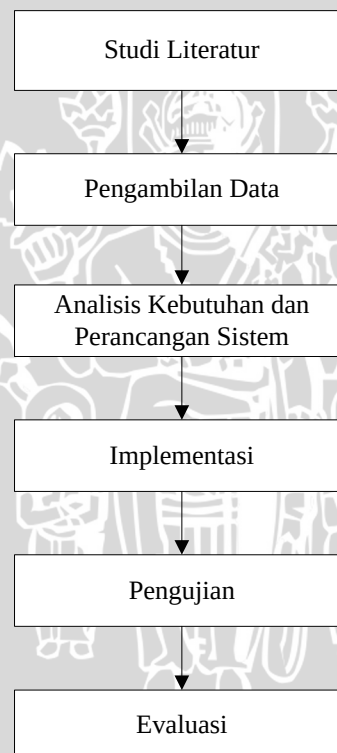


BAB III

METODOLOGI PENELITIAN

Bab ini berisi tentang langkah-langkah yang digunakan dalam penelitian. Tahapan yang dilakukan dalam penelitian ini meliputi studi literatur, penyusunan dasar teori, analisis sistem, perancangan sistem, perancangan antar muka, perhitungan secara manual dan pengujian sistem. Gambar 3.1 merupakan diagram Alir yang menjelaskan langkah-langkah yang dilakukan dalam penelitian ini.



Gambar 3.1 Diagram Alir Tahapan Penelitian

Penjelasan Gambar 3.1 Diagram Alir Tahapan Penelitian adalah sebagai berikut.

1. Mempelajari studi literatur yang berhubungan dengan metode yang dikembangkan, yaitu metode AHP-SVM untuk klasifikasi penerima RASKIN.

2. Melakukan pengumpulan data yang dibutuhkan dalam penunjang penulisan skripsi ini.
3. Melakukan analisis kebutuhan dan perancangan sistem yang berhubungan dengan mengembangkan aplikasi yang akan dibuat.
4. Melakukan implementasi aplikasi berdasarkan analisa dan perancangan yang telah dibuat.
5. Melakukan pengujian untuk klasifikasi penerima RASKIN dengan menggunakan metode AHP-SVM, serta mengukur tingkat akurasi terhadap aplikasi yang telah dibuat.
6. Melakukan evaluasi dan analisis hasil pengujian aplikasi.

3.1 Studi Literatur

Studi literatur mempelajari tentang penjelasan dasar teori yang digunakan untuk menunjang dalam penulisan skripsi. Teori pendukung tersebut diperoleh dari buku, jurnal, *e-book*, dan penelitian yang pernah dilakukan sebelumnya yang terkait dengan topik dalam skripsi ini. Dasar teori yang digunakan untuk menunjang dalam penyusunan penulisan skripsi ini adalah Metode *Analytical Hierarchy Process* (AHP), dan Metode SVM (*Support Vector Machine*).

3.2 Pengambilan Data

Data yang digunakan dan diolah perangkat lunak ini adalah data masyarakat yang menerima RASKIN di Kelurahan Ronggomulyo Kabupaten Tuban. Metode yang dilakukan dalam pengumpulan data ada metode primer, yaitu dengan mengumpulkan data dari sumber secara langsung melalui interview dan observasi lapangan. Data yang diperoleh dari hasil observasi lapangan berisi tentang data kondisi umum keluarga yang meliputi data pekerjaan, penghasilan, kondisi fisik rumah, jumlah tanggungan anak, dan jumlah aset yang dimiliki, baik aset pribadi berupa kendaraan bermotor, maupun aset dalam bentuk barang-barang elektronik. Dari data yang diperoleh kemudian dikelompokkan berdasarkan kriteria yang menjadi prioritas dalam penentuan masyarakat yang berhak untuk menerima RASKIN. Surat Keterangan Hasil Penelitian di Kelurahan Ronggomulyo Kabupaten Tuban dapat dilihat pada Lampiran 1. Format *Quisioner* hasil survey dapat dilihat pada Lampiran 2.

3.3 Analisis Kebutuhan dan Perancangan Sistem

Pada bab ini menjelaskan tentang kebutuhan dan perancangan yang digunakan untuk membangun aplikasi klasifikasi penerima RASKIN.

3.3.1 Analisis Kebutuhan

Analisis kebutuhan ini dilakukan untuk mengetahui kebutuhan apa saja yang diperlukan dalam membangun sistem. Pada tahap ini yang dilakukan analisis terhadap semua spesifikasi kebutuhan yang diinginkan oleh pengguna untuk membangun sistem. Hasil dari analisis kebutuhan tersebut berupa informasi yang akan digunakan dalam membangun sistem agar sesuai dengan harapan pengguna.

Berikut ini adalah analisis kebutuhan dalam implementasi AHP-SVM untuk klasifikasi penerima RAKSIN di Kelurahan Ronggomulyo Kabupaten Tuban.

Perangkat lunak yang dibangun memiliki fungsi sebagai berikut.

1. Perangkat lunak yang dikembangkan harus memiliki tampilan dapat memudahkan pengguna dalam menggunakan program tersebut.
2. Perangkat lunak dapat melakukan proses pembobotan berdasarkan kriteria yang digunakan dalam menentukan masyarakat penerima RASKIN.
3. Perangkat lunak dapat menghasilkan keputusan untuk masing-masing kelas dalam menentukan kelayakan masyarakat penerima RASKIN di Kelurahan Ronggomulyo Kabupaten Tuban.

3.3.2 Perancangan Sistem

Pada bagian perancangan ini menjelaskan tentang tahap yang dilakukan untuk membentuk sistem agar sesuai dengan harapan. Sistem yang akan dibangun adalah dengan menggunakan metode AHP-SVM (*Support Vector Machine*). Hasil yang akan diperoleh dalam perancangan sistem ini adalah hasil klasifikasi masyarakat yang layak dan tidak layak menerima RASKIN.

3.3.2.1 Deskripsi Umum

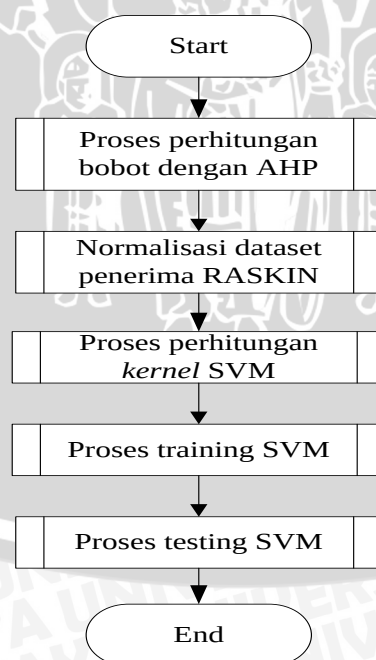
Aplikasi yang dibuat ini adalah perangkat lunak yang mengimplementasikan metode AHP-SVM untuk klasifikasi penerima RASKIN di

Kelurahan Ronggomulyo Kabupaten Tuban. Data yang digunakan dalam aplikasi ini adalah data yang berasal dari hasil interview dan observasi lapangan pada Kelurahan Ronggomulyo. Data penerima RASKIN tersebut akan diolah berdasarkan kriteria tertentu. Untuk menentukan kriteria tertentu dilakukan pembobotan dengan menggunakan metode AHP. Kriteria terpilih dari proses AHP digunakan untuk proses klasifikasi dengan menggunakan metode SVM.

Dataset penerima RASKIN akan dinormalisasi untuk menghasikan data dengan nilai lebih kecil, sehingga dapat mempermudah dalam proses komputasi. Setelah data dinormalisasi, langkah selanjutnya adalah proses *training* dan *testing* dengan menggunakan metode SVM. Proses *training* pada SVM dilakukan dengan metode *sequential training* yaitu untuk menemukan bidang pemisah atau *support vector* pada *dataset* penerima RASKIN. Kemudian data diuji dengan menggunakan *dataset* untuk mendapatkan klasifikasi kelayakan penerima RASKIN.

3.3.2.2 Alir Perancangan Sistem

Alir Perancangan Sistem untuk klasifikasi penerima RASKIN dapat dilihat pada Gambar 3.2.

