem selama proses perhitungan rasio data perbandingan, sehingga dihasilkan seperti Tabel 6.1.

6.2.2 Analisa Pengujian Waktu Eksekusi

Berdasarkan hasil grafik pada Gambar 6.3 diketahui bahwa hasil rata-rata keseluruhan dari pengujian waktu eksekusi yaitu 19,16 detik, waktu terlama yaitu 24,07 detik dan waktu tercepat yaitu 13,39 detik. Hasil dari pengujian waktu eksekusi dijelaskan pada Gambar 6.3.



Gambar 6.3 Grafik Waktu Eksekusi Pada Sistem Selama Proses Perhitungan

Gambar 6.3 menunjukkan hasil dari waktu eksekusi sistem untuk memproses perbandingan rasio,yang mana semakin banyak perbandingan data *testing*-nya semakin banyak memakan waktu eksekusinya dikarenakan proses perhitungan nilai *f(x)* mengunakan data *testing* jadi bila banyak datanya tersebut maka sebanyak itu juga melakukan proses perhitungannya.

6.3 Pengujian Terhadap Tingkat Akurasi dari Pengaruh Nilai Prameter *Lambda*.

Pengujian untuk mengetahui Tingkat Akurasi dari Pengaruh Nilai Prameter Lambda pada *Sequential Training* SVM, yang berupa skenario dan analisa.

6.3.1 Skenario Pengujian *Lambda*

Pengujian terhadap parameter dari *Sequential Training* SVM dilakukan untuk melihat pada skenario mana yang mendapatkan nilai terbaik untuk nilai λ (*Lambda*), pada pengujian ini menggunakan beberapa nilai yaitu 0.1, 0.3, 0.5, 1, 5, 7, 10, 20, 30 dan 40. Nilai parameter *Sequential Training* SVM yang digunakan adalah yaitu konstanta γ = 0.01, ε = 0.0001, *itermax* = 100, C = 1, rasio perbandingan 90%:10% dan nilai d=2. Adapun hasil dari pengujian *Lambda* dapat dilihat pada Tabel 6.3.

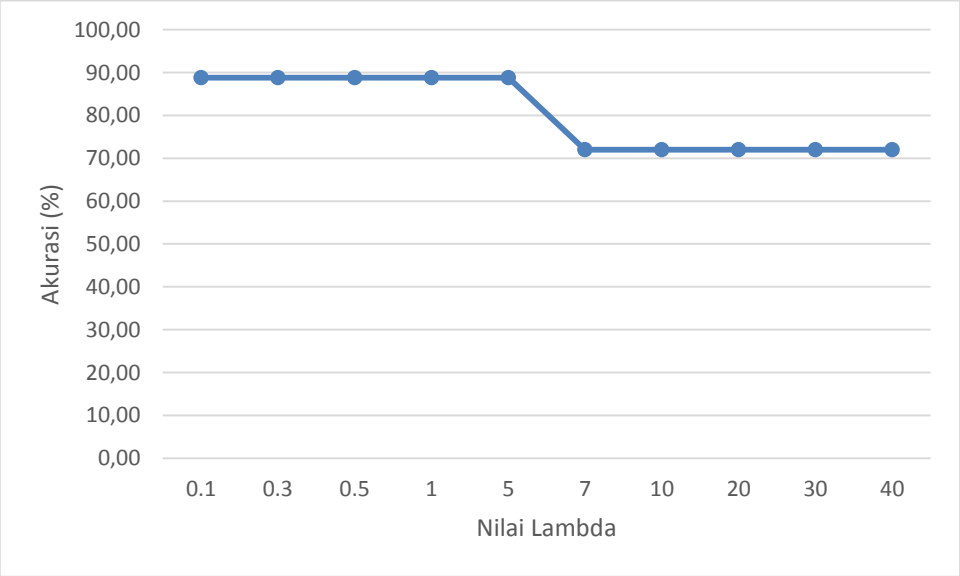
Tabel 6.3 Hasil Pengujian *Lambda*

Lambda	Percobaan Ke-*i*					Rata-rata
	1	2	3	4	5	
0.1	90,00	82,75	85,71	85,71	100,00	88,83
0.3	90,00	82,75	85,71	85,71	100,00	88,83
0.5	90,00	82,75	85,71	85,71	100,00	88,83
1	90,00	82,75	85,71	85,71	100,00	88,83
5	90,00	82,75	85,71	85,71	100,00	88,83
7	72,00	72,00	72,00	72,00	72,00	72,00
10	72,00	72,00	72,00	72,00	72,00	72,00
20	72,00	72,00	72,00	72,00	72,00	72,00
30	72,00	72,00	72,00	72,00	72,00	72,00
40	72,00	72,00	72,00	72,00	72,00	72,00

Tabel 6.3 merupakan tabel hasil dari pengujian nilai *lambda*, yang mana nilai *lambda* digunakan pada *Sequential Training* SVM.

6.3.2 Analisa Pengujian *Lambda*

Berdasarkan hasil grafik yang ada pada Gambar 6.4 diketahui bahwa rata-rata akurasi tertinggi sebesar 88.83% dan akurasi terbesar yaitu 100% pada nilai λ (*Lambda*) 0.1, 0.3, 0.5, 1, dan 5. Adapun hasil akurasi pengujian dijelaskan pada Gambar 6.4.



Gambar 6.4 Grafik Tingkat Akurasi Hasil Pengujian nilai *Lambda*

Gambar 6.4 merupakan hasil dari pengujian nilai λ (*Lambda)*, yang mana skenario pengujian *lambda* memiliki nilai rata-rata terbesar 88.83%, yang mana membuktikan dari Gambar 6.4 menunjukkan semakin besar nilai λ (*Lambda)* tidak membuat akurasi semakin besar. Akurasi terbesar pada nilai λ (*Lambda)* yaitu 0.1, 0.3, 0.5, 1 dan 5. Semakin besar nilai λ (*Lambda)* maka akurasi cenderung semakin

kecil, dikarenakan jika nilai λ (*Lambda)* semakin besar akan membuat proses perhitungan komputer atau komputasi pada tahap perhitungan matriks *Hessian* cenderung lebih lama. Kecenderungan proses komputasi yang lama juga disebabkan oleh *augmented factor* pada nilai λ (*Lambda)* sehingga proses perhitungan pada sistem sangat lambat untuk maksimalkan nilai margin dan akan terjadi ketidakstabilan pada proses *learning*.

6.4 Pengujian Untuk Mengetahui Pengaruh Nilai Konstanta *Gamma* Terhadap Nilai Akurasi

Pengujian untuk mengetahui pengaruh nilai konstanta *gamma* terhadap nilai akurasi, yang mana nilai *gamma* pada *Sequential Training* SVM akan berpengaruh terhadap nilai akurasi, dan pengujian untuk pengaruh nilai konstanta *gamma* terhadap akurasi menjelaskan skenario pengujian dan analisa pengujian nilai akurasi.

6.4.1 Skenario Pengujian Konstanta *Gamma*

Skenario pengujian konstanta *gamma* merupakan salah satu parameter *Sequential Training* SVM untuk melihat nilai akurasi, adapun konstanta yang digunakan yaitu 0.0001, 0.0001, 0.0005, 0.005, 0.02, 0.01, 0.05, 0.1, 0.5, dan 1. Dan nilai parameter yang digunakan lainnya yaitu λ = 0.1, ε = 0.0001, *itermax* = 100, C = 1, rasio perbandingan 90%:10% dan nilai d=2. Hasil pengujian terhadap nilai konstanta γ (*gamma)* ditunjukkan pada Tabel 6.4.

Tabel 6.4 Hasil Pengujian Kosntanta $\boldsymbol{\gamma}$ (*gamma)*

Konstanta Gamma	Percobaan Ke-*i*					Rata-rata
	1	2	3	4	5	
0,00001	72,00	72,00	72,00	72,00	77,77	73,15
0,0001	72,00	72,00	72,00	72,00	77,77	73,15
0,0005	72,00	72,00	72,00	72,00	77,77	73,15
0,005	72,00	72,00	72,00	72,00	77,77	73,15
0,01	90,00	82,75	85,71	85,71	100,00	88,83
0,02	90,00	82,75	93,10	93,10	93,10	90,41
0,05	90,00	93,10	100,00	93,10	93,10	93,86
0,1	57,69	50,00	50,00	28,57	40,90	45,43
0,5	37,50	26,08	39,13	28,57	28,57	31,97
1	37,50	26,08	39,13	28,57	28,57	31,97

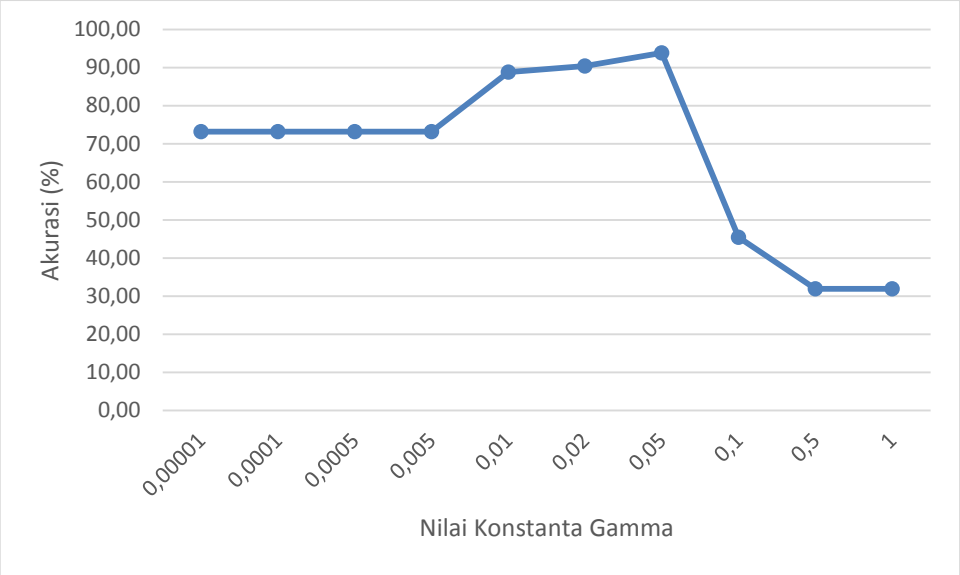
Tabel 6.4 merupakan hasil dari pengujian nilai Konstanta γ (*gamma),* yang merupakan *Sequential Training* SVM.

6.4.2 Analisa Pengujian Konstanta *Gamma*

Berdasarkan grafik pada Gambar 6.5 diketahui bahwa rata-rata tingkat akurasi tertinggi sebesar 93.86% dan akurasi terbaik sebesar 100%, pada nilai konstanta *gamma* 0.05. sehingga nilai parameter tertinggi akan digunakan pada

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

pengujian ε (*epsilon)*. Hasil pengujian konstanta *gamma* dijelaskan pada Gambar 6.5.



Gambar 6.5 Grafik Tingkat Akurasi Hasil Pengujian Konstanta *Gamma*

Gambar 6.5 merupakan hasil dari pengujian konstanta *gamma* yaitu tingkat akurasi terhadap pengaruh nilai konstanta γ (*gamma)*. Pada skenario ini hasil akurasi menunjukkan akurasi paling optimal ditunjukkan konstanta γ (*gamma*) yang memiliki nilai terbesar yaitu 0.05 dengan akurasi 93.86%. nilai parameter γ (*gamma)* berpengaruh terhadap proses learning atau pembelajaran dari sebuah sistem dan nilai konstanta γ (*gamma*) berpengaruh terhadap iterasi, dengan iterasi maksimum 10.000 dengan nilai konstanta 0.05 proses berhenti di iterasi ke-4.630, dengan semakin kecilnya nilai konstanta γ (*gamma*) maka cenderung tidak stabil dan akan mencapai iterasi maksimum. Banyaknya iterasi juga berpengaruh pada perubahan nilai α (alpha) yang baru, nilai α (alpha) akan mempengaruhi *support vector* yang dihasilkan oleh iterasi pada proses *sequential training.* nilai konstanta γ (*gamma*) sebaiknya $0 < \gamma < 2$ untuk mencapai proses *learning* atau pembelajaran yang bagus.

6.5 Pengujian Tingkat Akurasi Terhadap Nilai *Epsilon*

Pengujian tingkat akurasi terhadap nilai ε (*epsilon)* merupakan parameter *Sequential Training* SVM, dan akan menjelaskan skenario pengujian dan analisa pengujian.

6.5.1 Skenario Pengujian Nilai *Epsilon*

Pengujian nilai ε (*epsilon)* ini merupakan *Sequential Training* SVM untuk melihat skenario mana yang mendapatkan nilai terbaik ε (*epsilon)*, adapun skenario ε (*epsilon)* yaitu 0.000000001, 0.00000001, 0.0000001, 0.000001, 0.00001, 0.0001, 0.001, 0.01, 0.1, dan 1, adapun parameter lainnya terdiri dari λ = 0.1, konstanta γ (*gamma*) = 0.05, *itermax* = 100, *C* = 1, rasio perbandingan 90%:10% dan nilai *d*=2. Hasil pengujian dari skenario ditunjukkan pada Tabel 6.5.

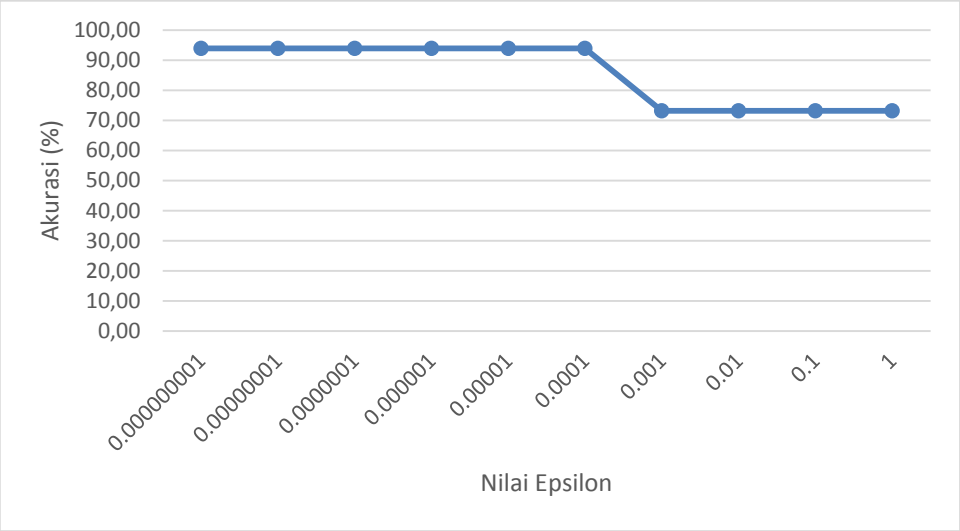
Tabel 6.5 Hasil Pengujian Nilai ε (*epsilon)*

Epsilon	Percobaan Ke-*i*					Rata-rata
	1	2	3	4	5	
0.000000001	90,00	93,10	100,00	93,10	93,10	93,86
0.00000001	90,00	93,10	100,00	93,10	93,10	93,86
0.0000001	90,00	93,10	100,00	93,10	93,10	93,86
0.000001	90,00	93,10	100,00	93,10	93,10	93,86
0.00001	90,00	93,10	100,00	93,10	93,10	93,86
0.0001	90,00	93,10	100,00	93,10	93,10	93,86
0.001	72,00	72,00	72,00	72,00	77,77	73,15
0.01	72,00	72,00	72,00	72,00	77,77	73,15
0.1	72,00	72,00	72,00	72,00	77,77	73,15
1	72,00	72,00	72,00	72,00	77,77	73,15

Tabel 6.5 merupakan hasil dari skenario pengujian nilai ε (*epsilon)* yang merupakan *Sequential Training SVM.*

6.5.2 Analisa Pengujian Nilai *Epsilon*

Berdasarkan grafik pada Gambar 6.6 diketahui bahwa rata-rata tertinggi dari hasil akurasi nilai *epsilon* sebesar 93.86% dan akurasi terbaik 100%. sehingga dari rata-rata tersebut nilai terbaik ε (*epsilon)* adalah 0.000000001, 0.00000001, 0.0000001, 0.000001, 0.00001, dan 0.0001 dari λ = 0.1, konstanta γ (*gamma*) = 0.05, *itermax* = 100, *C* = 1, rasio perbandingan 90%:10% dan nilai *d*=2. Nanti nilai *epsilon* digunakan untuk pengujian nilai iterasi maksimum. Hasil dari pengujian nilai ε (*epsilon)* ditunjukkan pada Gambar 6.6.



Gambar 6.6 Grafik Tingkat Akurasi Hasil Pengujian nilai *Epsilon*

Gambar 6.6 menunjukkan grafik dari hasil pengujian nilai ε (*epsilon)*, yang mana rata-rata sebesar 93.86 %, dan akurasi tertinggi sebesar 100%. nilai ε (*epsilon)* dipengaruhi oleh nilai konstanta γ (*gamma*) yang digunakan, Hal tersebut

diketahui bahwa nilai dari parameter nilai ε (*epsilon)* berpengaruh pada akurasi, semakin besar nilai ε (*epsilon)* maka iterasi yang dihasilkan semakin sedikit, dan semakin besar nilai ε (*epsilon)* maka iterasinya semakin banyak. Nilai ε (*epsilon)* juga berpengaruh terhadap proses *learning*, jika nilai ε (*epsilon)* semakin kecil maka proses *learning* akan berlangsung lama, sehingga akan menghasilkan data yang konvergen serta nilai α (*alpha*) dan *support vector* akan semakin optimal. Iterasi berhenti pada nilai ε (*epsilon)* 0.0000001, 0.000001, 0.00001, 0.0001, 0.001, 0.01, 0.1, dan 1, karena telah mencapai nilai konvergen $Max(|\delta\alpha_i|) < \varepsilon$, jika nilai α *(alpha)* dan *support vector* tidak optimal maka akan berpengaruh terhadap nilai akurasi.

6.6 Pengujian Tingkat Akurasi Terhadap Nilai Iterasi Maksimum

Pengujian tingkat akurasi terhadap nilai iterasi maksimum akan menjelaskan skenario pengujian dan hasil pengujian.

6.6.1 Skenario Pengujian Nilai Iterasi Maksimum

Pengujian terhadap parameter *Sequential Training* SVM dilakukan untuk mengetahui skenario terbaik dari skenario pengujian, dan untuk iterasi maksimum ini ada beberapa iterasi yang dilakukan untuk mendapatkan nilai akurasi terbaik yaitu terdiri dari 100, 500, 1000, 2000, 3000, 5000, 10000, 25000, 50000, dan 100000, nilai parameter lainnya yaitu λ = 0.1, konstanta γ (*gamma*) = 0.5, ε = 0.0001, C = 1, rasio perbandingan 90%:10% dan nilai *d*=2. Adapun hasil dari skenario pengujian ditunjukkan pada Tabel 6.6.

Tabel 6.6 Skenario Pengujian Nilai Iterasi Maksimum

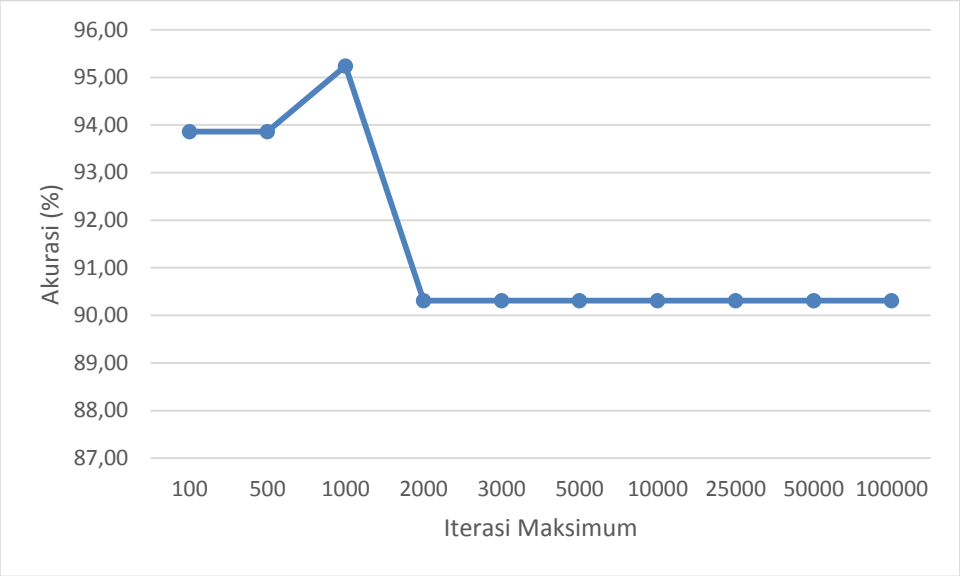
Iterasi	Percobaan Ke-*i*					Rata-rata
	1	2	3	4	5	
100	90,00	93,10	100,00	93,10	93,10	93,86
500	90,00	93,10	100,00	93,10	93,10	93,86
1000	90,00	93,10	100,00	100,00	93,10	95,24
2000	90,00	85,71	82,75	100,00	93,10	90,31
3000	90,00	85,71	82,75	100,00	93,10	90,31
5000	90,00	85,71	82,75	100,00	93,10	90,31
10000	90,00	85,71	82,75	100,00	93,10	90,31
25000	90,00	85,71	82,75	100,00	93,10	90,31
50000	90,00	85,71	82,75	100,00	93,10	90,31
100000	90,00	85,71	82,75	100,00	93,10	90,31

Tabel 6.6 menjelaskan hasil dari pengujian yang dilakukan pada *Sequential Training* terhadap parameter iterasi.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

6.6.2 Analisa Pengujian Nilai Iterasi Maksimum

Berdasarkan grafik pada Gambar 6.7 diketahui bahwa nilai rata-rata tertinggi yaitu 95.24% dan akurasi tertinggi sebesar 100%. dari rata-rata terbaik tersebut iterasinya sebanyak 1000, sehingga iterasinya akan digunakan pada perhitungan *C* (*Complexity)* dengan parameter λ = 0.1, konstanta γ (*gamma*) = 0.05, ε = 0.0001, *C* = 1, rasio perbandingan 90%:10% dan nilai *d*=2, sehingga didapatkan hasilnya berupa grafik yang ditunjukkan pada Gambar 6.7.



Gambar 6.7 Grafik Tingkat Akurasi Hasil Pengujian Iterasi Maksimum

Analisis dilakukan pada pengujian ini yaitu nilai iterasi maksimum terhadap nilai akurasi, pada skenario pengujian menunjukkan hasil akurasi rata-rata yang memiliki nilai paling besar yaitu iterasi maksimum sebanyak 1000. Nilai iterasi berhenti jika sudah memenuhi syarat konvergen yaitu $Max(|\delta\alpha_i|) < \varepsilon$. Nilai ε (*epsilon)* = 0.0001 dan pada iterasi maksimum 3.000, iterasi berhenti pada level 2, dan pada iterasi maksimum lebih dari 3.000 maka iterasi berhenti pada level 1 dan 2 pada iterasi ke 3.298 dan 2.546, hal tersebut terjadi dikarenakan nilai α *(alpha)* telah mencapai konvergen. Konvergen dapat diartikan sebagai tingkat perubahan dari nilai α *(alpha)*, dan pada pengujian ini nilai iterasi berhenti hanya akan berpengaruh pada perubahan α *(alpha)* dan nilai *b* (*bias*) dari proses pelatihan atau *training*.

6.7 Pengujian Untuk Mengetahui Pegaruh Nilai *C* Terhadap Akurasi

Pengujian untuk mengetahui pengaruh nilai *C* terhadap akurasi akan menjelaskan skenario pengujian dan analisa pengujian.

6.7.1 Skenario Pengujian Nilai *C* (*Complexity*)

Pengujian terhadap parameter *Sequential Training* SVM dilakukan untuk melihat pada skenario mana yang mendapatkan nilai tertinggi unutk nilai *C*. Adapun skenario atau parameter nilai *C* yaitu 1, 10, 20, 30, 40, 50, 60,70, 80, dan 90. Dan parameter lainnya yaitu λ = 0.1, konstanta γ (*gamma*) = 0.05, ε = 0.0001,

Iterasi maksimum = 1000, rasio perbandingan 90%:10% dan nilai *d*=2. Adapun skenario pengujian ditunjukkan pada Tabel 6.7.

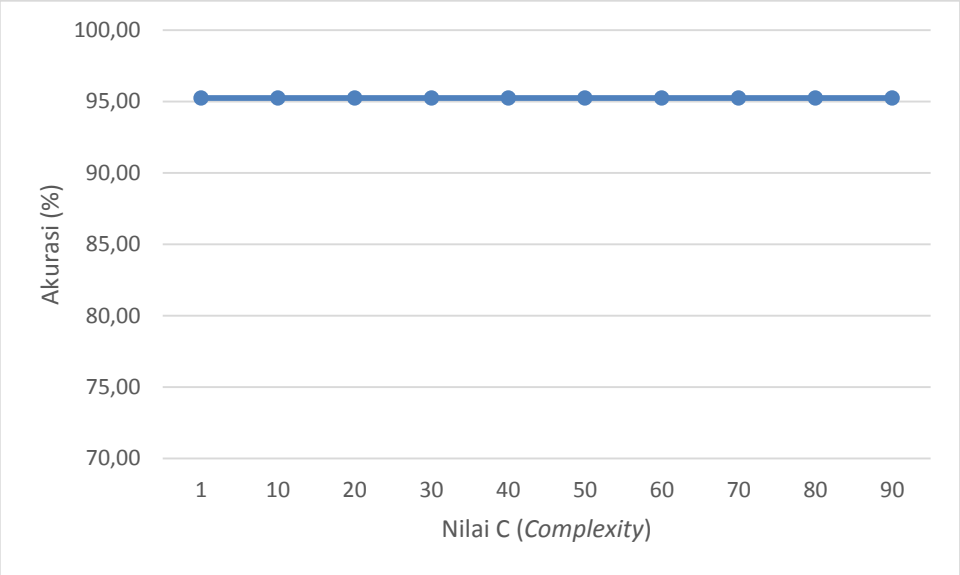
Tabel 6.7 Skenario Pengjian Nilai *C* (*Complexity*)

C	Percobaan Ke-*i*					Rata-rata
	1	2	3	4	5	
1	90,00	93,10	100,00	100,00	93,10	95,24
10	90,00	93,10	100,00	100,00	93,10	95,24
20	90,00	93,10	100,00	100,00	93,10	95,24
30	90,00	93,10	100,00	100,00	93,10	95,24
40	90,00	93,10	100,00	100,00	93,10	95,24
50	90,00	93,10	100,00	100,00	93,10	95,24
60	90,00	93,10	100,00	100,00	93,10	95,24
70	90,00	93,10	100,00	100,00	93,10	95,24
80	90,00	93,10	100,00	100,00	93,10	95,24
90	90,00	93,10	100,00	100,00	93,10	95,24

Tabel 6.7 menunjukkan skenario pengujian *Sequential Training* untuk nilai *Complexity*.

6.7.2 Analisa Pengujian Nilai *C* (*Complexity*)

Berdasarkan grafik pada Gambar 6.8 diketahui bahwa rata-rata tertinggi yaitu 95.24% dan nilai akurasi tertinggi sebesar 100%, dengan parameter yaitu yaitu λ = 0.1, konstanta γ (*gamma*) = 0.1, ε = 0.0001, *Iterasi* Maksimum = 10000, rasio perbandingan 90%:10% dan nilai *d*=2. Hasil akurasi pengujian nilai *C* (*Complexity)* dijelaskan pada Gambar 6.8.



Gambar 6.8 Grafik Tingkat Akurasi Hasil Pengujian Nilai *Complexity*

Gambar 6.8 menjelaskan hasil dari akurasi pengujian nilai *complexity*. Pada skenario pengujian ini akurasi dengan rata-rata tertinggi 95.24% .Tujuan adanya

nilai *C* (*Complexity)* untuk meminimalkan nilai *error* . Pada saat nilai *C* (*Complexity)* mendekati nilai 0 maka lebar margin pada bidang pemisah (*hyperplane)* menjadi maksimum, hal tersebut disebabkan karena nilai *C* (*Complexity)* digunakan untuk memperkecil nilai *error* pada proses *training* atau pelatihan yaitu pada perhitungan nilai *w* (*weight*) dan nilai *b* (*bias*), semakin kecil nilai C maka error yang terjadi pada α_i akan semakin kecil begitu pula dengan sebaliknya. Selain itu nilai *C* (*Complexity)* > 0 relatif penting untuk memaksimumkan *margin* dan meminimumkan jumlah *slack* (C.Corttes dan V.Vapnik, 1995).

6.8 Analisis Tingkat Akurasi

Analisis hasil dari pengujian terhadap waktu eksekusi didapatkan bahwa rata-rata sistem seleksi dan rekomendasi penerima beasiswa dilembaga GNOTA Kediri dengan implementasi metode SVM dan TOPSIS yaitu 19,16 detik, waktu tercepat yaitu 13,39 detik dan waktu terlama 24,07. Adapun hasil dari analisis terhadap beberapa paremeter seperti λ(*Lambda*), konstanta γ (*gamma*), ε (*Epsilon*), *Iterasi* Maksimum, dan *C* (*Complexity)* menunjukkan jika pemilihan beberapa parameter harus dilakukan dengan tepat, dikarenakan akan berpengaruh terhadap nilai akurasi pada sistem, dengan demikian diperlukan pengujian terhadap beberapa parameter untuk dapat menghasilkan nilai akurasi secara optimal. Berdasarkan hasil pengujian nilai akurasi terbaik pada pengujian rasio perbandingan, akurasi terbaik pada rasio 90%:10% dengan rata-rata akurasi sebesar 88,83%, untuk parameter λ (*Lambda*) adalah 0.1, konstanta nilai parameter γ (*gamma*) adalah 0.05, parameter ε (*Epsilon*) adalah 0.0001, banyaknya iterasi maksimum adalah 1000, dan nilai parameter *C* (*Complexity)* adalah 1, sehingga diperoleh nilai akurasi sistem terbaik yaitu 100% dan rata-rata akurasi terbaik sebesar 95,24%.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

BAB 7 PENUTUP

Bab ini akan membahas kesimpulan dan saran dari seluruh proses penelitian sistem pendukung keputusan Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri dengan Implementasi Metode SVM dan TOPSIS.

7.1 Kesimpulan

Bedasarkan hasil penelitian dan pengujian yang dilakukan, maka dapat diambil kesimpulan sebagai berikut.

1. Kecepatan rata-rata keseluruhan proses yang ada pada sistem seleksi dan rekomendasi penerimaan beasiswa dengan sistem dari implementasi metode SVM dan TOPSIS dalam proses seleksi beasiswa di lembaga GNOTA Kediri dengan menggunakan pengujian dari rasio perbandingan yaitu 19.16 detik, waktu tercepat yaitu 13,39 dan waktu terlama yaitu 24,07.
2. Tingkat akurasi dari seleksi dan rekomendasi penerimaan beasiswa GNOTA kediri dengan menggunakan rasio perbandingan data *training* dan data *testing* 90%:10% memiliki akurasi rata-rata sebesar 88.83% dan pengujian berdasarkan parameter terbaik dari proses *sequential training* SVM yaitu λ (*Lambda*) = 0.1, konstanta γ (*gamma*) = 0.05, ε = 0.0001, *Iterasi* Maksimum = 1000, rasio perbandingan 90%:10% dan nilai *d*=2, *C* (*Complexity)* = 1. Sehingga diperoleh akurasi terbaik sebesar 100% dan rata-rata akurasi terbaik sebesar 95.24%.

7.2 Saran

Saran yang dapat diberikan untuk pengembangan penelitian selanjutnya antara lain:

1. Penelitian selanjutnya agar ditambahkan data, karena dengan bertambahnya data maka hasil yang dihasilkan akan optimal.
2. Penelitian selanjutnya mencoba dengan kernel selain *polynomial*, yaitu seperti kernel RBF, kernel *sigmoid,* dan kernel *quadratic* untuk mengetahui kernel mana yang sesuai dengan data yang digunakan.
3. Pengembangan selanjutnya diharapkan membangun sistem berbasis *web*, sehingga proses seleksi dapat diakses oleh para masyarakat yang ingin mendaftar, dan memudahkan seleksi beasiswa di lembaga GNOTA Kediri.

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

DAFTAR PUSTAKA

Basjir, Mochammad, Spriyanto, Hari dan Suef, Mokh., 2014. *Pengembangan model Penentuan Prioritas Perbaikan Terhadap Mode Kegagalan Komponen dengan Metodologi FMEA, Fuzzy dan TOPSIS yang terintegrasi*. InstitutTeknologi Sepuluh Nopember, Jurusan Teknik Industri.

C. Cortes dan V. Vapnik, 1995, *Machine Learning*, *Support Vector Networks,* Vol.20, Hal 273-297.

Damayanti Fitri, Arifin Agus Zainal, Soelaiman Rully, 2010, Jurnal Ilmiah Kursor, *Pengenalan Citra Wajah Menggunakan Metode Two-Dimensional Linier Discriminant Analysis dan Support Vector Machine*, Vol. 5, No 3, Hal 147-156.

Ding, Ji-Feng, 2011. *An Integrated Fuzzy TOPSIS Method for Ranking Alternatifs and Its Application*. Journal of Marine Science and Technology, Vol. 19, No. 4 pp. 324

GNOTA: Apa itu GN-OTA. Diakses pada 24 September 2015 dari http://www.gn-ota.or.id/index.php?controller=page&action=view&page=tentanggnota&subpage=tentang-kami&lang=id .

Hermaduanti, N. Dan Kusuma dewi, S. 2008. *Sistem Pendukung Keputusan Berbasis Sms Untuk Menentukan Status Gizi Dengan Metode K-Nearest Neighbor.* Seminar Nasional Aplikasi Teknologi Informasi.

Jawa Pos: *Beasiswa jadi objek PPh*. Diakses pada 24 September 2015 dari http://www.infopajak.com/berita/310108jps.htm .