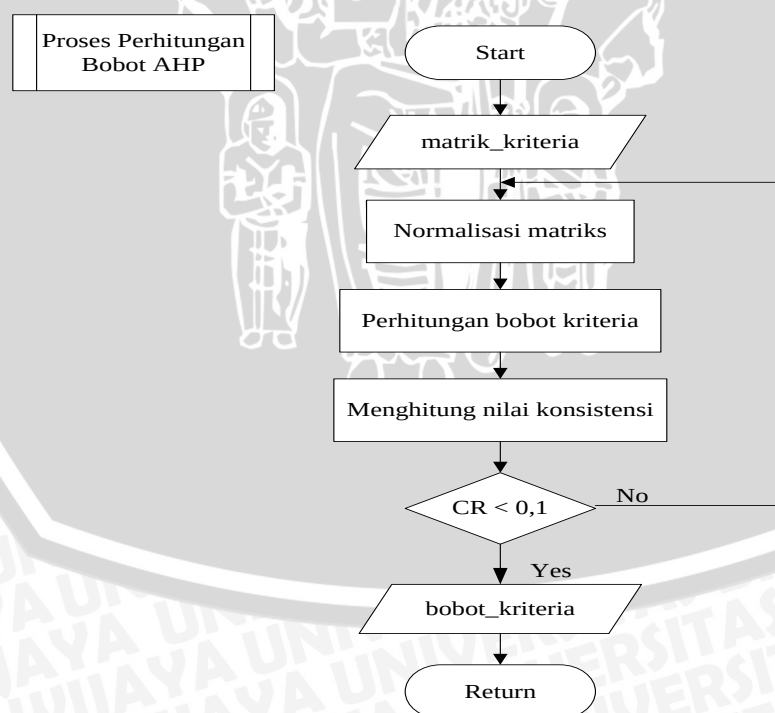


Gambar 3.2 Diagram Alir Perancangan Sistem

Pada Gambar 3.2 dijelaskan tentang Diagram Perancangan Sistem untuk klasifikasi penerima RASKIN dengan metode AHP-SVM. Proses klasifikasi dimulai dari perhitungan bobot kriteria dengan menggunakan metode AHP. Setelah diperoleh nilai dari masing-masing bobot kriteria terpilih dilakukan proses normalisasi untuk data penerima RASKIN. Kemudian memilih data yang akan digunakan sebagai data *training* maupun data *testing* untuk perhitungan SVM. Proses SVM dimulai dengan perhitungan *kernel* yang akan digunakan. Proses selanjutnya perhitungan data *training* menggunakan metode *sequential training*. Hasil perhitungan dengan proses *training* akan diperoleh parameter SVM. Dari parameter tersebut dapat dilakukan pengujian dengan menggunakan data *testing* untuk mendapatkan hasil klasifikasi dari penerima RASKIN. Setelah proses pengujian selesai, dilakukan evaluasi untuk mengukur nilai akurasi sistem.

3.3.2.3 Proses Perhitungan Bobot Kriteria dengan metode AHP

Pada Gambar 3.3 adalah diagram Alir proses perhitungan konsistensi dan bobot dengan menggunakan metode AHP.



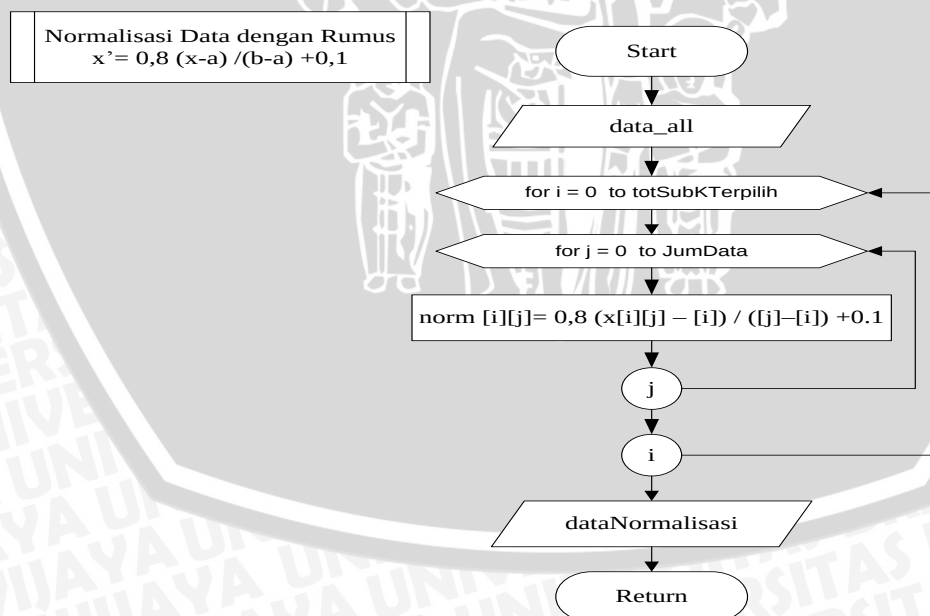
Gambar 3.3 Diagram Alir Konsistensi dan Bobot dengan AHP

Berikut adalah penjelasan tahapan dari proses AHP.

1. Menginputkan matriks perbandingan berpasangan berdasarkan kriteria yang akan digunakan. Kemudian melakukan normalisasi dari matriks perbandingan tersebut.
2. Selanjutnya menghitung bobot kriteria dengan menjumlahkan nilai tiap baris kemudian membagi dengan jumlah elemen yang ada sehingga didapatkan nilai rata-rata.
3. Mengukur nilai konsistensi berdasarkan pertimbangan pengambilan suatu keputusan dengan mengukur nilai *Consistency Index* (CI) dan menghitung rasio konsistensi atau *Consistency Ratio* (CR).
4. Apabila kurang atau sama dengan 0.1 maka perhitungan tersebut benar. Namun jika nilai CR lebih dari 0.1 maka dilakukan perhitungan ulang.
5. *Output* yang dihasilkan adalah nilai bobot kriteria yang kemudian akan dipilih berdasarkan nilai *threshold* tertentu.

3.3.2.4 Proses Normalisasi Dataset

Pada Gambar 3.4 adalah diagram Alir proses Normalisasi dataset.



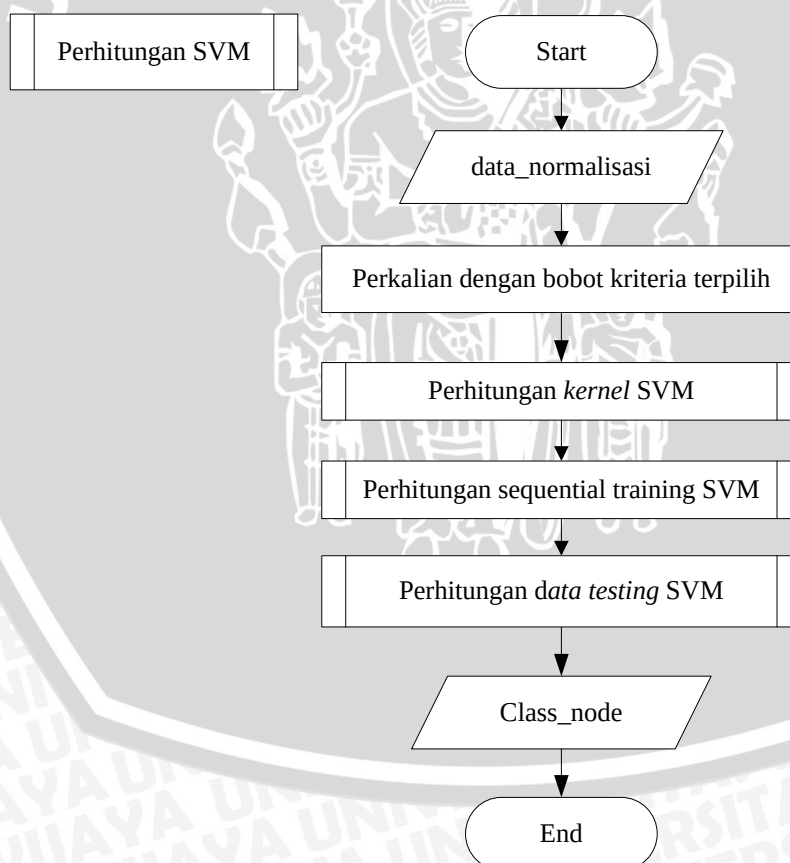
Gambar 3.4 Diagram Alir Normalisasi Dataset

Pada Gambar 3.4 dijelaskan tentang tahapan yang dilakukan untuk proses normalisasi. Berikut adalah penjelasan tahapan dari proses normalisasi.

1. Menginputkan *dataset* awal penerima RASKIN yang akan digunakan berdasarkan nilai bobot kriteria terpilih.
2. Perulangan terhadap jumlah banyaknya sub kriteria terpilih dan perulangan terhadap jumlah data untuk proses perhitungan Normalisasi.
3. Menghitung nilai normalisasi dengan menggunakan Persamaan 2.1.
4. *Output* yang dihasilkan adalah data yang telah ternormalisasi.

3.3.2.5 Proses Perhitungan SVM

Pada proses ini akan dijelaskan tentang tahapan yang dilakukan dalam proses SVM. Proses perhitungan SVM dapat dilihat pada Gambar 3.5 Diagram Alir Proses SVM.