Kartal Hasan Basri, Cebi Ferhan., 2013, *Internasional Journal of Machine Learning and Computing, Support Vector Machine for Multi-Attribute ABC Analysis*, Vol 3, No 1, Hal. 154-157.

Kusuma, Mukhtar Atmaja., 2010. *Sistem Pendukung Keputusan Untuk Pemilihan Bahan Obat Alternatif Dengan Metode AHP dan TOPSIS*. Sekolah Tinggi Manajemen Informatika dan Teknik Komputer. Surabaya

Lestari, S., & Priyodiprodjo., W. 2011. *Implementasi Metode Fuzzy TOPSIS untuk Seleksi Penerimaan Karyawan*. IJCCS. Vol 5. No.2.

Mujiasih, Subekti., 2011, *Jurnal Meteorologi dan Geofisika, Pemanfaatan Data Mining untuk Perkiraan Cuaca*, Vol 12, No 2, Hal 189-195.

Mohanty, S. dan Bebartta, H. N. D., 2011. *Performance Comparison of SVM dan K-NN for Oriya Character Recognition*. (IJACSA) International Journal of Advanced Computer Science and Applications, Special Issue on Image Processing and Analysis, Hal 112-116.

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, No 1, Hal D-147 – D-152.

Onüt. S, S. S., 2008. *Transhipment site selection using the AHP and Topsis approaches under fuzzy environment*. Waste management, 28(9), 1552-9.

Pemkabkediri. 14 Desember 2014. *Bantuan GN-OTA Menuju Kesuksesan Siswa,*(Online), (kedirikab.go.id, diakses 20 September 2015).

Purnami, S. W., dan Embong, A. 2008. *Smooth Support Vector Machine For Breast Cancer Classification*. ICMSA08, Banda Aceh, Indonesia.

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.

Sahar, A.M dan Elsayad,M, Allaa., 2010. *Predicting the Severity of Breast Masses with Data Mining Methods*. Computer and System Department , Electronic Reasearch Institute Dokki, 12622/ Cairo,Egypt.

Sorana,D.Bolboaca dan Jantschi,Lorentz., 2006. *Pearson versus Spearman, Kendall's Tau Correlation Analysis on Structure-Activity Relationships of Biologic Active Compounds.*Leonardo Journal of Sciences.Vol 9, Hal 179-200.

Sprague, R.H dan Watson H. J., 1993. *Decision Support Systems: Putting Theory Into Practice*. Englewood Clifts, N. J., Prentice Hall.

Sugiyono, 2008. *Metode Penelitian Kuantitatif Kualitatif dan R&D*. Alfabeta.Bandung.

Turban, E., Aronson, J.E,. Liang, T.P., 2005. *Decision Support Systems and Intellegent Systems*. New Jersey: Pearson Education.

Vijayakumar S, Wu S. Proc. International Conference on Soft Computing (SOCO'99'), *Sequential Support Vector Classifier and Regression, Genoa, Italy, pp.610-619,1999.*

Yang. T,H,C., 2007. *Multiple-Attribute Decision Making: An Introduction. In: Thousand Oaks*. Sage Publications, Inc.

Yang Yi, Du Quishi, Zhao Jinying., 2010, *Logistic System and Intelegent Management, The Application of Site Selection Based on AHP-SVM in 500KV Substations*, Vol 2, Hal 1225-1229.

Yung-Ming Li. Cheng-Yang Lai. Chien-Pang Kao., 2010. *Building a qualitative recruitment system via SVM with MCDM approach*. Springer Science Business.

LAMPIRAN

Lampiran A Form pendaftaran

Instrumen Penerima Beasiswa L-GNOTA Kabupaten Kediri

NO :

1. Biodata Penerima Beasiswa

Nama Siswa	:	
Jenis Kelamin	:	☐ Laki-laki ☐ 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

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Anak yang masih menjadi tanggungan	:	Jiwa
Tagihan PBB Tahun Sekarang	:	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	

Kediri, ,20..

Responden Surveyor

....

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Lampiran B Data Pendaftar Beasiswa

NO	NO PESERTA	NAMA	TOTAL PENGHASILAN	JUMLAH ANAK	PEKERJAAN	DAYA LISTRIK	BIAYA LISTRIK	JUMLAH RUMAH	STATUS RUMAH	STATUS DINDING	STATUS LANTAI	PBB	JML MOTOR	JML MOBIL	PROGRAM BEASISWA	KLASIFIKASI
1	263	NERY DWI EVITHELISARI	1.300.000	1	Kuli Bangunan	450 W	30.000	1	Rumah Sendiri	Tembok	Keramik	113.147	1	0	LAINNYA	A
2	264	MUHAMMAD MUKAINUL H	1.000.000	1	Pracangan	450 W	31.614	1	Rumah Sendiri	Tembok	Semen	118.536	1	0	BIDIK MISI	A
3	265	NOVA WANDA KARTIKA	900.000	2	Supir	450 W	31.000	1	Numpang	Tembok	Keramik	114.400	1	0	LAINNYA	A
4	266	LAILIL AFIF AGUSTINA	1.200.000	1	Buruh Tani	450 W	59.200	1	Rumah Sendiri	Tembok	Semen	53.050	1	0	BIDIK MISI	A
5	267	YUSUF	1.000.000	1	Makelar	900 W	100.000	1	Numpang	Tembok	Keramik	83.000	1	0	LAINNYA	A
6	268	FARADHYBA RIZKY R	1.112.000	1	karyawan BPR	450 W	49.000	1	Rumah Sendiri	Tembok	Keramik	70.319	2	0	BIDIK MISI	A
7	269	MOCH KRESNA ALVIAN P	1.650.000	3	Supir	450 W	58.600	1	Sewa	Tembok	Semen	40.435	2	0	BIDIK MISI	A
8	270	TOFA KUMALA SARI	900.000	3	Becak	900 W	125.000	1	Rumah Sendiri	Tembok	Keramik	233.000	1	0	BIDIK MISI	A
9	271	HIDAYATURROHMAN	750.000	5	Buruh Tani	450 W	14.000	1	Rumah Sendiri	Tembok	Semen	64.089	1	0	BIDIK MISI	A
10	272	MOCHMMAD ABDUL GHOFUR	1.050.000	2	Buruh Pabrik	900 W	51.000	1	Rumah Sendiri	Tembok	Keramik	39.430	2	0	BIDIK MISI	A
11	273	MIA AGUSTINA	1.000.000	2	Buruh ternak	450 W	25.000	1	Rumah Sendiri	Tembok	Tanah	40.000	1	0	BIDIK MISI	A
12	274	OKKI PUTRA PERDANA	750.000	1	TIDAK BEKERJA	900 W	126.000	1	Numpang	Tembok	Keramik	19.200	1	0	BIDIK MISI	A
13	152	NOVA RINELDA A F	1.000.000	2	Buruh Tani	450 W	30.000	1	Rumah Sendiri	Tembok	Tanah	22.165	1	0	LAINNYA	A
14	153	YUNIS ROISATUL M	500.000	2	Buruh	450 W	16.363	1	Rumah Sendiri	Tembok	Semen	156.737	2	0	BIDIK MISI	A
15	156	SEKAR PRATIWI S P	500.000	3	Pedagang	450 W	50.580	1	Numpang	Tembok	Semen	38.586	0	0	BIDIK MISI	A
16	157	MUMDIYAH	500.000	1	Buruh Tani	900 W	53.530	1	Rumah Sendiri	Tembok	Keramik	34.739	0	0	LAINNYA	A
17	159	MOCH FAIZIN	1.000.000	2	Buruh	450 W	16.000	1	Rumah Sendiri	Tembok	Semen	256.295	1	0	BIDIK MISI	A
18	160	DWI SAKINAH AINUR R	1.000.000	3	Buruh	900 W	67.000	1	Rumah Sendiri	Tembok	Tegel	195.820	1	0	LAINNYA	A

19	162	GEFI INDAH P A	1.000.000	2	Karyawan Toko	450 W	39.000	1	Rumah Sendiri	Tembok	Semen	41.180	0	0	LAINNYA	A
20	163	AYUNING DYAH SUSANTI	300.000	2	tidak bekerja	450 W	32.500	1	Rumah Sendiri	Tembok	Tegel	100.507	0	0	LAINNYA	A
21	165	BAGUS PRIBADI	1.500.000	4	Jualan Bakso	900 W	50.000	1	Rumah Sendiri	Tembok	Semen	41.202	2	0	BIDIK MISI	A
22	166	WAHYA LYN FIZA	600.000	1	Buruh	900 W	40.000	1	Rumah Sendiri	Tembok	Keramik	100.708	1	0	BIDIK MISI	A
23	167	AULIA NUR S	0	4	tidak bekerja	900 W	40.000	1	Rumah Sendiri	Tembok	Semen	44.480	0	0	BIDIK MISI	A
24	168	NAILIL HUSNIAH	200.000	3	tidak bekerja	450 W	33.614	1	Rumah Sendiri	Tembok	Semen	112.000	1	0	BIDIK MISI	A
25	169	YUSRON AMAN N	800.000	2	Jualan grabah	450 W	30.000	1	Rumah Sendiri	Papan Kayu	Semen	35.432	1	0	BIDIK MISI	A
26	170	DINDA MAULYDIA I	1.000.000	3	KSP	900 W	38.000	1	Kontrak	Tembok	Semen	32.042	0	0	BIDIK MISI	A
27	173	SETYA DWI A	1.000.000	1	Kuli Bangunan	450 W	41.000	1	Rumah Sendiri	Tembok	Semen	132.172	2	0	LAINNYA	A
28	194	MOHAMMAD KHOIRUZ ZAAD	1.000.000	2	Buruh	900 W	60.000	1	Rumah Sendiri	Tembok	Keramik	47.312	1	0	LAINNYA	A
29	197	SAHRUL MUNIR	0	2	Kuli Pikul	450 W	36.337	1	Rumah Sendiri	Tembok	Keramik	56.734	2	0	LAINNYA	A
30	199	LAILATUL HIDAYAH	1.500.000	2	Swasta	900 W	50.000	1	Rumah Sendiri	Tembok	Keramik	5.000	1	0	BIDIK MISI	A
31	203	KHOIRUL LIHAYATI	825.000	2	Buruh Tani	450 W	24.306	1	Rumah Sendiri	Tembok	Keramik	46.844	1	0	BIDIK MISI	A
32	205	NUR HALIMAH	0	1	Buruh Tani	450 W	114.200	1	Rumah Sendiri	Tembok	Keramik	182.100	1	0	BIDIK MISI	A
33	206	MAULANA MUJTABA	2.000.000	3	Buruh Tani	450 W	39.059	1	Rumah Sendiri	Papan Kayu	Semen	37.800	1	0	BIDIK MISI	A
34	211	WULAN PUJI UTAMI	60.000	1	tidak bekerja	450 W	14.606	1	Numpang	Tembok	Semen	25.536	1	0	LAINNYA	A
35	214	RIRIN FAULINA	600.000	1	Tidak Bekerja	450 W	29.500	1	Rumah Sendiri	Tembok	Semen	30.720	1	0	BIDIK MISI	A
36	216	AHMAD IRFAN FADRI	800.000	1	Buruh Tani	450 W	26.000	1	Rumah Sendiri	Tembok	Tegel	57.808	1	0	LAINNYA	A
37	217	PUTRI NUR M	800.000	3	Tidak Bekerja	450 W	29.300	1	Rumah Sendiri	Tembok	Keramik	57.242	2	0	BIDIK MISI	A
38	218	MACHICA ROUFUN N	800.000	4	Makelar	450 W	90.000	1	Numpang	Tembok	Semen	76.176	1	0	LAINNYA	A

39	220	PRIMA NUR ZAKI	1.500.000	3	Jualan Soto	450 W	56.700	1	Sewa	Tembok	Semen	18.560	3	0	BIDIK MISI	A
40	221	IKA VIKNI NAWANG RISMA	1.200.000	1	Kuli Bangunan	450 W	23.456	1	Rumah Sendiri	Tembok	Semen	75.806	1	0	LAINNYA	A
41	224	AGUS ZAINAL ARIFIN	900.000	2	Supir	450 W	60.000	1	Rumah Sendiri	Tembok	Keramik	46.437	2	0	BIDIK MISI	A
42	225	EVA DIAH AYUNINGTYAS	1.150.000	1	Buruh Tani	450 W	33.248	1	Rumah Sendiri	Tembok	Semen	100.000	1	0	BIDIK MISI	A
43	227	ASNA NURHAYATI	500.000	3	Buruh Tani	450 W	21.000	1	Rumah Sendiri	Tembok	Semen	9.720	1	0	LAINNYA	A
44	228	MUHAMMAD ADIB SULAIMAN	1.300.000	3	Buruh Tani	450 W	19.200	1	Numpang	Papan Kayu	Semen	153.152	1	0	LAINNYA	A
45	230	SHEYLA CHABIBAH	700.000	3	Makelar	900 W	96.000	1	Numpang	Tembok	Keramik	45.000	1	0	BIDIK MISI	A
46	232	HERWIN AGUSTYANILLA SARI	1.000.000	2	Buruh Tani	450 W	63.562	1	Rumah Sendiri	Bambu	Keramik	45.386	2	0	BIDIK MISI	A
47	236	MUHAMMAD ALI M	1.300.000	3	Buruh Tani	450 W	23.000	1	Rumah Sendiri	Tembok	Tegel	190.980	1	0	BIDIK MISI	A
48	241	SYAMSUL ARI WIDODO	1.800.000	1	Kuli Bangunan	450 W	41.000	1	Rumah Sendiri	Tembok	Keramik	79.740	1	0	LAINNYA	A
49	242	ERNA RAHAYU	1.000.000	2	Petani	450 W	47.060	1	Rumah Sendiri	Tembok	Keramik	189.712	2	0	BIDIK MISI	A
50	243	LAILY FITRIA NINGTYAS	950.000	1	Buruh Tani	450 W	23.000	1	Rumah Sendiri	Tembok	Keramik	68.090	2	0	BIDIK MISI	A
51	245	GUNTUR TRI WASONO	620.000	2	tidak bekerja	900 W	53.000	1	Rumah Sendiri	Tembok	Keramik	32.558	1	0	BIDIK MISI	A
52	246	WINDA FIRDAYANTI	1.050.000	1	Tukang Bangunan	450 W	27.000	1	Rumah Sendiri	Tembok	Semen	25.088	1	0	BIDIK MISI	A
53	247	LINDA PURNAMASARI	700.000	1	karyawan BPR	450 W	21.500	1	Rumah Sendiri	Bambu	Keramik	126.106	1	0	LAINNYA	A
54	248	AWIATUL MUTHOWIYAH	1.000.000	2	Buruh Tani	450 W	27.000	1	Rumah Sendiri	Tembok	Tegel	126.200	1	0	BIDIK MISI	A
55	249	RAHMAT HIDAYAH	1.300.000	1	Jualan sayur	900 W	30.000	1	Rumah Sendiri	Tembok	Keramik	25.250	1	0	BIDIK MISI	A
56	250	RIKA RAHAYU	1.250.000	1	Buruh Tani	450 W	29.400	1	Rumah Sendiri	Tembok	Tegel	621.224	1	0	BIDIK MISI	A
57	251	ALFIRA SARI	1.400.000	1	Buruh	450 W	28.000	1	Rumah Sendiri	Tembok	Semen	83.290	2	0	BIDIK MISI	A
58	253	DEVIANA INDAH PERMATA N	1.300.000	3	Karwayan Swasta	450 W	69.000	1	Numpang	Bambu	Keramik	17.920	1	0	LAINNYA	A

59	254	ABDUL LATIF	300.000	1	tidak bekerja	450 W	45.000	1	Rumah Sendiri	Tembok	Keramik	92.408	0	0	LAINNYA	A
60	256	DIFA ZUHROTUL FIKIRIYAH	600.000	2	Buruh Tani	900 W	36.000	1	Rumah Sendiri	Tembok	Keramik	54.586	1	0	LAINNYA	A
61	257	BAHDYA RICHANA PUTRI	500.000	3	Buruh Tani	900 W	39.000	1	Rumah Sendiri	Tembok	Keramik	68.206	2	0	LAINNYA	A
62	258	LISA L SHOLIKHAH	900.000	3	Kuli Listrik	900 W	50.000	1	Rumah Sendiri	Tembok	Keramik	31.920	1	0	LAINNYA	A
63	174	SITI KHIRROTUL A'YUN	500.000	4	Buruh Tani	450 W	25.991	1	Rumah Sendiri	Tembok	Keramik	27.656	0	0	BIDIK MISI	B
64	177	INNA FAUZIAH	1.000.000	3	Buruh	900 W	120.800	1	Rumah Sendiri	Tembok	Keramik	16.487	2	0	LAINNYA	B
65	178	NELLY NUR H	500.000	1	Karwayan Swasta	450 W	37.159	1	Rumah Sendiri	Tembok	Keramik	10.320	2	0	LAINNYA	B
66	179	ZAHRO NUR HELIZA	1.500.000	2	Buruh Tani	450 W	43.693	1	Rumah Sendiri	Tembok	Keramik	155.384	2	0	LAINNYA	B
67	180	M IQBAL BAIHAQI	500.000	2	Konstruksi	450 W	50.000	1	Rumah Sendiri	Tembok	Keramik	118.164	1	0	LAINNYA	B
68	182	INDIKA AZHARY	1.000.000	4	tidak bekerja	900 W	83.912	1	Rumah Sendiri	Tembok	Keramik	121.000	1	0	BIDIK MISI	B
69	183	IIK FITRIATUS SANAH	500.000	1	Swasta	450 W	50.000	1	Rumah Sendiri	Tembok	Keramik	5.000	1	0	LAINNYA	B
70	185	NILA PUTRI ANGGRAINI	1.000.000	2	Konstruksi	900 W	67.000	1	Rumah Sendiri	Tembok	Keramik	43.873	3	0	LAINNYA	B
71	187	MUHAMMAD DAMANURI	500.000	2	tidak bekerja	450 W	38.792	1	Rumah Sendiri	Tembok	Semen	14.471	1	0	BIDIK MISI	B
72	188	INTAN ARIKA SARI	1.500.000	1	Pedagang	450 W	68.000	1	Rumah Sendiri	Tembok	Keramik	46.264	1	0	BIDIK MISI	B
73	189	DEWI EKA MULIANA	500.000	2	Pedagang	450 W	27.625	1	Rumah Sendiri	Tembok	Keramik	50.945	2	0	LAINNYA	B
74	192	ADELIA DWI SAPUTRI	1.500.000	2	Supir	450 W	65.700	1	Rumah Sendiri	Tembok	Keramik	48.082	1	0	BIDIK MISI	B
75	132	EMILIA SUSANTI	1.000.000	2	Wiraswasta	450 W	60.000	1	Rumah Sendiri	Tembok	Keramik	82.000	1	0	LAINNYA	B
76	133	RATIH DYAH T R	1.500.000	1	tidak bekerja	450 W	50.000	1	Rumah Sendiri	Tembok	Keramik	50.000	1	0	LAINNYA	B
77	135	RAHMAD INDRIANTO	300.000	1	tidak bekerja	450 W	40.000	1	Rumah Sendiri	Tembok	Keramik	30.000	1	0	BIDIK MISI	B

78	137	FITRI F	400.000	2	Buruh Tani	450 W	21.000	0	Rumah Sendiri	Tembok	Semen	98.000	0	0	LAINNYA	B
79	138	SITI KUMALA DEWI	1.000.000	1	tidak bekerja	450 W	50.000	1	Rumah Sendiri	Tembok	Keramik	29.000	1	0	LAINNYA	B
80	139	NOER ZARINA	1.000.000	3	Buruh Tani	450 W	30.000	1	Rumah Sendiri	Tembok	Keramik	40.000	1	0	LAINNYA	B
81	140	ALVIN NAFIAH	500.000	3	Buruh Tani	450 W	28.000	1	Rumah Sendiri	Tembok	Keramik	18.000	1	0	BIDIK MISI	B
82	142	MUSTIKA ANGGI P	1.000.000	3	Wiraswasta	900 W	95.000	1	Rumah Sendiri	Tembok	Keramik	25.000	2	0	BIDIK MISI	B
83	144	DEWI MARSITOH	1.200.000	2	Pedagang Ayam	450 W	35.000	1	Numpang	Tembok	Keramik	70.000	1	0	BIDIK MISI	B
84	195	DEWI NABILA HIKMAH	1.000.000	3	Pedagang Tempe	450 W	37.426	1	Rumah Sendiri	Tembok	Keramik	230.000	2	0	LAINNYA	B
85	196	AISYAH NESTRIA AL-FALAH R	0	2	Karyawan Swasta	900 W	82.356	1	Numpang	Tembok	Keramik	254.892	2	0	BIDIK MISI	B
86	198	WIDYA OCTAVIANI RIDHO	2.000.000	3	Peternak	900 W	94.257	1	Rumah Sendiri	Tembok	Keramik	54.838	1	0	LAINNYA	B
87	200	INTAN PUTRI RAHAYU	600.000	1	tidak bekerja	900 W	75.488	1	Rumah Sendiri	Tembok	Tegel	30.336	3	0	BIDIK MISI	B
88	201	AMITORUL MA;RIFAH	2.250.000	1	Wiraswasta	900 W	151.430	1	Rumah Sendiri	Tembok	Keramik	29.500	0	0	BIDIK MISI	B
89	204	BAITI KHARISMA SANTI	5.000.000	1	tidak bekerja	900 W	130.739	1	Rumah Sendiri	Tembok	Keramik	73.200	1	0	BIDIK MISI	B
90	210	MILA GHURROTUL JANNAH	100.000	4	Pedagang Ikan	900 W	45.000	1	Rumah Sendiri	Tembok	Keramik	64.400	2	1	LAINNYA	B
91	215	HELLEN FINA R S D	500.000	3	Supir	900 W	40.588	1	Rumah Sendiri	Tembok	Semen	121.231	1	0	LAINNYA	B
92	226	ASNA REDAYANI	1.000.000	2	Kuli Batu	900 W	36.000	1	Rumah Sendiri	Tembok	Keramik	126.800	1	0	BIDIK MISI	B
93	229	NURFIKKA T F	1.500.000	2	Kuli Bangunan	450 W	14.742	1	Rumah Sendiri	Tembok	Keramik	90.825	2	0	LAINNYA	B
94	231	INDI DIAS C	1.000.000	2	Buruh Tani	450 W	30.000	1	Rumah Sendiri	Tembok	Keramik	83.574	1	0	LAINNYA	B
95	233	EFI JULIANA	2.500.000	2	Serabutan	450 W	65.000	1	Rumah Sendiri	Tembok	Keramik	27.645	2	0	LAINNYA	B
96	234	ITA DEWI FRANSISKA	1.250.000	1	Supir	450 W	23.000	1	Numpang	Tembok	Keramik	80.733	1	0	LAINNYA	B
97	255	PUTRA RAMA TITO R	2.000.000	1	Kuli Bangunan	450 W	55.000	1	Rumah Sendiri	Tembok	Keramik	92.920	2	0	LAINNYA	B

98	65	ANI RAHAYU	1.500.000	1	Kuli Bangunan	900 W	25.000	1	Rumah Sendiri	Tembok	Keramik	111.740	0	0	LAINNYA	C
99	66	M. ILHAM R KURNIAWAN	400.000	2	Buruh Tani	900 W	80.000	1	Rumah Sendiri	Tembok	Keramik	26.100	2	0	LAINNYA	C
100	67	ILHAM ARFIANSYAH	700.000	1	Pedagang Kecil	900 W	100.000	1	Rumah Sendiri	Tembok	Tegel	114.300	1	0	LAINNYA	C
101	68	LAYLA RAHAYU	800.000	1	Karyawan	900 W	60.000	1	Rumah Sendiri	Tembok	Keramik	37.943	2	0	LAINNYA	C
102	69	M. BAGUS ADE P	2.000.000	3	Pedagang Roti	900 W	60.000	1	Rumah Sendiri	Tembok	Keramik	43.578	1	0	LAINNYA	C
103	70	DESI TRI W	2.500.000	3	Tukang Las	1.300 W	130.000	1	Rumah Sendiri	Tembok	Tegel	43.576	1	0	LAINNYA	C
104	85	MOCH ZAENAL A	2.000.000	1	Buruh Tani	900 W	50.000	1	Rumah Sendiri	Tembok	Tegel	57.613	2	0	LAINNYA	C
105	86	M. KHOIRUL MUNA	1.000.000	3	Buruh Tani	450 W	40.000	1	Rumah Sendiri	Tembok	Keramik	99.120	2	0	LAINNYA	C
106	87	ZUMROTUS SHOLIKHAH	1.600.000	2	Buruh Tani	450 W	45.593	1	Rumah Sendiri	Tembok	Keramik	38.475	1	0	LAINNYA	C
107	400	M MUSYAFA YAZIO	4.200.000	2	Petani	450 W	22.000	1	Rumah Sendiri	Tembok	Tegel	167.000	2	0	BIDIK MISI	C
108	89	MIRA SAGOLA	1.500.000	2	Tukang Las	450 W	31.300	1	Rumah Sendiri	Tembok	Keramik	49.600	1	0	BIDIK MISI	C
109	90	FIRDA LAILA S	2.200.000	3	Guru swasta	450 W	35.000	1	Rumah Sendiri	Tembok	Keramik	75.642	2	0	LAINNYA	C
110	91	AMALIYA AYU ALFINA	2.700.000	1	Wiraswasta	900 W	110.000	1	Rumah Sendiri	Tembok	Keramik	87.652	2	0	LAINNYA	C
111	401	RINA RAHAYU N	1.200.000	1	Wiraswasta	450 W	40.000	1	Rumah Sendiri	Tembok	Keramik	173.000	1	0	LAINNYA	C

Lampiran C Parameter Dari Setiap Kriteria

Dinding Rumah	
Parameter	**Nilai**
Bambu	3
Kayu	2
Tembok	1

Status Rumah	
Parameter	**Nilai**
Kontrak/numpang	3
Milik Ortu	2
Milik Sendiri	1
tidak ada data	0

Lantai Rumah	
Parameter	**Nilai**
Tanah	3
Tegel	2
Keramik	1
tidak ada data	0

Pekerjaan	
Parameter	**Nilai**
Tidak bekerja	5
Buruh	4
Karyawan	3
Wiraswasta	2
Pengusaha	1

Jumlah Motor	
Parameter	**Nilai**
0	5
1	4
2	3
3	2
>3	1

Jumlah Mobil	
Parameter	**Nilai**
0	3
1	2
>1	1

Biaya Listrik	
Parameter	**Nilai**
Rp0	5
Rp 1 - 50.000	4
50.001 - 100.000	3
100.001 - 150.000	2
> 150.000	1

Total Penghasilan	
Prameter	Nilai
Rp. 0	5
1 - 500.000	4
501.000 - 1.000.000	3
1.000.001 - 2.000.000	2
>2.000.000	1

Program Beasiswa	
Parameter	**Nilai**
Bidik Misi	2
Lainnya	1

Lampiran C Parameter Dari Setiap Kriteria (Lanjutan)

Daya Listrik	
Parameter	**Nilai**
0 W	6
220 W	5
450 W	4
900 W	3
1.300 W	2
2.200 W	1

Jumlah Rumah	
Parameter	**Nilai**
0	3
1	2
>1	1

PBB	
Parameter	**Nilai**
Rp. 0 - 50.000	4
50.001 - 100.000	3
100.001 - 200.000	2
> 200.001	1

Jumlah Tanggungan Anak	
Parameter	**Nilai**
>5	3
3 - 5	2
0 - 3	1

Lampiran D Perhitungan Manualisasi Metode SVM Level 1

MATRIKS PENILAIAN ALTERNATIF (DATA TRAINING/LATIH) / NORMALISASI															
Ai	**C1**	**C2**	**C3**	**C4**	**C5**	**C6**	**C7**	**C8**	**C9**	**C10**	**C11**	**C12**	**C13**	**KELAS**	**Yi**
A23	1,000	1,000	1,000	0,550	1,000	0,100	0,100	0,100	0,550	1,000	1,000	1,000	1,000	A	1
A24	0,775	0,100	1,000	1,000	1,000	0,100	0,100	0,100	0,550	0,400	0,700	1,000	1,000	A	1
A25	0,550	0,100	0,775	1,000	1,000	0,100	0,100	0,550	0,550	1,000	0,700	1,000	1,000	A	1
A73	0,775	0,100	0,325	1,000	1,000	0,100	0,100	0,100	0,100	0,700	0,400	1,000	0,100	B	-1
A74	0,325	0,100	0,550	1,000	0,700	0,100	0,100	0,100	0,100	1,000	0,700	1,000	1,000	B	-1
A75	0,550	0,100	0,325	1,000	0,700	0,100	0,100	0,100	0,100	0,700	0,700	1,000	0,100	B	-1
A102	0,325	0,100	0,100	0,550	0,700	0,100	0,100	0,100	0,100	1,000	0,700	1,000	0,100	C	-1
A103	0,100	0,100	0,550	0,100	0,400	0,100	0,100	0,100	0,550	1,000	0,700	1,000	0,100	C	-1
A104	0,325	0,100	0,775	0,550	1,000	0,100	0,100	0,100	0,550	0,700	0,400	1,000	0,100	C	-1

MATRIKS PENILAIAN ALTERNATIF (DATA TESTING/UJI) / NORMALISASI														
Ai	**C1**	**C2**	**C3**	**C4**	**C5**	**C6**	**C7**	**C8**	**C9**	**C10**	**C11**	**C12**	**C13**	**KELAS**
A14	0,775	0,100	0,775	1,000	1,000	0,100	0,100	0,100	0,550	0,400	0,400	1,000	1,000	A
A64	0,550	0,100	0,775	0,550	0,400	0,100	0,100	0,100	0,100	1,000	0,400	1,000	0,100	B
A99	0,775	0,100	0,775	0,550	0,700	0,100	0,100	0,100	0,100	1,000	0,400	1,000	0,100	C
A32	1,000	0,100	0,775	1,000	0,400	0,100	0,100	0,100	0,100	0,400	0,700	1,000	1,000	A
A65	0,775	0,100	0,550	1,000	1,000	0,100	0,100	0,100	0,100	1,000	0,400	1,000	0,100	B

Lampiran D Perhitungan Manualisasi Metode SVM Level 1 (Lanjutan)

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

BA	1	
BB	0,1	
Lamda	0,5	
d	2	
C	1	
iterMAX	2	
αi	0	
ε	0,001	
γ	0,01	0,000133666

PERHITUNGAN KERNEL POLYNOMIAL DEGREE 2									
K(x,y)	**A23**	**A24**	**A25**	**A73**	**A74**	**A75**	**A102**	**A103**	**A104**
A23	74,56322500	47,02530625	49,73775625	25,35122500	36,12010000	23,13610000	19,47015625	18,81390625	25,35122500
A24	47,02530625	43,46929727	41,97420156	21,90825039	30,10579727	19,50326406	13,59304727	11,97160000	20,97067539
A25	49,73775625	41,97420156	49,39454102	23,99162852	35,22422500	21,94336914	17,94581406	15,67170156	23,01000977
A73	25,35122500	21,90825039	23,99162852	19,50326406	18,15292539	16,49882852	12,77615664	8,07270156	14,50657656
A74	36,12010000	30,10579727	35,22422500	18,15292539	29,57320352	17,53515625	14,75040039	11,56000000	15,65685977
A75	23,13610000	19,50326406	21,94336914	16,49882852	17,53515625	15,50882852	12,25875156	8,46082656	12,64246914
A102	19,47015625	13,59304727	17,94581406	12,77615664	14,75040039	12,25875156	11,95862852	9,10530625	10,69698789
A103	18,81390625	11,97160000	15,67170156	8,07270156	11,56000000	8,46082656	9,10530625	11,05562500	10,53813906
A104	25,35122500	20,97067539	23,01000977	14,50657656	15,65685977	12,64246914	10,69698789	10,53813906	16,09012656

Lampiran D Perhitungan Manualisasi Metode SVM Level 1 (Lanjutan)

MATRIKS HESSIAN									
Dij	**A23**	**A24**	**A25**	**A73**	**A74**	**A75**	**A102**	**A103**	**A104**
A23	74,81	47,28	49,99	-25,60	-36,37	-23,39	-19,72	-19,06	-25,60
A24	47,28	43,72	42,22	-22,16	-30,36	-19,75	-13,84	-12,22	-21,22
A25	49,99	42,22	49,64	-24,24	-35,47	-22,19	-18,20	-15,92	-23,26
A73	-25,60	-22,16	-24,24	19,75	18,40	16,75	13,03	8,32	14,76
A74	-36,37	-30,36	-35,47	18,40	29,82	17,79	15,00	11,81	15,91
A75	-23,39	-19,75	-22,19	16,75	17,79	15,76	12,51	8,71	12,89
A102	-19,72	-13,84	-18,20	13,03	15,00	12,51	12,21	9,36	10,95
A103	-19,06	-12,22	-15,92	8,32	11,81	8,71	9,36	11,31	10,79
A104	-25,60	-21,22	-23,26	14,76	15,91	12,89	10,95	10,79	16,34

ITERASI										
ERROR ITERASI 1 KELAS LEVEL 1										
αiDij	**A23**	**A24**	**A25**	**A73**	**A74**	**A75**	**A102**	**A103**	**A104**	**Ei**
A23	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
A24	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
A25	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
A73	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
A74	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
A75	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
A102	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
A103	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00
A104	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00

Lampiran D Perhitungan Manualisasi Metode SVM Level 1 (Lanjutan)

MENCARI nilai δa_i	
δαi	**1**
A23	0,000133666
A24	0,000133666
A25	0,000133666
A73	0,000133666
A74	0,000133666
A75	0,000133666
A102	0,000133666
A103	0,000133666
A104	0,000133666

MENCARI nilai αi (alpha baru iterasi 1)	
αi	**1**
A23	0,000133666
A24	0,000133666
A25	0,000133666
A73	0,000133666
A74	0,000133666
A75	0,000133666
A102	0,000133666
A103	0,000133666
A104	0,000133666