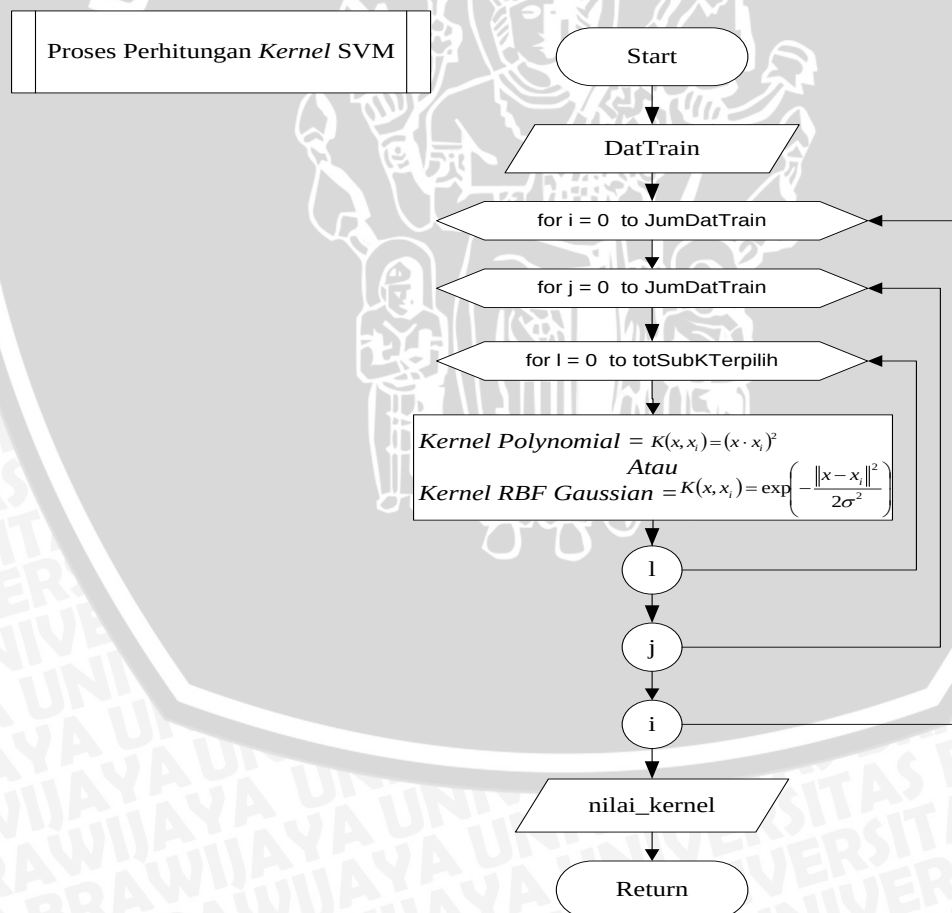
Gambar 3.5 Diagram Alir Proses SVM

Pada Gambar 3.5 dijelaskan tentang tahapan yang dilakukan dalam proses SVM. Berikut adalah penjelasan tahapan dari proses SVM.

1. Melakukan input *dataset* yang telah dinormalisasi.
2. Melakukan perkalian *dataset* dengan nilai bobot kriteria terpilih.
3. Proses perhitungan *kernel* yang digunakan dalam SVM.
4. Proses perhitungan *dataset training* SVM dengan menggunakan metode *sequential training* SVM.
5. Proses selanjutnya perhitungan data terpilih untuk dilakukan *testing*.
5. *Output* yang dihasilkan adalah berupa klasifikasi atau *predicted class*.

3.3.2.6 Proses Perhitungan Kernel SVM

Proses perhitungan *kernel* SVM dapat dilihat pada Gambar 3.6 Diagram Alir perhitungan *kernel* SVM.



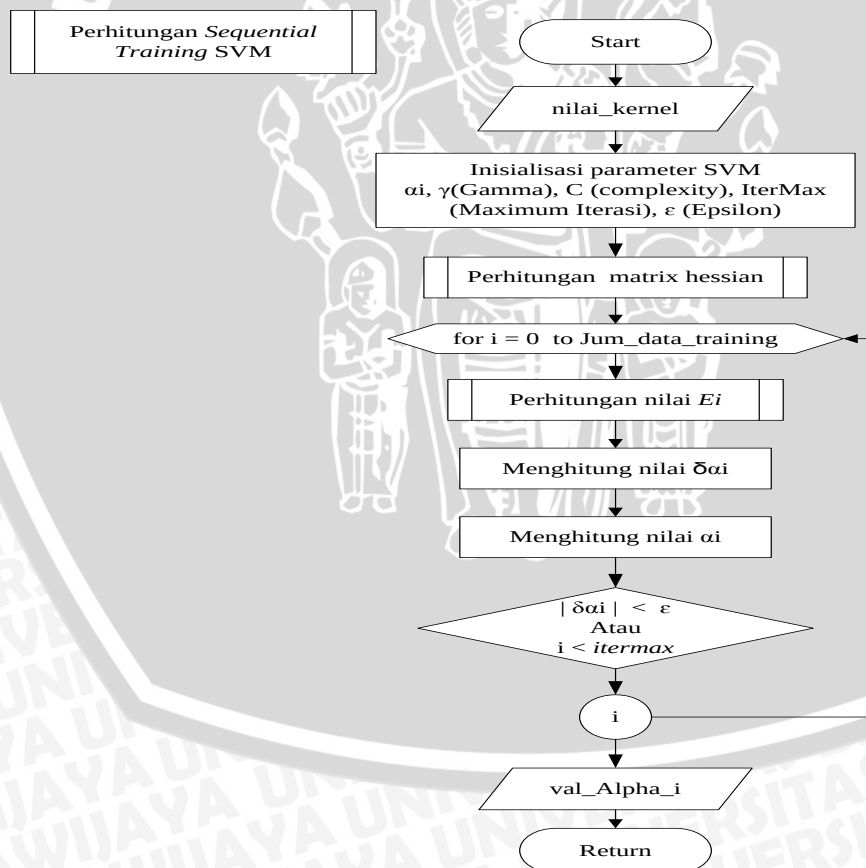
Gambar 3. 6 Diagram Alir Proses Perhitungan Kernel SVM

Pada Gambar 3.6 dijelaskan tahapan yang dilakukan dalam proses perhitungan *kernel* SVM. Berikut adalah penjelasan proses perhitungan *kernel* SVM.

1. Melakukan input data *training* yang telah dinormalisasi dan dikalikan bobot kriteria terpilih.
2. Perulangan terhadap jumlah banyaknya jumlah data *training* dan dan perulangan terhadap sub kriteria terpilih.
3. Proses perhitungan *kernel* yang digunakan dalam SVM. *Kernel* yang digunakan adalah *polynomial* atau *Gaussian RBF*.
4. *Output* yang dihasilkan adalah berupa nilai *kernel*.

3.3.2.7 Proses Perhitungan Metode *Sequential Training* SVM

Proses perhitungan Metode *Sequential Training* dapat dilihat pada Gambar 3.7 Diagram Alir perhitungan *Sequential Training* SVM.



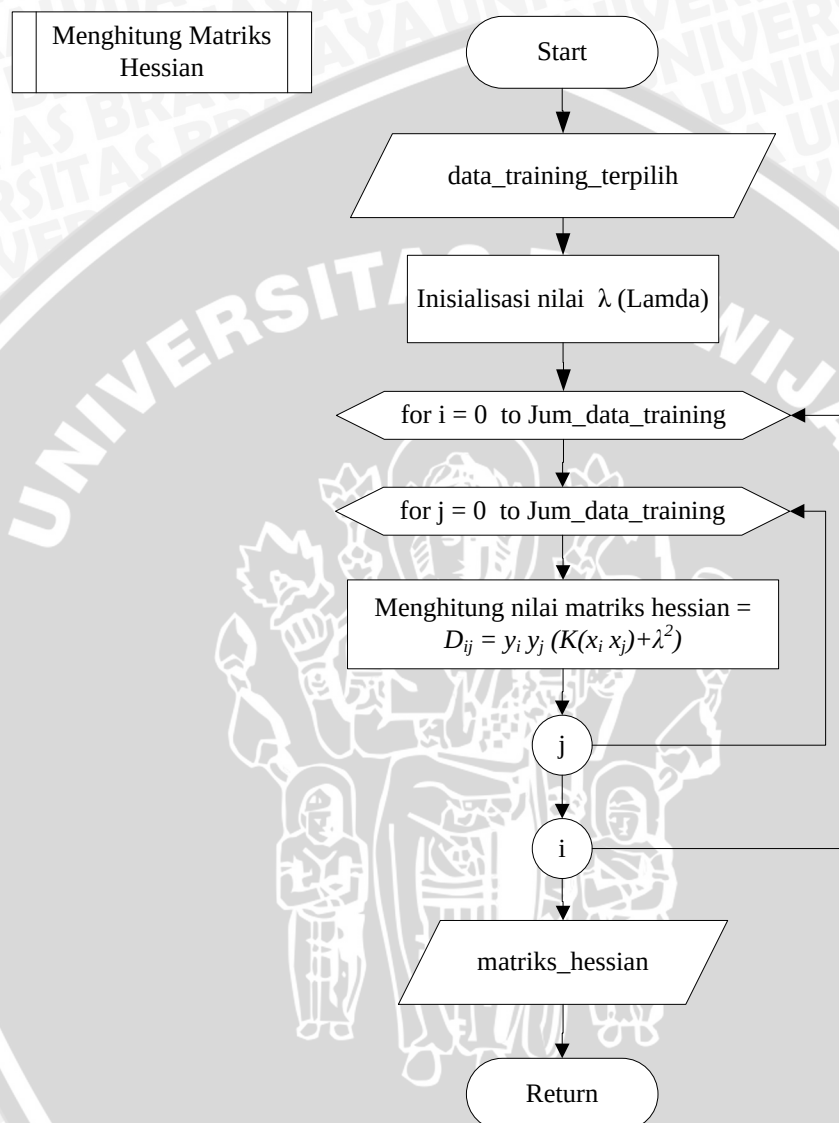
Gambar 3.7 Diagram Alir Proses *Sequential Training* SVM