ERROR ITERASI 2 KELAS LEVEL 1										
αiDij	**A23**	**A24**	**A25**	**A73**	**A74**	**A75**	**A102**	**A103**	**A104**	**Ei**
A23	0,010	0,006	0,007	-0,003	-0,005	-0,003	-0,003	-0,003	-0,003	0,003
A24	0,006	0,006	0,006	-0,003	-0,004	-0,003	-0,002	-0,002	-0,003	0,002
A25	0,007	0,006	0,007	-0,003	-0,005	-0,003	-0,002	-0,002	-0,003	0,000
A73	-0,003	-0,003	-0,003	0,003	0,002	0,002	0,002	0,001	0,002	0,003
A74	-0,005	-0,004	-0,005	0,002	0,004	0,002	0,002	0,002	0,002	0,001
A75	-0,003	-0,003	-0,003	0,002	0,002	0,002	0,002	0,001	0,002	0,003
A102	-0,003	-0,002	-0,002	0,002	0,002	0,002	0,002	0,001	0,001	0,003
A103	-0,003	-0,002	-0,002	0,001	0,002	0,001	0,001	0,002	0,001	0,002
A104	-0,003	-0,003	-0,003	0,002	0,002	0,002	0,001	0,001	0,002	0,002

Lampiran D Perhitungan Manualisasi Metode SVM Level 1 (Lanjutan)

MENCARI nilai δa_i	
δαi	**2**
A23	0,00013327
A24	0,00013342
A25	0,00013362
A73	0,00013333
A74	0,00013355
A75	0,00013333
A102	0,00013329
A103	0,00013343
A104	0,00013346

MENCARI nilai αi (alpha 2 iterasi 2)	
αi	**2**
A23	0,00026693
A24	0,00026709
A25	0,00026729
A73	0,00026699
A74	0,00026722
A75	0,00026699
A102	0,00026695
A103	0,00026710
A104	0,00026713

X+	0,00026729	A25
X-	0,00026722	A74

Lampiran D Perhitungan Manualisasi Metode SVM Level 1 (Lanjutan)

ID	K(xi,x+)	K(xi,x-)
A23	49,73775625	36,1201
A24	41,97420156	30,10579727
A25	49,39454102	35,224225
A73	23,99162852	18,15292539
A74	35,224225	29,57320352
A75	21,94336914	17,53515625
A102	17,94581406	14,75040039
A103	15,67170156	11,56
A104	23,01000977	15,65685977

ID	w.x+	w.x-
A23	0,000132963	0,000096559
A24	0,000112210	0,000080482
A25	0,000132047	0,000094166
A73	-0,000064137	-0,000048528
A74	-0,000094165	-0,000079058
A75	-0,000058661	-0,000046877
A102	-0,000047974	-0,000039432
A103	-0,000041895	-0,000030903
A104	-0,000061513	-0,000041856
JUMLAH	0,00000887503	-0,00001544722

MENGHITUNG NILAI BIAS	
MENCARI NILAI B KELAS LEVEL 1	0,000325461

Lampiran D Perhitungan Manualisasi Metode SVM Level 1 (Lanjutan)

Mencari nilai f(x) Data UJI 1 kelas A level 1		
ID	**k(xi,x)**	**αiyiK(xi,x)**
A23	40,10055625	0,000107200
A24	37,92250352	0,000101378
A25	37,14140664	0,000099291
A73	20,13765625	-0,000053834
A74	26,55469727	-0,000070989
A75	17,08272227	-0,000045667
A102	11,93270664	-0,000031900
A103	9,773439063	-0,000026127
A104	18,361225	-0,000049085
JUMLAH		0,003026147

MENCARI NILAI F(X) data UJI 1 kelas A level 1	0,000033553
FUNGSI KLASIFIKASI	1

Mencari nilai f(x) Data UJI 2 kelas A level 1		
ID	**k(xi,x)**	**αiyiK(xi,x)**
A23	36,1201	0,000096559
A24	32,433025	0,000086703
A25	31,14337539	0,000083256
A73	17,11373477	-0,000045750
A74	24,71332656	-0,000066066
A75	16,01500352	-0,000042813
A102	10,70925625	-0,000028629
A103	8,244076563	-0,000022039
A104	13,18143789	-0,000035238
JUMLAH		-0,003521469

MENCARI NILAI F(X) data UJI 2 kelas A level 1	0,000029270
FUNGSI KLASIFIKASI	1

Lampiran D Perhitungan Manualisasi Metode SVM Level 1 (Lanjutan)

Mencari nilai f(x) Data UJI 1 kelas B level 1		
ID	**k(xi,x)**	**αiyiK(xi,x)**
A23	22,20765625	0,000059367
A24	16,21068906	0,000043336
A25	19,12422227	0,000051125
A73	12,58919102	-0,000033655
A74	14,938225	-0,000039935
A75	11,72634414	-0,000031348
A102	10,10445156	-0,000027012
A103	9,494101563	-0,000025381
A104	11,88094727	-0,000031761
JUMLAH		-0,0036794

MENCARI NILAI F(X) data UJI 1 kelas B level 1	-0,000031976
FUNGSI KLASIFIKASI	-1

Mencari nilai f(x) Data UJI 2 kelas B level 1		
ID	**k(xi,x)**	**αiyiK(xi,x)**
A23	30,9136	0,000082641
A24	25,2569066	0,000067519
A25	28,8637563	0,000077162
A73	22,0841254	-0,000059037
A74	21,9433691	-0,000058661
A75	18,879025	-0,000050469
A102	15,1856348	-0,000040596
A103	10,660225	-0,000028498
A104	17,5822973	-0,000047003
JUMLAH		-0,0036794

MENCARI NILAI F(X) data UJI 2 kelas B level 1	-0,000053656
FUNGSI KLASIFIKASI	-1

Lampiran D Perhitungan Manualisasi Metode SVM Level 1 (Lanjutan)

Mencari nilai f(x) Data UJI 1 kelas C level 1		
ID	**k(xi,x)**	**αiyiK(xi,x)**
A23	27,4314063	0,000073332
A24	20,2556254	0,000054149
A25	23,0100098	0,000061513
A73	16,1805063	-0,000043255
A74	17,206941	-0,000046000
A75	14,1235035	-0,000037756
A102	11,9845785	-0,000032038
A103	10,3925641	-0,000027782
A104	14,5924	-0,000039010
JUMLAH		-0,0036794

MENCARI NILAI F(X) data UJI 1 kelas C level 1	-0,000033561
FUNGSI KLASIFIKASI	-1

Lampiran E Perhitungan Manualisasi Metode SVM Level 2

MATRIKS PENILAIAN ALTERNATIF (DATA TRAINING/LATIH)															
Ai	**C1**	**C2**	**C3**	**C4**	**C5**	**C6**	**C7**	**C8**	**C9**	**C10**	**C11**	**C12**	**C13**	**KELAS**	**Yi**
A73	0,775	0,100	0,325	1,000	1,000	0,100	0,100	0,100	0,100	0,700	0,400	1,000	0,100	B	1
A74	0,325	0,100	0,550	1,000	0,700	0,100	0,100	0,100	0,100	1,000	0,700	1,000	1,000	B	1
A75	0,550	0,100	0,325	1,000	0,700	0,100	0,100	0,100	0,100	0,700	0,700	1,000	0,100	B	1
A102	0,325	0,100	0,100	0,550	0,700	0,100	0,100	0,100	0,100	1,000	0,700	1,000	0,100	C	-1
A103	0,100	0,100	0,550	0,100	0,400	0,100	0,100	0,100	0,550	1,000	0,700	1,000	0,100	C	-1
A104	0,325	0,100	0,775	0,550	1,000	0,100	0,100	0,100	0,550	0,700	0,400	1,000	0,100	C	-1

MATRIKS PENILAIAN ALTERNATIF (DATA TESTING/UJI)														
Ai	**C1**	**C2**	**C3**	**C4**	**C5**	**C6**	**C7**	**C8**	**C9**	**C10**	**C11**	**C12**	**C13**	**KELAS**
A64	0,5500	0,1000	0,7750	0,5500	0,4000	0,1000	0,1000	0,1000	0,1000	1,0000	0,4000	1,0000	0,1000	B
A99	0,7750	0,1000	0,7750	0,5500	0,7000	0,1000	0,1000	0,1000	0,1000	1,0000	0,4000	1,0000	0,1000	C
A65	0,775	0,100	0,550	1,000	1,000	0,100	0,100	0,100	0,100	1,000	0,400	1,000	0,100	B

	1	2	3	4	5	6
αi	0	0	0	0	0	0

Lampiran E Perhitungan Manualisasi Metode SVM Level 2 (Lanjutan)

PERHITUNGAN KERNEL POLYNOMIAL DEGREE 2						
K(x,y)	**A73**	**A74**	**A75**	**A102**	**A103**	**A104**
A73	19,50326406	18,15292539	16,49882852	12,77615664	8,07270156	14,50657656
A74	18,15292539	29,57320352	17,53515625	14,75040039	11,56000000	15,65685977
A75	16,49882852	17,53515625	15,50882852	12,25875156	8,46082656	12,64246914
A102	12,77615664	14,75040039	12,25875156	11,95862852	9,10530625	10,69698789
A103	8,07270156	11,56000000	8,46082656	9,10530625	11,05562500	10,53813906
A104	14,50657656	15,65685977	12,64246914	10,69698789	10,53813906	16,09012656

MATRIKS HESSIAN						
Dij	**A73**	**A74**	**A75**	**A102**	**A103**	**A104**
A73	19,75326406	18,40292539	16,74882852	-13,02615664	-8,32270156	-14,75657656
A74	18,40292539	29,82320352	17,78515625	-15,00040039	-11,81000000	-15,90685977
A75	16,74882852	17,78515625	15,75882852	-12,50875156	-8,71082656	-12,89246914
A102	-13,02615664	-15,00040039	-12,50875156	12,20862852	9,35530625	10,94698789
A103	-8,32270156	-11,81000000	-8,71082656	9,35530625	11,30562500	10,78813906
A104	-14,75657656	-15,90685977	-12,89246914	10,94698789	10,78813906	16,34012656

Lampiran E Perhitungan Manualisasi Metode SVM Level 2 (Lanjutan)

ERROR ITERASI 1 KELAS LEVEL 2							
αiDij	**A73**	**A74**	**A75**	**A102**	**A103**	**A104**	**Ei**
A73	0,00000000	0,00000000	0,00000000	0,00000000	0,00000000	0,00000000	0,00000000
A74	0,00000000	0,00000000	0,00000000	0,00000000	0,00000000	0,00000000	0,00000000
A75	0,00000000	0,00000000	0,00000000	0,00000000	0,00000000	0,00000000	0,00000000
A102	0,00000000	0,00000000	0,00000000	0,00000000	0,00000000	0,00000000	0,00000000
A103	0,00000000	0,00000000	0,00000000	0,00000000	0,00000000	0,00000000	0,00000000
A104	0,00000000	0,00000000	0,00000000	0,00000000	0,00000000	0,00000000	0,00000000

MENCARI nilai δa_i	
δαi	**1**
A73	0,000335309
A74	0,000335309
A75	0,000335309
A102	0,000335309
A103	0,000335309
A104	0,000335309

MENCARI nilai αi (alpha baru iterasi 1)	
αi	**1**
A73	0,000335309
A74	0,000335309
A75	0,000335309
A102	0,000335309
A103	0,000335309
A104	0,000335309

Lampiran E Perhitungan Manualisasi Metode SVM Level 2 (Lanjutan)

ERROR ITERASI 2 KELAS LEVEL 2							
Dij	**A73**	**A74**	**A75**	**A102**	**A103**	**A104**	**Ei**
A73	0,00662345	0,00617067	0,00561604	-0,00436779	-0,00279068	-0,00494802	0,00630368
A74	0,00617067	0,01000000	0,00596353	-0,00502978	-0,00396000	-0,00533372	0,00781071
A75	0,00561604	0,00596353	0,00528408	-0,00419430	-0,00292082	-0,00432297	0,00542556
A102	-0,00436779	-0,00502978	-0,00419430	0,00409367	0,00313692	0,00367063	-0,00269065
A103	-0,00279068	-0,00396000	-0,00292082	0,00313692	0,00379088	0,00361736	0,00087366
A104	-0,00494802	-0,00533372	-0,00432297	0,00367063	0,00361736	0,00547900	-0,00183771

MENCARI nilai δa_i	
δαi	**2**
A73	0,000333196
A74	0,00033269
A75	0,00033349
A102	0,000336212
A103	0,000335016
A104	0,000335926

MENCARI nilai αi (alpha 2 iterasi 2)	
αi	**2**
A73	0,000668505
A74	0,000668
A75	0,0006688
A102	0,000671521
A103	0,000670326
A104	0,000671235

X+	0,00066880	A75
X-	0,00067152	A102

Lampiran E Perhitungan Manualisasi Metode SVM Level 2 (Lanjutan)

ID	K(xi,x+)	K(xi,x-)
A73	16,49882852	12,77615664
A74	17,53515625	14,75040039
A75	15,50882852	12,25875156
A102	12,25875156	11,95862852
A103	8,460826563	9,10530625
A104	12,64246914	10,69698789

ID	w.x+	w.x-
A73	0,011029551	0,008540926
A74	0,01171348	0,009853264
A75	0,010372297	0,008198647
A102	-0,008232009	-0,00803047
A103	-0,005671511	-0,006103522
A104	-0,008486067	-0,007180192
JUMLAH	0,010725742	0,005278653

MENGHITUNG NILAI BIAS	
MENCARI NILAI B KELAS LEVEL 2	-0,008002197

Lampiran E Perhitungan Manualisasi Metode SVM Level 2 (Lanjutan)

Mencari nilai f(x) Data UJI 1 kelas B level 2		
ID	**k(xi,x)**	**αiyiK(xi,x)**
A73	12,58919102	0,008415938
A74	14,938225	0,009978731
A75	11,72634414	0,007842573
A102	10,10445156	-0,006785351
A103	9,494101563	-0,006364141
A104	11,88094727	-0,007974907
JUMLAH		0,005112843

MENCARI NILAI F(X) data UJI 1 kelas B level 2	-0,002889355
FUNGSI KLASIFIKASI	-1

Mencari nilai f(x) Data UJI 2 kelas B level 2		
ID	**k(xi,x)**	**αiyiK(xi,x)**
A73	22,08412539	0,01476335
A74	21,94336914	0,014658165
A75	18,879025	0,012626283
A102	15,18563477	-0,010197472
A103	10,660225	-0,007145824
A104	17,58229727	-0,011801853
JUMLAH		0,006947547

MENCARI NILAI F(X) data UJI 2 kelas B level 2	0,004900452
FUNGSI KLASIFIKASI	1

Lampiran E Perhitungan Manualisasi Metode SVM Level 2 (Lanjutan)

Mencari nilai f(x) Data UJI 2 kelas B level 2		
ID	**k(xi,x)**	**αiyiK(xi,x)**
A73	16,18050625	0,010816751
A74	17,20694102	0,011494233
A75	14,12350352	0,009445792
A102	11,98457852	-0,008047896
A103	10,39256406	-0,006966404
A104	14,5924	-0,009794929
JUMLAH		0,006947547

MENCARI NILAI F(X) data UJI 1 kelas C level 2	-0,00105465
FUNGSI KLASIFIKASI	-1

ID	f^1(x) / level 1	f^2(x) / level 2	HASIL KLASIFIKASI	ACTUAL CLASS
A14	0,00003355347	-	1 DILEVEL 1 = A	A
A32	0,00002926982	-	1 DILEVEL 1= A	A
A64	-0,00003197642	-0,00288935461	-1 DILEVEL 2 = C	B
A65	-0,00005365596	0,00490045230	1 DILEVEL 2 = B	B
A99	-0,00003356100	-0,00105465048	-1 DILEVEL 2 = C	C

Lampiran F Perhitungan Manulisasi Metode TOPSIS

NAMA	KRITERIA													KELAS
Ai	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	
A14	0,775	0,100	0,775	1,000	1,000	0,100	0,100	0,100	0,550	0,400	0,400	1,000	1,000	A
A32	1,000	0,100	0,775	1,000	0,400	0,100	0,100	0,100	0,100	0,400	0,700	1,000	1,000	A

NAMA	KRITERIA													KELAS
Ai	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	
A65	0,775	0,100	0,550	1,000	1,000	0,100	0,100	0,100	0,100	1,000	0,400	1,000	0,100	B

BOBOT														
	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	TOTAL
BOBOT	0,15	0,15	0,05	0,1	0,1	0,05	0,05	0,05	0,05	0,1	0,05	0,05	0,05	1

MATRIKS NORMALISASI													
NAMA	KRITERIA												
Ai	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13
A14	0,6126	0,7071	0,7071	0,7071	0,9285	0,7071	0,7071	0,7071	0,9839	0,7071	0,4961	0,7071	0,7071
A32	0,7904	0,7071	0,7071	0,7071	0,3714	0,7071	0,7071	0,7071	0,1789	0,7071	0,8682	0,7071	0,7071

Lampiran F Perhitungan Manualsiasi Metode TOPSIS (Lanjutan)

MATRIKS NORMALISASI													
NAMA	KRITERIA												
Ai	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13
A65	1,0000	1,0000	1,0000	1,0000	1,0000	1,0000	1,0000	1,0000	1,0000	1,0000	1,0000	1,0000	1,0000

PEMBOBOTAN MATRIKS NORMALISASI													
NAMA	KRITERIA												
Ai	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13
A14	0,0919	0,1061	0,0354	0,0707	0,0928	0,0354	0,0354	0,0354	0,0492	0,0707	0,0248	0,0354	0,0354
A32	0,1186	0,1061	0,0354	0,0707	0,0371	0,0354	0,0354	0,0354	0,0089	0,0707	0,0434	0,0354	0,0354

PEMBOBOTAN MATRIKS NORMALISASI													
NAMA	KRITERIA												
Ai	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13
A65	0,1500	0,1500	0,0500	0,1000	0,1000	0,0500	0,0500	0,0500	0,0500	0,1000	0,0500	0,0500	0,0500

Menentukan Ideal Positif Dan Negatif

	NAMA	KRITERIA												
	Ai	**C1**	**C2**	**C3**	**C4**	**C5**	**C6**	**C7**	**C8**	**C9**	**C10**	**C11**	**C12**	**C13**
MAX A+	**A14**	0,0919	0,1061	0,0354	0,0707	0,0928	0,0354	0,0354	0,0354	0,0492	0,0707	0,0248	0,0354	0,0354
	A32	0,1186	0,1061	0,0354	0,0707	0,0371	0,0354	0,0354	0,0354	0,0089	0,0707	0,0434	0,0354	0,0354
	MAX A+	0,1186	0,1061	0,0354	0,0707	0,0928	0,0354	0,0354	0,0354	0,0492	0,0707	0,0434	0,0354	0,0354

	NAMA	KRITERIA												
	Ai	**C1**	**C2**	**C3**	**C4**	**C5**	**C6**	**C7**	**C8**	**C9**	**C10**	**C11**	**C12**	**C13**
MAX A+	**A65**	0,1500	0,1500	0,0500	0,1000	0,1000	0,0500	0,0500	0,0500	0,0500	0,1000	0,0500	0,0500	0,0500
	MAX A+	0,1500	0,1500	0,0500	0,1000	0,1000	0,0500	0,0500	0,0500	0,0500	0,1000	0,0500	0,0500	0,0500

	NAMA	KRITERIA												
	Ai	**C1**	**C2**	**C3**	**C4**	**C5**	**C6**	**C7**	**C8**	**C9**	**C10**	**C11**	**C12**	**C13**
MIN A-	**A14**	0,0919	0,1061	0,0354	0,0707	0,0928	0,0354	0,0354	0,0354	0,0492	0,0707	0,0248	0,0354	0,0354
	A32	0,1186	0,1061	0,0354	0,0707	0,0371	0,0354	0,0354	0,0354	0,0089	0,0707	0,0434	0,0354	0,0354
	MIN A-	0,0919	0,1061	0,0354	0,0707	0,0371	0,0354	0,0354	0,0354	0,0089	0,0707	0,0248	0,0354	0,0354

	NAMA	KRITERIA												
	Ai	**C1**	**C2**	**C3**	**C4**	**C5**	**C6**	**C7**	**C8**	**C9**	**C10**	**C11**	**C12**	**C13**
MAX A+	**A65**	0,1500	0,1500	0,0500	0,1000	0,1000	0,0500	0,0500	0,0500	0,0500	0,1000	0,0500	0,0500	0,0500
	MIN A-	0,1500	0,1500	0,0500	0,1000	0,1000	0,0500	0,0500	0,0500	0,0500	0,1000	0,0500	0,0500	0,0500

Lampiran F Perhitungan Manualsiasi Metode TOPSIS (Lanjutan)

MENGHITUNG SEPARATION MEASURE	
NAMA	SI+
A14	0,0325
A32	0,0687

MENGHITUNG SEPARATION MEASURE	
NAMA	SI+
A65	0,0000

MENGHITUNG SEPARATION MEASURE	
NAMA	SI-
A14	0,06873
A32	0,03252

MENGHITUNG SEPARATION MEASURE	
NAMA	SI-
A65	0,0000

MENGHITUNG NILAI PREFENSI	
NAMA	PREFENSI
A14	0,6788
A32	0,3212

MENGHITUNG NILAI PREFENSI	
NAMA	PREFENSI
A65	0,0000

PERANGKINGAN		
NAMA	PREFENSI	RANGKING
A14	0,6788	A
A32	0,3212	A
A65	0,0000	B

Lampiran G Hasil Pengujian

1. Pengujian Rasio Perbandingan

Parameter	
lambda	0.5
konstanta gamma	0.01
epsilon	0.0001
itermac	100
c	1
d	2

Nilai *Ei*

No	*Ei* Level 1	No	*Ei* Level 2
197	0,79068351	264	0,5375399
205	0,94634366	195	0,53870356
196	0,75892288	132	0,35635757
174	0,7337631	230	0,05414894
210	0,66722012	142	0,114279
211	0,89030618	165	0,31478071
187	0,74651367	133	0,79826248
163	0,94822013	228	0,95686078
168	0,78520012	206	0,77343857
135	0,66227627	246	0,65597475
254	0,86668754	225	0,72773308
137	0,83354539	221	0,78501785
227	0,72386009	251	0,72661787
153	0,55557108	236	0,78949899
140	0,48197985	250	0,84207952
180	0,64737612	87	0,62484205
157	0,78553152	241	0,68485647
66	0,67031491	263	0,73514462
178	0,44084781	242	0,6784575
215	0,64192486	179	0,71810925
183	0,34692019	232	0,66403878
189	0,32047883	255	0,56381595
156	0,13327873	85	0,62345469
182	0,63284421	257	0,53623414
271	0,37824184	65	0,59690666
214	0,32732207	272	0,31927246
138	0,42534667	234	0,62075412
217	0,24812301	89	0,36603242
200	0,38404825	268	0,45191389

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

245	0,41300261	253	0,58198345
274	0,29883966	269	0,33501017
169	0,06098151	192	0,26312453
273	0,23580724	144	-0,2339998
152	0,42934462	401	-0,425362
248	0,2406296	220	-0,1771076
173	0,33707535	188	-0,1027979
159	0,27384278	199	-0,0872085
203	0,08911446	249	-0,0872085
139	0,27867782	69	0,01398718
231	0,29920787	198	-0,0570801
243	0,09581421	204	-0,328519
86	0,25818777	233	-0,4219928
258	0,37633815	70	-0,0636303
166	0,28533331	90	-0,2886471
160	0,48787388	201	0,24434048
67	0,48787388	400	-0,2090976
194	0,40652335	-	-
177	0,38566533	-	-
270	0,38072458	-	-
265	0,02480613	-	-
247	0,03955065	-	-
162	0,24768379	-	-
224	-0,0358415	-	-
170	-0,0153122	-	-
68	0,2410769	-	-
264	0,0518931	-	-
195	-0,0605305	-	-
132	-0,0907751	-	-
230	0,18408594	-	-
142	0,05044471	-	-
165	0,2432487	-	-
133	-0,0604394	-	-
228	0,21658953	-	-
206	0,33126155	-	-
246	0,22974053	-	-
225	0,17162134	-	-
221	-0,0263858	-	-
251	0,20041408	-	-
236	0,11862594	-	-
250	0,07020284	-	-
87	0,07006076	-	-
241	0,03431493	-	-

263	0,00313906	-	-
242	0,19938098	-	-
179	0,0304059	-	-
232	0,43130994	-	-
255	0,00462124	-	-
85	-0,106337	-	-
257	-0,0524233	-	-
65	-0,1366421	-	-
272	0,10266662	-	-
234	0,28673559	-	-
89	0,43040633	-	-
268	0,38251331	-	-
253	0,50962961	-	-
269	0,52019453	-	-
192	0,34944695	-	-
144	0,69228238	-	-
401	0,22256252	-	-
220	0,66311401	-	-
188	0,48061061	-	-
199	0,40496656	-	-
249	0,40496656	-	-
69	0,21996325	-	-
198	0,17742156	-	-
204	0,14799509	-	-
233	0,34938753	-	-
70	0,08252647	-	-
90	0,55416542	-	-
201	0,46259311	-	-
400	0,79855889	-	-

Nilai *Alpha*

No	Alpha	No	Alpha
167	0,001032999	162	0,010296882
197	0,007087396	224	0,013354837
196	0,007888552	170	0,013239834
174	0,006425561	68	0,011094151
210	0,007734622	264	0,013649139
211	0,005839367	195	0,014844641
187	0,007424405	132	0,014846849
163	0,005105334	230	0,011872936
168	0,006985183	267	0,014109301
135	0,007920259	142	0,013099972

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

254	0,005673889	165	0,012813081
137	0,005989992	133	0,015377751
227	0,007136408	228	0,013426919
153	0,008877511	206	0,012198183
140	0,009306736	246	0,012644685
180	0,00748204	225	0,013123233
157	0,006982605	221	0,014994987
178	0,009351698	251	0,012752213
215	0,007879115	236	0,013549116
183	0,009962905	250	0,013926184
189	0,010234059	87	0,014319765
156	0,012623938	241	0,014622935
182	0,007538415	229	0,014255062
271	0,009005757	263	0,014876936
218	0,011071295	242	0,012897633
214	0,01020243	232	0,011348132
138	0,008910482	255	0,014344642
217	0,010760065	85	0,015184049
200	0,010530735	257	0,014852643
245	0,009962443	65	0,015821371
169	0,011929495	272	0,013041954
273	0,011075508	234	0,012466244
216	0,00943168	89	0,011199634
248	0,010790569	268	0,011439671
173	0,009784122	269	0,009680022
203	0,011907067	192	0,011462569
139	0,010074816	144	0,008963247
231	0,009869017	401	0,013072013
243	0,011948539	220	0,008494157
86	0,010332984	188	0,01043962
258	0,009590568	199	0,011137619
256	0,009432076	249	0,011137619
166	0,010524404	69	0,01244339
226	0,010524404	198	0,01279599
160	0,00904292	233	0,011812635
67	0,00904292	70	0,013260514
194	0,00968806	90	0,010635595
185	0,010423356	91	0,011966781
270	0,010347523	201	0,010249114
265	0,012090865	400	0,008389881
247	0,011290762		

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Nilai *bias*

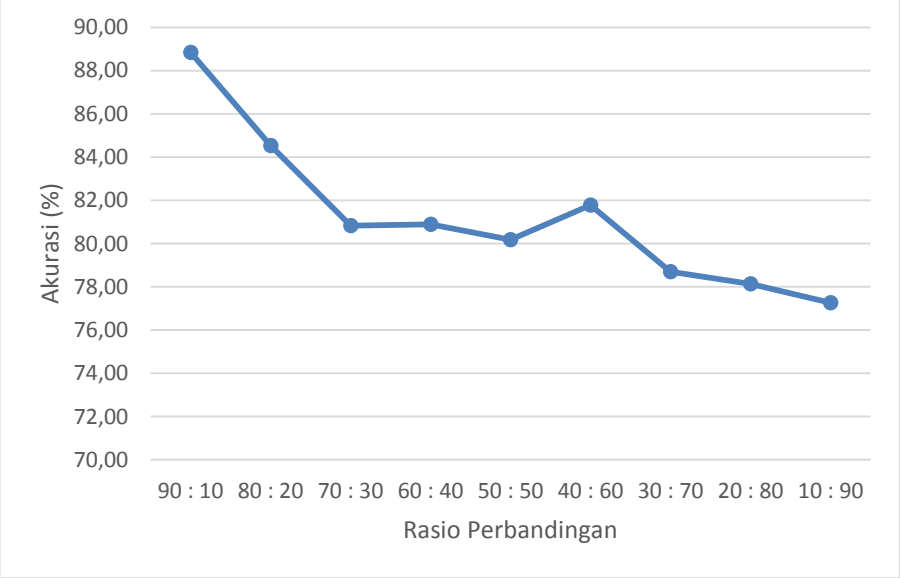
Bias Level 1	Bias Level 2
-0,069205247	-0,247380227

Kelas Prediksi

No	Kelas Prediksi	Kelas Aktual
167	1	1
218	1	1
216	1	1
256	1	1
226	1	1
185	1	1
267	1	-1
229	1	-1
266	1	-1
91	-1	-1

Akurasi Terbaik

Rasio (%)	**Percobaan Ke-*i***					**Rata-rata**
	1	**2**	**3**	**4**	**5**	
90 : 10	90,00	82,75	85,71	85,71	100,00	88,83



2. Pengujian Nilai *Lambda*

Parameter	
rasio	90:10
Konstanta gamma	0.01
epsilon	0.0001
itermac	100
c	1
d	2

Nilai *Ei*

No	*Ei* Level 1	No	*Ei* Level 2
167	1,461696	264	0,55287
197	0,723643	195	0,503879
196	0,687598	132	0,324463
174	0,742332	230	0,123273
210	0,692241	267	0,263498
211	0,818	142	0,153906
187	0,649066	165	0,328942
163	0,912145	133	0,80904
168	0,72032	228	0,917583
135	0,588874	206	0,788805
254	0,834743	246	0,714091
137	0,786663	225	0,791263
227	0,660648	221	0,769471
153	0,499887	251	0,77301
140	0,418077	236	0,857121
180	0,637497	250	0,912476
157	0,714836	87	0,611406
178	0,405557	241	0,679719
215	0,613985	229	0,656492
183	0,319266	263	0,736988
189	0,308265	242	0,740086
156	0,033555	232	0,647513
182	0,660273	255	0,557888
271	0,421068	85	0,625417
218	0,190171	257	0,547172
214	0,277166	65	0,638714
138	0,415269	272	0,411356
217	0,234436	234	0,596644
200	0,310603	89	0,40628
245	0,357849	268	0,482418
169	0,061744	269	0,369564

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

273	0,169943	192	0,316328
216	0,363057	144	-0,26531
248	0,225732	401	-0,38941
173	0,34317	220	-0,17056
203	0,069901	188	-0,12826
139	0,275155	199	-0,12979
231	0,311242	249	-0,12979
243	0,088923	69	0,045606
86	0,268474	198	-0,03468
258	0,362895	233	-0,39584
256	0,392424	70	-0,09077
166	0,285933	90	-0,23311
226	0,285933	91	-0,05119
160	0,467786	201	0,157063
67	0,467786	400	-0,18759
194	0,372315	-	-
185	0,313796	-	-
270	0,351421	-	-
265	0,05708	-	-
247	0,143669	-	-
162	0,233738	-	-
224	-0,07104	-	-
170	-0,06878	-	-
68	0,213457	-	-
264	0,053569	-	-
195	-0,10126	-	-
132	-0,09251	-	-
230	0,233539	-	-
267	-0,03161	-	-
142	0,081841	-	-
165	0,17548	-	-
133	-0,0898	-	-
228	0,124059	-	-
206	0,286492	-	-
246	0,227019	-	-
225	0,156682	-	-
221	-0,05628	-	-
251	0,187783	-	-
236	0,093725	-	-
250	0,037598	-	-
87	0,036062	-	-
241	-0,01315	-	-
229	0,016965	-	-

263	-0,05552	-	-
242	0,157771	-	-
232	0,353779	-	-
255	-0,01722	-	-
85	-0,12272	-	-
257	-0,08445	-	-
65	-0,18635	-	-
272	0,111893	-	-
234	0,244358	-	-
89	0,400652	-	-
268	0,345349	-	-
269	0,541231	-	-
192	0,3438	-	-
144	0,653358	-	-
401	0,158365	-	-
220	0,673983	-	-
188	0,468385	-	-
199	0,378388	-	-
249	0,378388	-	-
69	0,205161	-	-
198	0,15145	-	-
233	0,311504	-	-
70	0,087629	-	-
90	0,473928	-	-
91	0,241848	-	-
201	0,459521	-	-
400	0,730158	-	-

Nilai *Alpha*

No	Alpha	No	Alpha
197	0,007238078	170	0,014388816
205	0,006214629	68	0,011902352
196	0,007948183	264	0,015037756
174	0,007450397	195	0,016168384
210	0,008660418	132	0,016432492
211	0,005829517	230	0,013363067
187	0,007469777	142	0,014699326
163	0,005258245	165	0,013321741
168	0,007151604	133	0,016699204
135	0,00818158	228	0,013955697
254	0,005938388	206	0,0129

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

137	0,006180668	246	0,013784038
227	0,007334637	225	0,0142625
153	0,009282693	221	0,016294131
140	0,009788122	251	0,013906166
180	0,008044467	236	0,014697801
157	0,007063928	250	0,01509442
66	0,00825325	87	0,015494982
178	0,009952342	241	0,015761968
215	0,008352714	263	0,015990179
183	0,01071878	242	0,013900819
189	0,011075091	179	0,015638283
156	0,013275816	232	0,011764753
182	0,008625456	255	0,015729235
271	0,010514982	85	0,016652377
214	0,010932694	257	0,016179509
138	0,009713559	65	0,017086923
217	0,011730677	272	0,014442168
200	0,010979807	234	0,013308489
245	0,01053943	89	0,012001242
274	0,011930387	268	0,01228602
169	0,013300728	253	0,011235698
273	0,011797765	269	0,010694288
152	0,009750224	192	0,012536923
248	0,011770148	144	0,009407563
173	0,010721046	401	0,013953001
159	0,011497114	220	0,009344562
203	0,013033664	188	0,011356658
139	0,011051952	199	0,011976735
231	0,010908724	249	0,011976735
243	0,013097528	69	0,013639858
86	0,01137808	198	0,013973851
258	0,010427456	204	0,014143392
166	0,01156734	233	0,012783163
160	0,00967485	70	0,014721062
67	0,00967485	90	0,011113933
194	0,010326065	201	0,01128299
177	0,010740061	400	0,008677561
270	0,011012544		
265	0,013577985		
247	0,013272981		
162	0,011267081		
224	0,014417811		