Pada Gambar 3.7 dijelaskan tentang tahapan yang dilakukan dalam proses *sequential training* SVM. Berikut adalah penjelasan tahapan dari proses *training* SVM.

1. Menginputkan data bobot yang merupakan hasil perkalian data normalisasi dengan bobot.
2. Kemudian melakukan inisialisasi parameter *Support Vector Machine* yaitu nilai α_i (fungsi *Lagrange*), γ (*Gamma*) ε (*slack variable*) , Itermax (maximum iterasi) dan nilai C (*Complexity*). Data dipetakan kedalam *dot product kernel* SVM, dan *kernel* yang digunakan adalah *kernel polynomial* dan *kernel Gaussian RBF*.
3. Proses awal dari *sequential training* adalah perhitungan Matriks Hessian sesuai dengan Persamaan 2.21.
4. Melakukan iterasi sejumlah data untuk proses *sequential training* SVM. Banyaknya iterasi ditentukan ketika *user* menginputkan nilai *epsilon* (ε).
5. Proses perhitungan *sequential training* langkah pertama untuk menentukan nilai E_i sesuai dengan Persamaan 2.22.
6. Proses *sequential training* langkah kedua untuk menghitung nilai $\delta\alpha_i$ sesuai dengan Persamaan 2.23.
7. Proses *sequential training* langkah ketiga untuk menghitung nilai α_i baru sesuai dengan Persamaan 2.24.
8. Iterasi yang dilakukan pada data *training* jika telah mencapai nilai konvergen ($|\delta\alpha_i| < \varepsilon$), atau mencapai nilai *itermax* maka iterasi akan dihentikan. Proses diulangi sampai α mencapai nilai konvergen. Konvergen dapat didefinisikan dari tingkat perubahan pada nilai α .
9. *Output* yang dihasilkan adalah variabel α_i fungsi *hyperplane* (pemisah) yang optimal yaitu perubahan nilai α_i yang akan digunakan untuk menentukan *Support Vector*. Fungsi pemisah tersebut akan digunakan dalam proses *testing*.

3.3.2.8 Proses Perhitungan Matriks Hessian

Proses perhitungan Matriks dapat dilihat pada Gambar 3.8 Diagram Alir perhitungan Matriks Hessian.



Gambar 3.8 Diagram Alir Proses Perhitungan Matriks Hessian

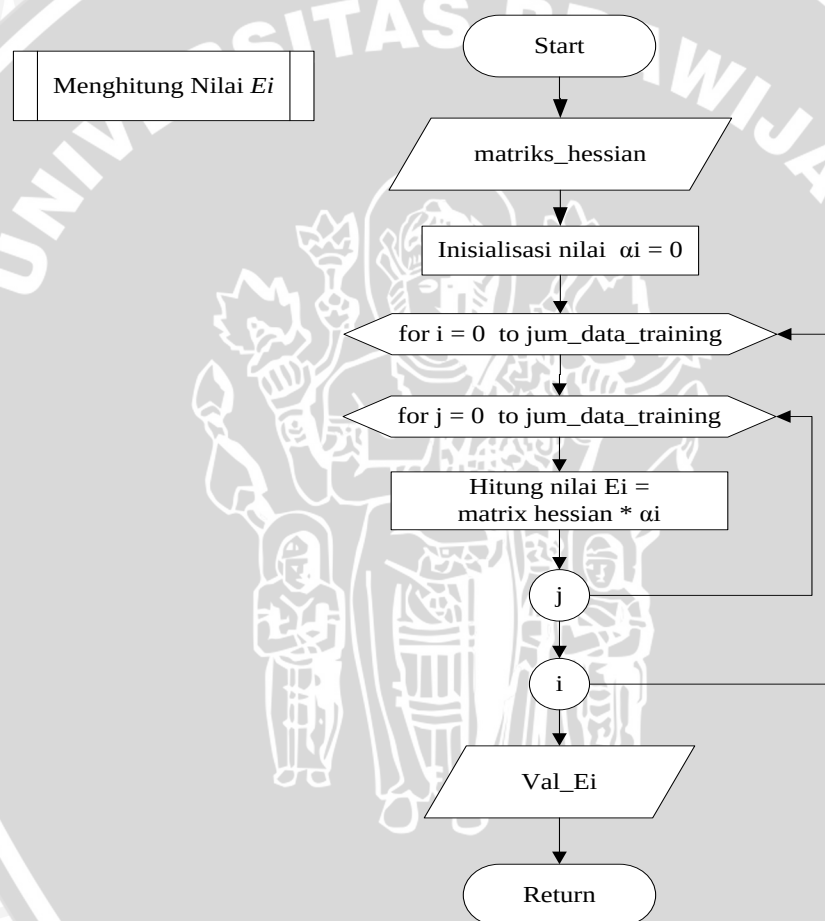
Pada Gambar 3.8 dijelaskan tentang tahapan yang dilakukan dalam proses perhitungan Matriks Hessian. Berikut adalah penjelasan tahapan dari proses perhitungan.

1. Menginputkan data *training* yang terpilih berdasarkan kriteria tertentu.
2. Kemudian melakukan inisialisasi parameter Matriks Hessian yaitu nilai λ (*lamda*).

3. Proses perhitungan Matriks Hessian sesuai dengan Persamaan 2.21 .
4. Perulangan terhadap jumlah data *training* untuk proses perhitungan Matriks Hessian.
5. *Output* yang dihasilkan adalah nilai Matriks Hessian.

3.3.2.9 Proses Perhitungan nilai E_i

Proses perhitungan nilai E_i dapat dilihat pada Gambar 3.9 Diagram Alir perhitungan nilai E_i .



Gambar 3.9 Diagram Alir Proses Perhitungan Nilai E_i

Pada Gambar 3.9 dijelaskan tentang tahapan yang dilakukan dalam proses perhitungan nilai E_i . Berikut adalah penjelasan tahapan dari proses perhitungan.

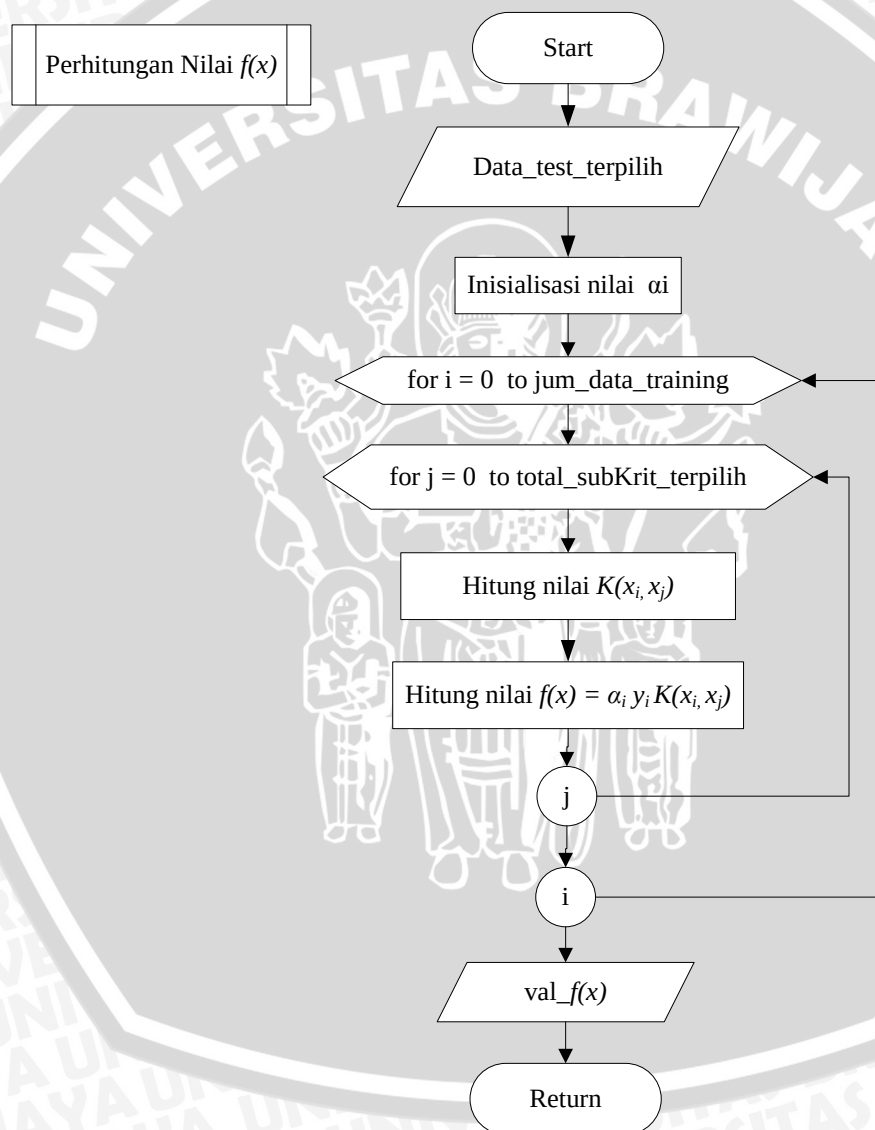
1. Menginputkan data matriks Hessian.
2. Melakukan inisialisasi nilai $\alpha_i = 0$.

3. Proses perhitungan E_i sesuai dengan Persamaan 2.22 .
4. Perulangan terhadap data *training* untuk proses perhitungan nilai E_i .
5. *Output* yang dihasilkan adalah nilai E_i .

3.3.2.10 Proses Perhitungan Nilai $f(x)$ pada Data Testing SVM

Proses perhitungan nilai $f(x)$ pada data *testing* dapat dilihat pada Gambar

3. 10 Diagram Alir perhitungan nilai $f(x)$ pada data *testing*.



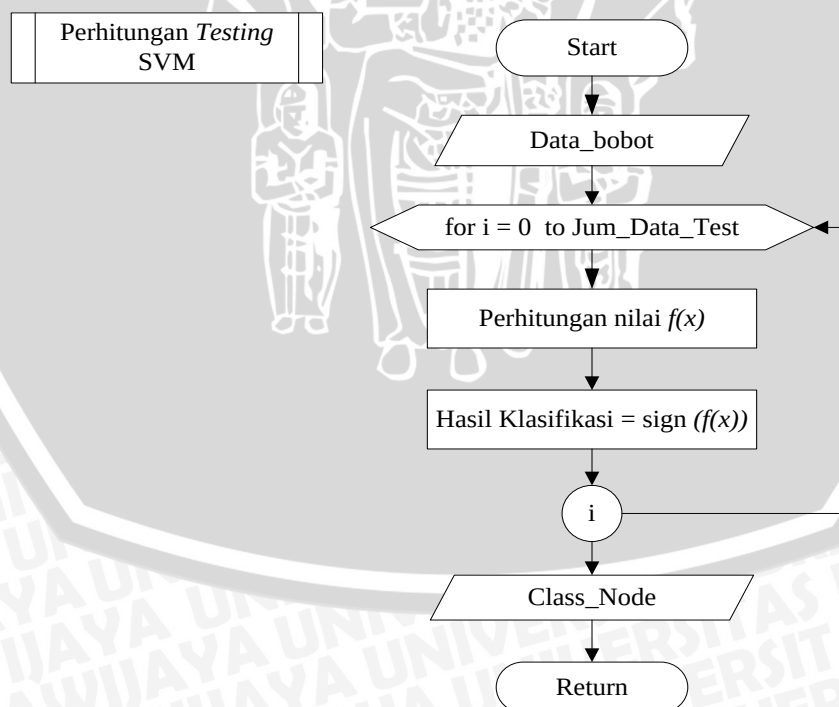
Gambar 3.10 Diagram Alir Proses Data Testing dengan Metode SVM

Pada Gambar 3.10 akan dijelaskan tentang tahapan yang dilakukan dalam proses perhitungan nilai $f(x)$ pada data *testing* SVM. Berikut adalah penjelasan tahapan dari proses perhitungan nilai $f(x)$.

1. Menginputkan *dataset testing* terpilih.
2. Melakukan inisialisasi terhadap nilai α_i yang diperoleh dari proses iterasi terakhir perhitungan *sequential training*.
3. Perhitungan nilai $K(x_i, x_j)$ terhadap data *testing*.
4. Proses selanjutnya adalah perhitungan $f(x) = \alpha_i y_i K(x_i, x_j)$. Nilai y_i adalah nilai *class* dari data *testing*.
5. Proses perulangan terjadi pada data *training* dan total dari sub kriteria terpilih.
6. *Output* yang dihasilkan dari proses *testing* SVM adalah nilai $f(x)$ data *testing*.

3.3.2.11 Proses Perhitungan Data *Testing* dengan Metode SVM

Proses perhitungan Data *Testing* dapat dilihat pada Gambar 3.10 Diagram Alir perhitungan data *testing*.



Gambar 3.11 Diagram Alir Proses Data *Testing* dengan Metode SVM

Pada Gambar 3.11 akan dijelaskan tentang tahapan yang dilakukan dalam proses *testing* SVM. Berikut adalah penjelasan tahapan dari proses *testing* SVM.

1. Menginputkan data bobot kriteria.
2. Data tersebut akan diuji dengan fungsi *hyperplane* yang diperoleh dari perhitungan *sequential training* SVM, sehingga dilakukan proses perulangan terhadap data *training*.
3. Data dipetakan dan dihitung berdasarkan fungsi $f(x)$ yang akan digunakan.
4. Proses selanjutnya adalah untuk menghitung hasil klasifikasi yaitu dengan fungsi $\text{sign}(f(x))$ pada masing-masing data *testing* terpilih.
5. Proses perulangan terjadi pada jumlah data *testing*.
6. *Output* yang dihasilkan dari proses *testing* SVM adalah nilai klasifikasi *class* atau *predicted class* dari setiap data *testing*.

3.4 Perhitungan Manual

Proses perhitungan manual dengan menggunakan dataset dari data *sample* penerima RASKIN yang terdiri dari 39 atribut atau kriteria dan akan dibagi menjadi dua kelas, yaitu layak dan tidak layak menerima RASKIN. Data yang digunakan diperoleh berdasarkan hasil observasi dan wawancara secara langsung dengan warga Kelurahan Ronggomulyo Kabupaten Tuban. Data warga hasil *quisioner* dapat dilihat pada Lampiran 3 Tabel Rekap data hasil *quisioner* warga. Data tersebut diubah berdasarkan parameter penilaian tertentu dari masing-masing kriteria. Dengan *range* parameter antara 0-5 dari tiap kriteria. Data kemudian dinormalisasi dengan *interval* antara [0.1,0.9] sesuai dengan Persamaan 2.1. Kelas layak menerima RASKIN akan diberi nilai -1, dan kelas untuk masyarakat yang tidak menerima RASKIN akan diberi nilai 1. Pada perhitungan manual data yang akan digunakan dengan menggunakan perbandingan rasio 60:40 dari 20 *dataset*, yang terdiri dari 11 *dataset training* dan 9 *dataset testing*. Data *sample* yang akan digunakan sebagai sebagai *dataset training* dan *dataset testing* dapat dilihat pada Tabel 3.1 *Sample Dataset*. Untuk Tabel secara lebih jelas dapat dilihat pada Lampiran 4 Tabel Data *sample* untuk data *training* dan data *testing*.