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

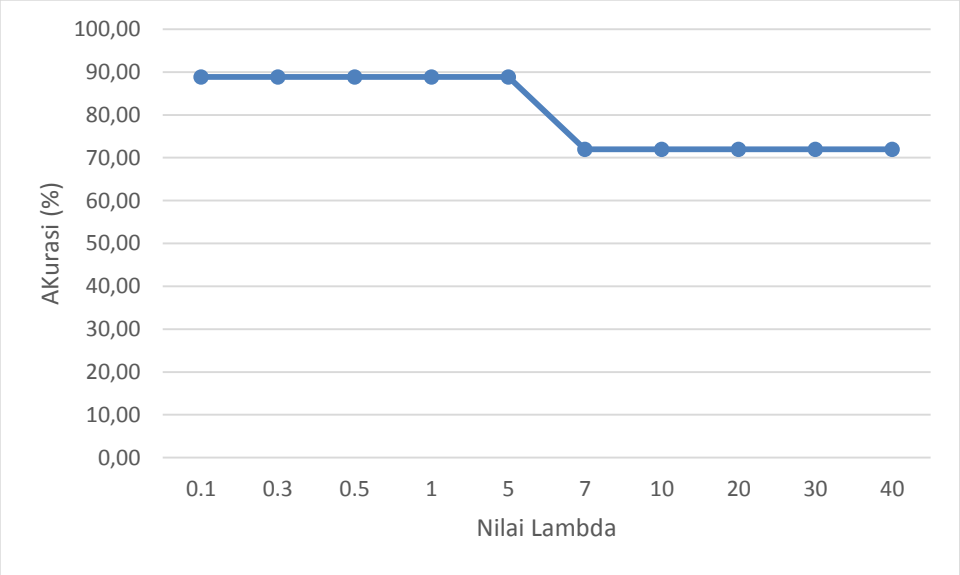
Nilai *bias*

Bias Level 1	Bias Level 2
-0,05836596	-0,271423042

Kelas Prediksi

No	Kelas Prediksi	Kelas Aktual
205	1	1
66	1	1
274	1	1
152	1	1
159	1	1
177	1	1
179	1	-1
266	1	-1
253	1	-1
204	1	-1

Lambda	**Percobaan Ke-*i***					**Rata-rata**
	1	**2**	**3**	**4**	**5**	
0.1	90,00	82,75	85,71	85,71	100,00	88,83



3. Pengujian Konstanta *Gamma*

Parameter	
rasio	90 : 10
Lambda	0.1
epsilon	0.0001
itermac	100
c	1
d	2

Nilai *Ei*

No	*Ei* Level 1	No	*Ei* Level 2
167	2,940308	264	0,657336
197	1,770352	132	0,247657
205	1,977813	230	0,298491
174	1,244185	267	0,493107
210	1,433765	142	0,230544
163	1,588461	165	0,565243
168	1,503718	133	1,032339
254	1,40954	228	1,408251
137	1,290709	206	1,019643
227	1,36158	246	0,843433
153	1,181778	225	0,965501
140	1,029726	251	0,985536
180	1,107316	236	1,063316
157	1,371821	250	1,140406
66	1,287742	87	0,638794
178	0,893426	241	0,767209
215	1,209555	229	0,767583
183	0,706397	263	0,872551
189	0,693332	242	0,937086
156	0,718196	179	0,86269
182	0,919418	266	0,749384
218	0,309715	255	0,594364
214	0,554286	85	0,839064
138	0,521391	257	0,732967
200	0,814553	65	0,803789
245	0,722906	272	0,532563
274	0,709116	234	0,725385
169	0,310539	89	0,293778
273	0,49703	268	0,464134
152	0,711095	253	0,733071
216	0,513611	269	0,35332

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

248	0,379339	192	0,118276
173	0,500364	144	-0,10284
159	0,407629	401	-0,16503
203	0,124549	220	0,025352
139	0,310019	188	0,249945
231	0,330813	249	0,051694
243	0,167029	69	0,481175
86	0,327565	198	0,32474
258	0,513807	204	-0,35528
256	0,527353	233	-0,14174
166	0,462515	70	0,269484
226	0,462515	90	0,250808
160	0,762596	91	0,354384
67	0,762596	201	0,883399
194	0,594934	400	0,390412
185	0,567116	-	-
177	0,643005	-	-
270	0,646201	-	-
265	-9,2E-05	-	-
247	0,206685	-	-
162	0,292547	-	-
224	0,011099	-	-
170	0,199878	-	-
68	0,392709	-	-
264	0,05875	-	-
132	-0,07466	-	-
230	0,090109	-	-
267	-0,10416	-	-
142	-0,02743	-	-
165	0,610143	-	-
133	0,324701	-	-
228	0,450939	-	-
206	0,605493	-	-
246	0,602638	-	-
225	0,524551	-	-
251	0,51366	-	-
236	0,453766	-	-
250	0,388842	-	-
87	0,498994	-	-
241	0,441975	-	-
229	0,413607	-	-
263	0,390526	-	-
242	0,568566	-	-

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

179	0,365433	-	-
266	0,365931	-	-
255	0,276053	-	-
85	0,033443	-	-
257	0,169035	-	-
65	0,160793	-	-
272	0,284511	-	-
234	0,643368	-	-
89	0,956152	-	-
268	0,827055	-	-
253	0,789662	-	-
269	0,794531	-	-
192	0,78525	-	-
144	1,218955	-	-
401	0,673226	-	-
220	0,974234	-	-
188	0,982135	-	-
249	0,83474	-	-
69	0,570941	-	-
198	0,493125	-	-
204	0,60665	-	-
233	0,974259	-	-
70	0,427894	-	-
90	1,348635	-	-
91	0,727477	-	-
201	1,080248	-	-
400	1,632787	-	-

Nilai *Alpha*

No	Alpha	No	Alpha
167	0	224	0,070306443
197	0	170	0,061573893
205	0	68	0,046674423
174	0,005643711	264	0,063835904
210	0,000361251	132	0,072834559
163	0	230	0,057366069
168	0	267	0,072281376
254	0	142	0,065878622
137	0	165	0,043906208
227	0,00221318	133	0,059239298
153	0,015137742	228	0,047965389
140	0,023122542	206	0,037929974
180	0,01217279	246	0,04036741

157	0	225	0,044976521
66	0,006522243	251	0,044013795
178	0,026076451	236	0,049194623
215	0,007220694	250	0,053085487
183	0,03391869	87	0,05019683
189	0,035193972	241	0,053466786
156	0,043393645	229	0,053053092
182	0,018173994	263	0,056416322
218	0,051182769	242	0,042231333
214	0,041980732	179	0,055885639
138	0,037953433	266	0,05137457
200	0,031850569	255	0,058392625
245	0,033566918	85	0,071668237
274	0,039187726	257	0,065220632
169	0,056391727	65	0,069314629
273	0,045453027	272	0,054216187
152	0,030229473	234	0,039309334
216	0,038423914	89	0,023253297
248	0,048655547	268	0,028688742
173	0,039883234	253	0,027318127
159	0,046339292	269	0,023893351
203	0,062566988	192	0,030385191
139	0,048598967	144	0,006704023
231	0,046896391	401	0,041481171
243	0,061011195	220	0,014262639
86	0,048715319	188	0,020790095
258	0,037977912	249	0,029803341
256	0,03684764	69	0,044694081
166	0,043831553	198	0,048568089
226	0,043831553	204	0,043003555
160	0,026461883	233	0,025878537
67	0,026461883	70	0,053897977
194	0,035262965	90	0,009960938
185	0,038890935	91	0,037564896
177	0,035384715	201	0,019468347
270	0,036169577	400	0
265	0,066004291		
247	0,057128266		
162	0,048540581		

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

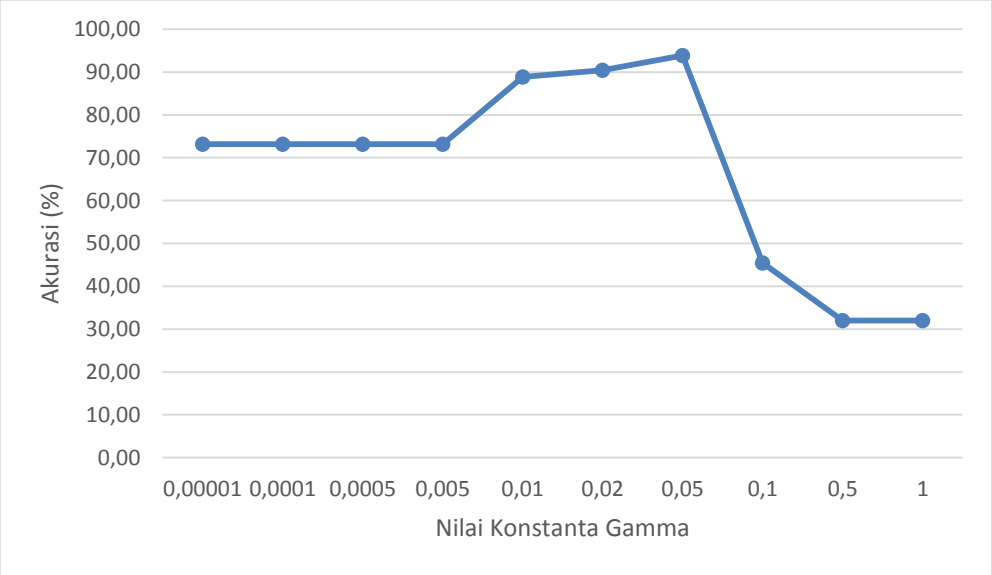
Nilai *bias*

Bias Level 1	Bias Level 2
-0,0471658	-0,295828819

Kelas Prediksi

No	Kelas Prediksi	Kelas Aktual
196	1	1
211	1	1
187	1	1
135	1	1
271	1	1
217	1	1
195	1	-1
221	1	-1
232	1	-1
199	-1	-1

Konstanta Gamma	**Percobaan Ke-*i***					**Rata-rata**
	1	**2**	**3**	**4**	**5**	
0,05	90,00	93,10	100,00	93,10	93,10	93,86



4. Pengujian Nilai *Epsilon*

Parameter	
rasio	90 : 10
Lambda	0.1
Konstanta Gamma	0.05
itermac	100
c	1
d	2

Nilai *Ei*

No	*Ei* Level 1	No	*Ei* Level 2
197	1,832585	264	0,641882
205	2,08284	195	0,654441
196	1,645386	132	0,281975
174	1,268078	230	0,081292
210	1,403642	142	0,141955
211	1,607393	165	0,518337
187	1,604034	133	1,001889
163	1,52527	228	1,408445
168	1,546122	206	1,030472
135	1,40628	246	0,775843
254	1,407521	225	0,928339
137	1,235152	221	0,922933
227	1,329048	251	0,992239
153	1,191376	236	1,056522
140	1,082694	250	1,163907
180	1,049096	87	0,611129
157	1,38403	241	0,768751
66	1,28845	263	0,90324
178	0,866854	242	0,945454
215	1,091147	179	0,92443
183	0,662551	232	1,040121
189	0,619863	255	0,669194
156	0,799037	85	0,825116
182	0,936542	257	0,689915
271	0,487009	65	0,720055
214	0,617013	272	0,457005
138	0,530996	234	0,640934
217	0,44585	89	0,198408
200	0,851274	268	0,44517

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

245	0,765759	253	0,757182
274	0,844865	269	0,325614
169	0,268439	192	0,085222
273	0,519989	144	0,035198
152	0,632949	401	-0,20075
248	0,376753	220	0,0029
173	0,404096	188	0,282814
159	0,386885	199	0,234671
203	0,16059	249	0,234671
139	0,296832	69	0,532631
231	0,291277	198	0,352614
243	0,184804	204	-0,33095
86	0,286016	233	-0,23921
258	0,471571	70	0,295121
166	0,419653	90	0,199505
160	0,696582	201	0,908132
67	0,696582	400	0,346023
194	0,589941	-	-
177	0,66253	-	-
270	0,665935	-	-
265	-0,00806	-	-
247	0,036407	-	-
162	0,227945	-	-
224	0,050391	-	-
170	0,219624	-	-
68	0,367021	-	-
264	0,122045	-	-
195	0,097381	-	-
132	-0,03518	-	-
230	0,057239	-	-
142	0,007077	-	-
165	0,717069	-	-
133	0,328573	-	-
228	0,532769	-	-
206	0,66436	-	-
246	0,589236	-	-
225	0,524994	-	-
221	0,363681	-	-
251	0,513533	-	-
236	0,469171	-	-
250	0,420286	-	-
87	0,524095	-	-
241	0,491444	-	-

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

263	0,465483	-	-
242	0,578367	-	-
179	0,439682	-	-
232	0,660025	-	-
255	0,288206	-	-
85	0,131223	-	-
257	0,240964	-	-
65	0,254248	-	-
272	0,285294	-	-
234	0,681653	-	-
89	0,965668	-	-
268	0,85062	-	-
253	0,817202	-	-
269	0,742607	-	-
192	0,757692	-	-
144	1,214457	-	-
401	0,786852	-	-
220	0,963993	-	-
188	0,983334	-	-
199	0,91754	-	-
249	0,91754	-	-
69	0,629687	-	-
198	0,567252	-	-
204	0,572645	-	-
233	0,979902	-	-
70	0,481867	-	-
90	1,444453	-	-
201	1,08025	-	-
400	1,738746	-	-

Nilai *Alpha*

No	Alpha	No	Alpha
167	0	162	0,048540581
197	0	224	0,070306443
205	0	170	0,061573893
174	0,005643711	68	0,046674423
210	0,000361251	264	0,063835904
163	0	132	0,072834559
168	0	230	0,057366069
254	0	267	0,072281376
137	0	142	0,065878622
227	0,00221318	165	0,043906208
153	0,015137742	133	0,059239298

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

140	0,023122542	228	0,047965389
180	0,01217279	206	0,037929974
157	0	246	0,04036741
66	0,006522243	225	0,044976521
178	0,026076451	251	0,044013795
215	0,007220694	236	0,049194623
183	0,03391869	250	0,053085487
189	0,035193972	87	0,05019683
156	0,043393645	241	0,053466786
182	0,018173994	229	0,053053092
218	0,051182769	263	0,056416322
214	0,041980732	242	0,042231333
138	0,037953433	179	0,055885639
200	0,031850569	266	0,05137457
245	0,033566918	255	0,058392625
274	0,039187726	85	0,071668237
169	0,056391727	257	0,065220632
273	0,045453027	65	0,069314629
152	0,030229473	272	0,054216187
216	0,038423914	234	0,039309334
248	0,048655547	89	0,023253297
173	0,039883234	268	0,028688742
159	0,046339292	253	0,027318127
203	0,062566988	269	0,023893351
139	0,048598967	192	0,030385191
231	0,046896391	144	0,006704023
243	0,061011195	401	0,041481171
86	0,048715319	220	0,014262639
258	0,037977912	188	0,020790095
256	0,03684764	249	0,029803341
166	0,043831553	69	0,044694081
226	0,043831553	198	0,048568089
160	0,026461883	204	0,043003555
67	0,026461883	233	0,025878537
194	0,035262965	70	0,053897977
185	0,038890935	90	0,009960938
177	0,035384715	91	0,037564896
270	0,036169577	201	0,019468347
265	0,066004291	400	0
247	0,057128266		

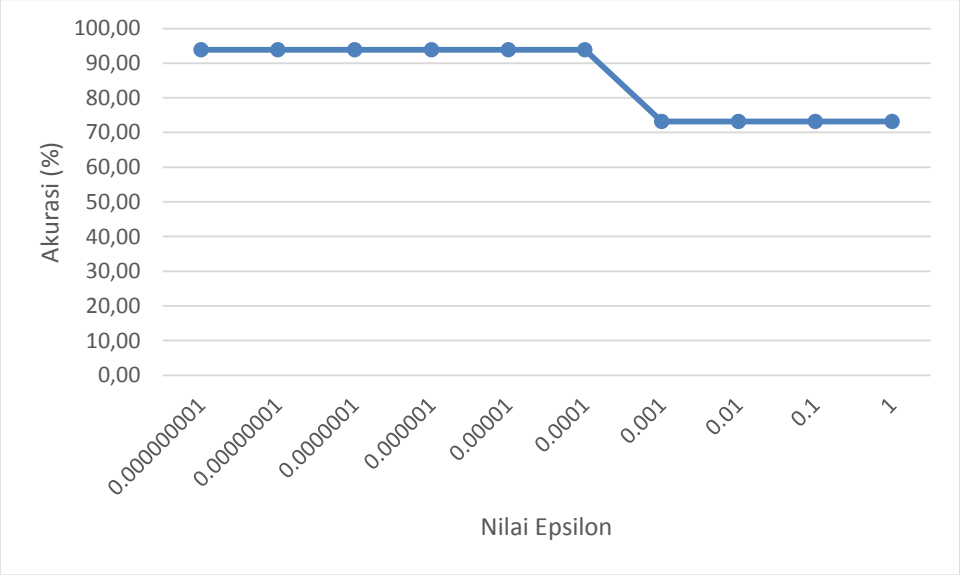
Nilai *bias*

Bias Level 1	Bias Level 2
-0,01754462	-0,161978543

Kelas Prediksi

No	Kelas Prediksi	Kelas Aktual
167	1	1
218	1	1
216	1	1
256	1	1
226	1	1
185	1	1
267	1	-1
229	1	-1
266	1	-1
91	-1	-1