Tabel 3.1 Sample Dataset

	K1	K2	K3	K4	K5	K6	K7		K37	K38	K39	Actual Class
KL_1	0	4	0	0	0	0	3		0	0	0	-1
KL_2	5	0	0	0	0	0	0		0	0	0	-1
KL_3	0	0	0	2	0	0	3		0	0	0	-1
KL_4	0	4	0	0	0	4	0		0	0	0	-1
KL_5	0	0	0	2	0	4	0		0	0	0	-1
KL_6	0	4	0	0	0	4	0		0	0	0	-1
KL_7	0	4	0	0	0	4	0		0	0	0	-1
KL_8	0	4	0	0	0	4	0		0	0	0	-1
KL_9	0	0	0	0	1	0	0		0	0	1	1
KL_10	0	0	0	2	0	0	0		0	2	0	1
KL_11	0	0	0	0	1	0	0		0	0	1	1
KL_12	0	0	0	2	0	0	0		0	2	0	1
KL_13	0	0	0	2	0	0	0		0	2	0	1
KL_14	0	0	0	0	1	0	0		0	0	1	1
....	...	...				...	...					
KL_18	0	0	0	2	0	0	3		0	0	0	-1
KL_19	0	0	0	2	0	0	3		0	2	0	1
KL_20	0	0	0	2	0	0	3		0	2	0	1

3.4.1 Perhitungan Manual Proses AHP

Proses perhitungan AHP dimulai dengan menentukan matriks perbandingan berpasangan. Dalam penentuan bobot kriteria dilakukan dengan menerapkan tolak ukur Saaty. Matriks perbandingan berpasangan dapat dilihat pada Tabel 3.2.

Tabel 3.2 Data Matriks Perbandingan

	PK	UR	SR	LR	DR	AR	PG	TG	AP	AE
PK	1	3.00	3.00	3.00	3.00	3.00	4.00	4.00	3.00	3.00
UR	0.33	1	2.00	2.00	2.00	2.00	3.00	3.00	4.00	4.00
SR	0.33	0.50	1	2.00	2.00	2.00	3.00	3.00	4.00	4.00
LR	0.33	0.50	0.50	1	2.00	2.00	3.00	3.00	4.00	4.00
DR	0.33	0.50	0.50	0.50	1	2.00	3.00	3.00	2.00	2.00
AR	0.33	0.50	0.50	0.50	0.50	1	3.00	3.00	2.00	2.00
PG	0.25	0.33	0.33	0.33	0.33	0.33	1	4.00	2.00	2.00
TG	0.25	0.33	0.33	0.33	0.33	0.33	0.25	1	2.00	2.00
AP	0.33	0.25	0.25	0.25	0.50	0.50	0.50	0.50	1	2.00
AE	0.33	0.25	0.25	0.25	0.50	0.50	0.50	0.50	0.50	1
TOTAL	3.83	7.17	8.67	10.17	12.17	13.67	21.25	25.00	24.50	26.00

Keterangan Tabel 3.2 adalah sebagai berikut.

- PK : Pekerjaan
 UR : Ukuran Rumah
 SR : Status Rumah
 LR : Lantai Rumah
 DR : Dinding Rumah
 AR : Atap Rumah
 PG : Penghasilan
 TG : Tanggungan Anak
 AP : Aset Kendaraan Pribadi
 AE : Aset Elektronik

Tahap selanjutnya adalah proses normalisasi matriks yaitu dengan membagi tiap elemen dari matriks perbandingan dengan total elemen kriteria. Hasil normalisasi matriks perbandingan dapat dilihat pada Tabel 3.3.

Tabel 3.3 Data Normalisasi Matriks Perbandingan

	PK	UR	SR	LR	DR	AR	PG	TG	AP	AE
PK	0.261	0.419	0.346	0.295	0.247	0.220	0.188	0.160	0.122	0.115
UR	0.087	0.140	0.231	0.197	0.164	0.146	0.141	0.120	0.163	0.154
SR	0.087	0.070	0.115	0.197	0.164	0.146	0.141	0.120	0.163	0.154
LR	0.087	0.070	0.058	0.098	0.164	0.146	0.141	0.120	0.163	0.154
DR	0.087	0.070	0.058	0.049	0.082	0.146	0.141	0.120	0.082	0.077
AR	0.087	0.070	0.058	0.049	0.041	0.073	0.141	0.120	0.082	0.077
PG	0.065	0.047	0.038	0.033	0.027	0.024	0.047	0.160	0.082	0.077
TG	0.065	0.047	0.038	0.033	0.027	0.024	0.012	0.040	0.082	0.077
AP	0.087	0.035	0.029	0.025	0.041	0.037	0.024	0.020	0.041	0.077
AE	0.087	0.035	0.029	0.025	0.041	0.037	0.024	0.020	0.020	0.038
TOTAL	1	1	1	1	1	1	1	1	1	1

Setelah diperoleh data hasil normalisasi matriks perbandingan berpasangan, maka tahap selanjutnya adalah dengan menghitung bobot kriteria dengan menjumlahkan tiap baris elemen kriteria, kemudian hasil penjumlahan dibagi dengan banyaknya elemen prioritas yang bersangkutan. Dari hasil perhitungan

tersebut diperoleh bobot dari tiap elemen kriteria. Hasil perhitungan bobot dapat dilihat pada Tabel 3.4.

Tabel 3.4 Data Bobot Kriteria

Kriteria	Bobot
Pekerjaan	0.237
Ukuran Rumah	0.154
Status Rumah	0.136
Lantai Rumah	0.120
Dinding Rumah	0.091
Atap Rumah	0.080
Penghasilan	0.060
Tanggungan Anak	0.045
Aset Kendaraan Pribadi	0.041
Aset Elektronik	0.036

Tahap selanjutnya adalah mendapatkan nilai λ_{maks} dengan menjumlahkan hasil bagi dari perkalian antara matriks perbandingan dengan bobot kriteria. Berikut adalah salah satu contoh perhitungan untuk kriteria Pekerjaan. Hasil perhitungan dapat dilihat pada Tabel 3.5.

$$Ax = (1 \times 0.237) + (3 \times 0.154) + (3 \times 0.136) + (3 \times 0.120) + (3 \times 0.091) + (3 \times 0.080) + (4 \times 0.060) + (4 \times 0.045) + (3 \times 0.041) + (3 \times 0.036) = 2.630$$

Tabel 3.5 Hasil Perhitungan Nilai Ax

Kriteria	Bobot	Ax
Pekerjaan	0.237	2.630
Ukuran Rumah	0.154	1.709
Status Rumah	0.136	1.496
Lantai Rumah	0.120	1.308
Dinding Rumah	0.091	1.002
Atap Rumah	0.080	0.887
Penghasilan	0.060	0.645
Tanggungan Anak	0.045	0.466
Aset Kendaraan Pribadi	0.041	0.432
Aset Elektronik	0.036	0.376

Untuk menghitung λ_{maks} dapat dengan menggunakan Persamaan 2.2. Sehingga diperoleh nilai $\lambda_{maks} = 10.827$. Proses selanjutnya adalah menghitung nilai *Consistency Index* (CI) menggunakan Persamaan 2.3.

$$CI = \frac{(10.827 - 10)}{9} = 0.092$$

Setelah diperoleh nilai CI, langkah selanjutnya adalah menghitung rasio konsistensi atau *Consistency Ratio* (CR) menggunakan Persamaan 2.4.

$$CR = \frac{0.092}{1.49} = 0.062$$

Dari hasil perhitungan CR tersebut diketahui nilainya kurang atau sama dengan 0,1 maka perhitungan tersebut benar dan konsisten.

Setelah perhitungan level 1 selesai dan mendapatkan bobot dari masing-masing kriteria, langkah selanjutnya adalah menghitung bobot dari tiap sub kriteria. Untuk proses perhitungannya sama dengan tahapan perhitungan AHP yang sudah dibahas sebelumnya. Berikut adalah perhitungan tiap sub kriteria dan hasil yang diperoleh bahwa perhitungan dari semua sub kriteria menunjukkan hasil yang konsisten.

a. Kriteria Pekerjaan

Sub Kriteria Pekerjaan terdiri dari sub kriteria Tidak Bekerja, Buruh, Karyawan, Wiraswasta, dan Pengusaha. Matriks perbandingan dari kriteria Pekerjaan dapat dilihat pada Tabel 3.6.

Tabel 3.6 Matrik Perbandingan Berpasangan Kriteria Pekerjaan

	TB	BR	KW	WS	PH
TB	1	2.00	2.00	3.00	3.00
BR	0.50	1	2.00	3.00	3.00
KW	0.50	0.50	1	3.00	3.00
WS	0.33	0.33	0.33	1	4.00
PH	0.33	0.33	0.33	0.25	1
TOTAL	2.67	4.17	5.67	10.25	14.00

Bobot dari sub kriteria yang diperoleh dari proses AHP, kemudian dikalikan dengan bobot kriteria. Sehingga diperoleh bobot akhir seperti pada Tabel 3.7.

Tabel 3.7 Bobot Sub Kriteria dari Kriteria Pekerjaan

Bobot Kriteria PK =		0.237
Tidak Bekerja	0.343	0.081
Buruh	0.257	0.061
Karyawan	0.198	0.047
Wiraswasta	0.129	0.031
Pengusaha	0.072	0.017

b. Kriteria Ukuran Rumah

Sub Kriteria Ukuran Rumah terdiri dari sub kriteria Sangat Kecil, Kecil, Sedang, dan Besar. Matriks perbandingan dari kriteria Ukuran Rumah, dapat dilihat pada Tabel 3.8.

Tabel 3.8 Matrik Perbandingan Berpasangan Kriteria Ukuran Rumah

	SK	K	S	B
SK	1	2.00	2.00	3.00
K	0.50	1	2.00	3.00
S	0.50	0.50	1	4.00
B	0.33	0.33	0.25	1
TOTAL	2.33	3.83	5.25	11.00

Bobot dari sub kriteria yang diperoleh dari proses AHP, kemudian dikalikan dengan bobot kriteria. Sehingga diperoleh bobot akhir seperti pada Tabel 3.9.

Tabel 3.9 Bobot Sub Kriteria dari Kriteria Ukuran Rumah

Bobot Kriteria UR =		0.154
SK	0.401	0.062
K	0.282	0.044
S	0.225	0.035
B	0.092	0.014

c. Kriteria Status Rumah

Sub Kriteria Status Rumah terdiri dari sub kriteria Kontrak, Milik Orang Tua, dan Milik Sendiri. Matriks perbandingan dari kriteria Status Rumah dapat dilihat pada Tabel 3.10.

Tabel 3.10 Matrik Perbandingan Berpasangan Kriteria Status Rumah

	K	MO	MS
K	1	3.00	5.00
MO	0.33	1	3.00
MS	0.20	0.33	1
TOTAL	1.53	4.33	9.00

Bobot dari sub kriteria yang diperoleh dari proses AHP, kemudian dikalikan dengan bobot kriteria. Sehingga diperoleh bobot akhir seperti pada Tabel 3.11.

Tabel 3.11 Bobot Sub Kriteria dari Kriteria Status Rumah

Bobot Kriteria SR =		0.136
K	0.633	0.086
MO	0.260	0.035
MS	0.106	0.014

d. Kriteria Lantai Rumah

Sub Kriteria Lantai Rumah terdiri dari sub kriteria Tanah, Tegel, dan Keramik. Matriks perbandingan dari kriteria Lantai Rumah dapat dilihat pada Tabel 3.12.

Tabel 3.12 Matrik Perbandingan Berpasangan Kriteria Lantai Rumah

	T	TG	K
T	1	2.00	3.00
TG	0.50	1	3.00
K	0.33	0.33	1
TOTAL	1.83	3.33	7.00

Bobot dari sub kriteria yang diperoleh dari proses AHP, kemudian dikalikan dengan bobot kriteria. Sehingga diperoleh bobot akhir seperti pada Tabel 3.13.

Tabel 3.13 Bobot Sub Kriteria dari Kriteria Lantai Rumah

Bobot Kriteria LR =		0.120
T	0.525	0.063
TG	0.334	0.040
K	0.142	0.017

e. Kriteria Dinding Rumah

Sub Kriteria Dinding Rumah terdiri dari Bambu, Kayu, Tembok. Matriks perbandingan dari kriteria Dinding Rumah dapat dilihat pada Tabel 3.14.

Tabel 3.14 Matrik Perbandingan Berpasangan Dinding Rumah

	BM	KY	TB
BM	1	4.00	5.00
KY	0.25	1	3.00
TB	0.20	0.33	1
TOTAL	1.45	5.33	9.00

Bobot dari sub kriteria yang diperoleh dari proses AHP, kemudian dikalikan dengan bobot kriteria. Sehingga diperoleh bobot akhir seperti pada Tabel 3.15.

Tabel 3.15 Bobot Sub Kriteria dari Kriteria Dinding Rumah

Bobot Kriteria DR =		0.091
BM	0.665	0.061
KY	0.231	0.021
TB	0.104	0.009

f. Kriteria Atap Rumah

Sub Kriteria Atap Rumah terdiri dari Seng, Asbes, dan Genteng. Matriks perbandingan dari kriteria Atap Rumah dapat dilihat pada Tabel 3.16.

Tabel 3.16 Matrik Perbandingan Berpasangan Atap Rumah

	SG	AB	GT
SG	1	2.00	5.00
AB	0.50	1	3.00
GT	0.20	0.33	1
TOTAL	1.70	3.33	9.00

Bobot dari sub kriteria yang diperoleh dari proses AHP, kemudian dikalikan dengan bobot kriteria. Sehingga diperoleh bobot akhir seperti pada Tabel 3.17.

Tabel 3.17 Bobot Sub Kriteria dari Kriteria Atap Rumah

Bobot Kriteria AR =		0.080
SG	0.581	0.046
AB	0.309	0.025
GT	0.110	0.009

g. Kriteria Penghasilan

Sub Kriteria Penghasilan terdiri dari 0, 50-100, 100-300, 300-500, >500.

Matriks perbandingan dari kriteria penghasilan dapat dilihat pada Tabel 3.18.

Tabel 3.18 Matrik Perbandingan Berpasangan Penghasilan

	0	50-100	100-300	300-500	>500
0	1	2.00	2.00	3.00	3.00
50-100	0.50	1	2.00	3.00	3.00
100-300	0.50	0.50	1	3.00	3.00
300-500	0.33	0.33	0.33	1	3.00
> 500	0.33	0.33	0.33	0.33	1
TOTAL	2.67	4.17	5.67	10.33	13.00

Bobot dari sub kriteria yang diperoleh dari proses AHP, kemudian dikalikan dengan bobot kriteria. Sehingga diperoleh bobot akhir seperti pada Tabel 3.19.

Tabel 3.19 Bobot Sub Kriteria dari Kriteria Penghasilan

Bobot Kriteria PG =		0.060
0	0.346	0.021
50-100	0.260	0.016
100-300	0.201	0.012
300-500	0.118	0.007
> 500	0.075	0.004

h. Kriteria Jumlah Tanggungan

Sub Kriteria Jumlah Tanggungan terdiri dari > 5 orang, 3-5 orang, 0-3 orang. Matriks perbandingan dari kriteria Jumlah Tanggungan dapat dilihat pada Tabel 3.20.

Tabel 3.20 Matrik Perbandingan Berpasangan Tanggungan

	>5	3-5	0-3
>5	1	2.00	3.00
3-5	0.50	1	3.00
0-3	0.33	0.33	1
TOTAL	1.83	3.33	7.00

Bobot dari sub kriteria yang diperoleh dari proses AHP, kemudian dikalikan dengan bobot kriteria. Sehingga diperoleh bobot akhir seperti pada Tabel 3.21.

Tabel 3.21 Bobot Sub Kriteria dari Kriteria Tanggungan

Bobot Kriteria TG =		0.045
>5	0.525	0.023
3-5	0.334	0.015
0-3	0.142	0.006

i. Kriteria Aset Kendaraan Pribadi

Sub Kriteria Aset Kendaraan Pribadi terdiri dari 0, < 1 Juta, 1-4 Juta, 4 – 8 Juta, >5 Juta. Matriks perbandingan dari kriteria Aset Kendaraan Pribadi dapat dilihat pada Tabel 3.22.

Tabel 3.22 Matrik Perbandingan Berpasangan Aset Kendaraan Pribadi

	0	< 1	1-4	4-8	>5
0	1	2.00	3.00	3.00	3.00
< 1	0.50	1	3.00	3.00	3.00
1-4	0.33	0.33	1	2.00	3.00
4-8	0.33	0.33	0.50	1	4.00
>5	0.33	0.33	0.33	0.25	1
TOTAL	2.50	4.00	7.83	9.25	14.00

Bobot dari sub kriteria yang diperoleh dari proses AHP, kemudian dikalikan dengan bobot kriteria. Sehingga diperoleh bobot akhir seperti pada Tabel 3.23.

Tabel 3.23 Bobot Sub Kriteria dari Kriteria Aset Kendaraan Pribadi

Bobot Kriteria AP =		0.041
0	0.364	0.015
< 1	0.274	0.011
1-4	0.155	0.006
4-8	0.135	0.006
>5	0.072	0.003

j. Kriteria Aset Elektronik

Sub Kriteria Aset Elektronik terdiri dari: 0, < 1 Juta, 1-4 Juta, 4 – 8 Juta, >5 Juta. Matriks perbandingan dari kriteria Aset Elektronik dapat dilihat pada Tabel 3.24.

Tabel 3.24 Matrik Perbandingan Berpasangan Aset Elektronik

	0	< 1	1-4	4-8	>5
0	1	2.00	3.00	3.00	2.00
< 1	0.50	1	2.00	3.00	2.00
1-4	0.33	0.50	1	2.00	2.00
4-8	0.33	0.33	0.50	1	3.00
>5	0.50	0.50	0.50	0.33	1
TOTAL	2.67	4.33	7.00	9.33	10.00

Bobot dari sub kriteria yang diperoleh dari proses AHP, kemudian dikalikan dengan bobot kriteria. Sehingga diperoleh bobot akhir seperti pada Tabel 3.25.