Epsilon	Percobaan Ke-*i*					Rata-rata
	1	2	3	4	5	
0.0001	90,00	93,10	100,00	93,10	93,10	93,86



5. Pengujian Iterasi Maksimum

Parameter	
rasio	90 : 10
Lambda	0.1
Konstanta Gamma	0.05
epsilon	0.0001
c	1
d	2

Nilai *Ei*

No	*Ei* Level 1	No	*Ei* Level 2
167	6,592864	264	1,23106
197	3,949406	195	0,969845
205	3,971627	132	0,763024
196	3,009975	230	0,761358
174	3,495991	267	1,166236
210	4,050613	142	0,868241
211	3,353821	165	1,008247
187	3,472904	133	1,698595
163	3,01228	228	1,953422
168	3,209916	206	1,520452
135	3,270914	246	1,323491
254	3,061213	225	1,370502
137	2,313226	221	1,34156
227	2,750339	251	1,571359
153	2,362231	236	1,376353
180	2,099955	250	1,365836
157	2,68102	87	0,970721
66	2,387326	229	1,12076
178	1,831089	263	1,095069
215	1,891913	179	1,173181
183	1,442571	232	1,320477
189	1,192564	266	1,060397
182	2,089837	255	0,885243
271	1,585836	85	1,275476
218	1,007266	257	1,042823
214	1,452587	65	0,797179
217	1,197515	272	0,783772
200	1,338643	234	0,912812
245	1,414449	89	0,246807
274	1,131474	268	0,562246
169	0,980587	253	1,018251

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

273	1,033208	269	0,382911
152	1,183581	144	0,675387
216	0,805104	401	0,249434
248	0,769384	220	0,379961
159	0,72447	188	0,658869
139	0,771257	199	0,577681
231	0,644865	249	0,577681
243	0,604544	198	0,923838
86	0,584112	204	0,468296
258	0,960114	233	0,423693
256	0,853557	70	1,066794
166	0,873915	90	1,342542
160	0,982372	91	1,213135
67	0,982372	201	2,483762
194	1,023276	400	1,741037
185	0,783263	-	-
177	0,979832	-	-
270	0,98695	-	-
265	0,135438	-	-
247	0,654105	-	-
162	0,57805	-	-
224	0,058035	-	-
170	0,474504	-	-
68	0,499957	-	-
264	0,39075	-	-
195	0,473233	-	-
132	0,129539	-	-
230	0,604429	-	-
267	0,134418	-	-
142	0,33599	-	-
165	1,122453	-	-
133	0,516548	-	-
228	0,641532	-	-
206	0,923261	-	-
246	1,019184	-	-
225	0,988677	-	-
221	0,864053	-	-
251	1,035793	-	-
236	0,964527	-	-
250	0,934851	-	-
87	0,911897	-	-
229	0,964744	-	-
263	1,026766	-	-

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

179	1,011951	-	-
232	0,828401	-	-
266	0,801496	-	-
255	0,755327	-	-
85	0,588493	-	-
257	0,669982	-	-
65	0,659647	-	-
272	0,781278	-	-
234	1,182102	-	-
89	1,648024	-	-
268	1,534305	-	-
253	1,110438	-	-
269	1,51059	-	-
144	2,165068	-	-
401	1,751901	-	-
220	2,055238	-	-
188	2,013331	-	-
199	1,850016	-	-
249	1,850016	-	-
198	1,465259	-	-
204	1,513072	-	-
233	2,13403	-	-
70	1,368674	-	-
90	3,029334	-	-
91	1,950862	-	-
201	2,725514	-	-
400	3,718231	-	-

Nilai *Alpha*

No	Alpha	No	Alpha
167	0	162	0,366753
197	0	224	0,630745
205	0	170	0,469724
196	0	68	0,359081
174	0	264	0,500181
210	0	195	0,493879
211	0	132	0,667002
187	0	230	0,430158
163	0	267	0,665687
168	0	142	0,558518
135	0	165	0,031514
254	0	133	0,371701
137	0	228	0,259181

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

227	0	206	0,119361
153	0	246	0,089355
180	0	225	0,125735
157	0	221	0,250968
66	0	251	0,110822
178	0	236	0,156577
215	0	250	0,185784
183	0	87	0,173486
189	0,014933525	229	0,173078
182	0	263	0,157452
271	0	179	0,166175
218	0,171700016	232	0,18076
214	0	266	0,265109
217	0,118242159	255	0,310812
200	0	85	0,418477
245	0	257	0,351532
274	0,001380584	65	0,360531
169	0,228308275	272	0,288371
273	0,151763409	234	0,007722
152	0,023027679	89	0
216	0,22178179	268	0
248	0,282056689	253	0,019347
159	0,292501032	269	0
139	0,290263116	144	0
231	0,334499121	401	0
243	0,405831993	220	0
86	0,362834066	188	0
258	0,168514103	199	0
256	0,20505625	249	0
166	0,224209219	198	0
160	0,073474877	204	0
67	0,073474877	233	0
194	0,10209401	70	0,01457
185	0,208844155	90	0
177	0,092195183	91	0
270	0,091015995	201	0
265	0,641414404	400	0
247	0,381582409		

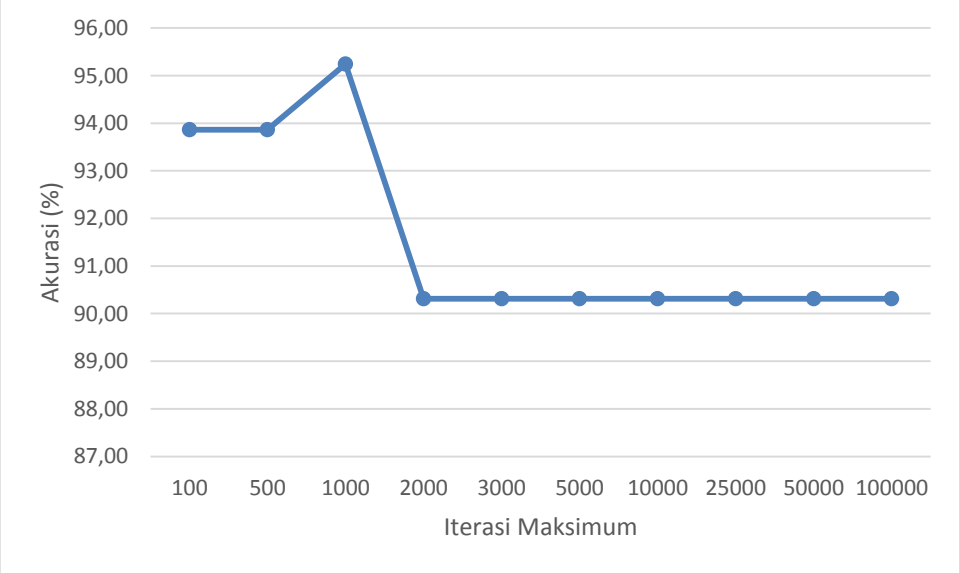
Nilai *bias*

Bias Level 1	Bias Level 2
-0,01784649	-0,011689484

Kelas Prediksi

No	Kelas Prediksi	Kelas Aktual
140	1	1
156	1	1
138	1	1
173	1	1
203	1	1
226	1	1
241	1	-1
242	1	-1
192	1	-1
69	-1	-1

Iterasi	Percobaan Ke-*i*					Rata-rata
	1	2	3	4	5	
1000	90,00	93,10	100,00	100,00	93,10	95,24



5. Pengujian Nilai *C* (*Complexity*)

Parameter	
rasio	90 : 10
Lambda	0.1
Konstanta Gamma	0.05
epsilon	0.0001
Iterasimax	1000
d	2

Nilai *Ei*

No	*Ei* Level 1	No	*Ei* Level 2
167	6,592864	264	1,23106
197	3,949406	195	0,969845
205	3,971627	132	0,763024
196	3,009975	230	0,761358
174	3,495991	267	1,166236
210	4,050613	142	0,868241
211	3,353821	165	1,008247
187	3,472904	133	1,698595
163	3,01228	228	1,953422
168	3,209916	206	1,520452
135	3,270914	246	1,323491
254	3,061213	225	1,370502
137	2,313226	221	1,34156
227	2,750339	251	1,571359
153	2,362231	236	1,376353
180	2,099955	250	1,365836
157	2,68102	87	0,970721
66	2,387326	229	1,12076
178	1,831089	263	1,095069
215	1,891913	179	1,173181
183	1,442571	232	1,320477
189	1,192564	266	1,060397
182	2,089837	255	0,885243
271	1,585836	85	1,275476
218	1,007266	257	1,042823
214	1,452587	65	0,797179
217	1,197515	272	0,783772
200	1,338643	234	0,912812
245	1,414449	89	0,246807
274	1,131474	268	0,562246
169	0,980587	253	1,018251

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

273	1,033208	269	0,382911
152	1,183581	144	0,675387
216	0,805104	401	0,249434
248	0,769384	220	0,379961
159	0,72447	188	0,658869
139	0,771257	199	0,577681
231	0,644865	249	0,577681
243	0,604544	198	0,923838
86	0,584112	204	0,468296
258	0,960114	233	0,423693
256	0,853557	70	1,066794
166	0,873915	90	1,342542
160	0,982372	91	1,213135
67	0,982372	201	2,483762
194	1,023276	400	1,741037
185	0,783263	-	-
177	0,979832	-	-
270	0,98695	-	-
265	0,135438	-	-
247	0,654105	-	-
162	0,57805	-	-
224	0,058035	-	-
170	0,474504	-	-
68	0,499957	-	-
264	0,39075	-	-
195	0,473233	-	-
132	0,129539	-	-
230	0,604429	-	-
267	0,134418	-	-
142	0,33599	-	-
165	1,122453	-	-
133	0,516548	-	-
228	0,641532	-	-
206	0,923261	-	-
246	1,019184	-	-
225	0,988677	-	-
221	0,864053	-	-
251	1,035793	-	-
236	0,964527	-	-
250	0,934851	-	-
87	0,911897	-	-
229	0,964744	-	-
263	1,026766	-	-

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

179	1,011951	-	-
232	0,828401	-	-
266	0,801496	-	-
255	0,755327	-	-
85	0,588493	-	-
257	0,669982	-	-
65	0,659647	-	-
272	0,781278	-	-
234	1,182102	-	-
89	1,648024	-	-
268	1,534305	-	-
253	1,110438	-	-
269	1,51059	-	-
144	2,165068	-	-
401	1,751901	-	-
220	2,055238	-	-
188	2,013331	-	-
199	1,850016	-	-
249	1,850016	-	-
198	1,465259	-	-
204	1,513072	-	-
233	2,13403	-	-
70	1,368674	-	-
90	3,029334	-	-
91	1,950862	-	-
201	2,725514	-	-
400	3,718231	-	-

Nilai *Alpha*

No	Alpha	No	Alpha
167	0	162	0,366753
197	0	224	0,630745
205	0	170	0,469724
196	0	68	0,359081
174	0	264	0,500181
210	0	195	0,493879
211	0	132	0,667002
187	0	230	0,430158
163	0	267	0,665687
168	0	142	0,558518
135	0	165	0,031514
254	0	133	0,371701
137	0	228	0,259181

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

227	0	206	0,119361
153	0	246	0,089355
180	0	225	0,125735
157	0	221	0,250968
66	0	251	0,110822
178	0	236	0,156577
215	0	250	0,185784
183	0	87	0,173486
189	0,014933525	229	0,173078
182	0	263	0,157452
271	0	179	0,166175
218	0,171700016	232	0,18076
214	0	266	0,265109
217	0,118242159	255	0,310812
200	0	85	0,418477
245	0	257	0,351532
274	0,001380584	65	0,360531
169	0,228308275	272	0,288371
273	0,151763409	234	0,007722
152	0,023027679	89	0
216	0,22178179	268	0
248	0,282056689	253	0,019347
159	0,292501032	269	0
139	0,290263116	144	0
231	0,334499121	401	0
243	0,405831993	220	0
86	0,362834066	188	0
258	0,168514103	199	0
256	0,20505625	249	0
166	0,224209219	198	0
160	0,073474877	204	0
67	0,073474877	233	0
194	0,10209401	70	0,01457
185	0,208844155	90	0
177	0,092195183	91	0
270	0,091015995	201	0
265	0,641414404	400	0
247	0,381582409		

Nilai *bias*

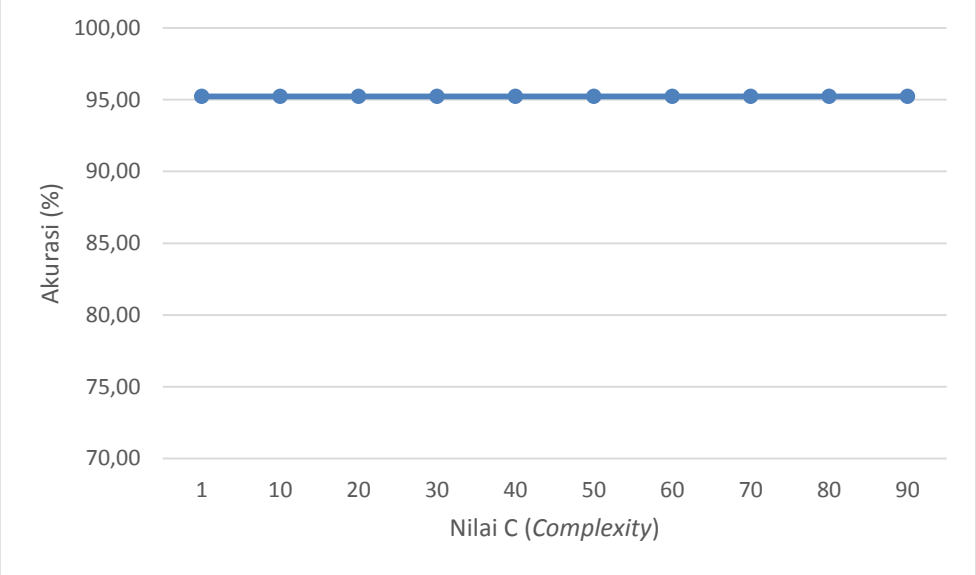
Bias Level 1	Bias Level 2
-0,01784649	-0,011689484

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

Kelas Prediksi

No	Kelas Prediksi	Kelas Aktual
140	1	1
156	1	1
138	1	1
173	1	1
203	1	1
226	1	1
241	1	-1
242	1	-1
192	1	-1
69	-1	-1

C	Percobaan Ke-*i*					Rata-rata
	1	2	3	4	5	
1	90,00	93,10	100,00	100,00	93,10	95,24



Lampiran H Surat Izin Penelitian Pemerintah Kabupaten Kediri

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 metodeSVM 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, 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 I Bobot Metode TOPSIS

BOBOT METODE TOPSIS

PEMBOBOTAN	
KRITERIA	**PERSENTASE**
Penghasilan	15%
Jumlah Tanggungan	15%
Pekerjaan	5%
Daya Listrik	10%
Biaya Listrik	10%
Jumlah Rumah	5%
Stattus Rumah	5%
Status Dinding	5%
Status Lantai	5%
PBB	10%
Jumlah Motor	5%
Jumlah Mobil	5%
Program beasiswa	5%
Total	100%

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

DR. Mokh. Muhsin, M.Pd
NIP.

Lampiran J 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

PENGURUS KAB. KEDIRI LEMBAGA GNOTA

DR. Mokh. Muhsin, M.Pd
NIP.

Lampiran K Surat Persetujuan Menjadi Pakar

SURAT PERSETUJUAN MENJADI PAKAR

Saya yang bertanda tangan di bawah ini :

Nama : AJI INDRA LAKSANA
Pekerjaan : KARYAWAN GNOTA KEDIRI

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

Nama : M. Gilvy Langgawan Putra
NIM : 145150209111011
Jurusan : Teknik Informatika
Fakultas : Fakultas Ilmu Komputer
Universitas : Universitas Brawijaya Malang
Judul Skripsi : Seleksi dan Rekomendasi Penerima Beasiswa di Lembaga GNOTA Kediri Dengan Implementasi Metode SVM dan TOPSIS

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.

repository.ub.ac.id

UNIVERSITAS BRAWIJAYA