Tabel 3.25 Bobot Sub Kriteria dari Kriteria Aset Elektronik

Bobot Kriteria AP =		0.036
0	0.357	0.013
< 1	0.245	0.009
1-4	0.160	0.006
4-8	0.136	0.005
>5	0.102	0.004

Setelah proses perhitungan bobot untuk kriteria dan sub kriteria dilakukan, proses selanjutnya adalah reduksi data. Reduksi data dilakukan untuk memperoleh kriteria tertentu berdasarkan nilai batas yang ditetapkan. Dalam proses ini *threshold* atau nilai batas yang ditetapkan adalah lebih dari 0.015. Hasil nilai bobot AHP dan hasil *sorting* data dapat dilihat pada Tabel 3.26 dan Tabel 3.27. Tabel 3.26 adalah tabel untuk keseluruhan bobot sub kriteria yang belum dilakukan proses *sorting*. Pada Tabel 3.27 ditunjukkan hasil *sorting* nilai bobot sub kriteria yang terpilih diatas nilai *threshold*.

Tabel 3.26 Nilai Bobot seluruh kriteria dan Sub Kriteria

Kriteria	K1	K2	K3	K4	K5	K6	K7	K8	K9	K10
Bobot	0.081	0.061	0.047	0.031	0.017	0.062	0.044	0.035	0.014	0.086
Kriteria	K11	K12	K13	K14	K15	K16	K17	K18	K19	K20
Bobot	0.035	0.014	0.063	0.040	0.017	0.061	0.021	0.009	0.046	0.025
Kriteria	K21	K22	K23	K24	K25	K26	K27	K28	K29	K30
Bobot	0.009	0.021	0.016	0.012	0.007	0.004	0.023	0.015	0.006	0.015
Kriteria	K31	K32	K33	K34	K35	K36	K37	K38	K39	K40
Bobot	0.011	0.006	0.006	0.003	0.013	0.009	0.006	0.005	0.004	-

Tabel 3.27 Kriteria setelah proses Shorting

Kriteria	K10	K1	K13	K6	K2	K16	K3	K19	K7	K14
Bobot	0.086	0.081	0.063	0.062	0.061	0.061	0.047	0.046	0.044	0.040
Kriteria	K11	K8	K4	K20	K27	K17	K22	K5	K15	K23
Bobot	0.035	0.035	0.031	0.025	0.023	0.021	0.021	0.017	0.017	0.016
Kriteria	K30	K28	K12	K9	K35	K24	K31	K18	K21	K36
Bobot	0.015	0.015	0.014	0.014	0.013	0.012	0.011	0.009	0.009	0.009
Kriteria	K25	K32	K29	K37	K33	K38	K26	K39	K34	K40
Bobot	0.007	0.006	0.006	0.006	0.006	0.005	0.004	0.004	0.003	-

3.4.2 Perhitungan Manual Proses Normalisasi

Setelah proses perhitungan untuk mendapatkan bobot dengan AHP selesai, proses selanjutnya adalah proses normalisasi. Normalisasi digunakan untuk mendapatkan nilai data dengan ukuran yang lebih kecil sehingga dapat mewakili data asli yaitu dengan interval $[0.1, 0.9]$. Data yang akan dinormalisasi adalah *dataset* penerima RASKIN sebanyak 20 data seperti pada Lampiran 4 Tabel Data

sample. *Dataset* diperoleh dari perbandingan Rasio 40 : 60 dari jumlah *dataset* yang digunakan sebanyak 20 data. Proses perhitungan normalisasi dimulai dengan mencari nilai *maximum* dan *minimum* dari seluruh atribut. Berikut adalah perhitungan normalisasi dengan menggunakan Persamaan 2.1 pada atribut pertama.

Nilai $x = 0$ (adalah nilai *record* pertama pada atribut pertama). $a = 0$ (adalah nilai data minimum dari seluruh atribut) dan $b = 5$ (adalah nilai data maximum dari seluruh atribut).

$$x' = \frac{0.8(0-0)}{5-0} + 0.1 = \frac{0.8 \times 0}{4} + 0.1 = 0.1$$

Proses dilakukan untuk seluruh atribut pada *dataset* penerima RASKIN. Hasil proses normalisasi untuk data *training* dan *testing* dapat dilihat pada Tabel 3.28 dan Tabel 3.29. Data hasil proses normalisasi dapat dilihat pada Lampiran 5 Tabel Hasil Normalisasi Data *Training* dan Lampiran 6 Hasil Normalisasi Data *Testing*.

Tabel 3.28 Hasil Normalisasi Data *Training*

	K1	K2	K3	K4	K5	K6	K7	...	K23	K27	K28	K30	Class
KL_1	0.1	0.7	0.1	0.1	0.1	0.1	0.6	...	0.7	0.6	0.1	0.9	-1
KL_2	0.9	0.1	0.1	0.1	0.1	0.1	0.1	...	0.1	0.1	0.1	0.9	-1
KL_3	0.1	0.1	0.1	0.4	0.1	0.1	0.6	...	0.1	0.1	0.1	0.1	-1
KL_4	0.1	0.7	0.1	0.1	0.1	0.7	0.1	...	0.7	0.1	0.4	0.1	-1
KL_5	0.1	0.1	0.1	0.4	0.1	0.7	0.1	...	0.7	0.1	0.1	0.9	-1
...	...	...	...	...	...	...	...	...	...	...	...	...	...
KL_10	0.1	0.1	0.1	0.4	0.1	0.1	0.1	...	0.1	0.1	0.4	0.1	1
KL_11	0.1	0.1	0.1	0.1	0.3	0.1	0.1	...	0.1	0.1	0.1	0.1	1
KL_12	0.1	0.1	0.1	0.4	0.1	0.1	0.1	...	0.1	0.1	0.4	0.1	1

Tabel 3.29 Hasil Normalisasi Data *Testing*

	K1	K2	K3	K4	K5	K6	K7	...	K23	K27	K28	K30	Class
KL_8	0.1	0.7	0.1	0.1	0.1	0.7	0.1	...	0.7	0.6	0.1	0.9	-1
KL_16	0.1	0.7	0.1	0.1	0.1	0.1	0.1	...	0.1	0.1	0.1	0.9	-1
KL_17	0.1	0.1	0.1	0.4	0.1	0.1	0.6	...	0.1	0.1	0.1	0.1	-1
KL_13	0.1	0.1	0.1	0.4	0.1	0.1	0.1	...	0.7	0.1	0.4	0.1	1
KL_14	0.1	0.1	0.1	0.1	0.3	0.1	0.1	...	0.7	0.1	0.1	0.9	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...
KL_18	0.1	0.7	0.1	0.1	0.1	0.7	0.1	...	0.7	0.1	0.1	0.9	-1
KL_19	0.1	0.1	0.1	0.1	0.3	0.1	0.1	...	0.1	0.1	0.1	0.1	1
KL_20	0.1	0.1	0.1	0.4	0.1	0.1	0.1	...	0.1	0.1	0.4	0.1	1

3.4.3 Perhitungan Manual Proses *Training* SVM

Perhitungan manual proses *training* SVM, pada bagian ini akan menggunakan metode *sequential*. Data dari sekumpulan objek akan diklasifikasikan dalam proses *training*. Pada *Support Vectore Machine* (SVM), diketahui *hyperplane* yang merupakan fungsi pemisah dengan nilai margin yang maksimal. *Hyperplane* adalah suatu bidang pemisah yang terletak ditengah antara dua set objek dari dua kelas. Untuk mendapatkan nilai margin yang maksimal dapat dirumuskan menggunakan metode *sequential*.

Parameter yang terdapat pada fungsi SVM dan digunakan dalam proses *training* antara lain adalah w (*weight*), b (posisi bidang relatif terhadap pusat koordinat) dan C (*Complexity*). Setelah data dinormalisasi dengan menggunakan Persamaan 2.1, kemudian dicari bobot untuk data yang telah dinormalisasi. Untuk mencari hasil bobot hasil normalisasi, data pada *record* dan atribut pertama dikalikan dengan tersebut dikalikan hasil akhir bobot masing-masing sub kriteria yang diperoleh dari perhitungan AHP sebelumnya.

Berikut adalah salah satu contoh perhitungan untuk mendapatkan nilai data normalisasi yang telah dikalikan dengan bobot sub kriteria. *Record* pertama pada data ternormalisasi = 0.1. Bobot sub kriteria atribut pertama = 0.081. Maka, hasil akhir = $0.1 \times 0.081 = 0.0081$.

Hasil perhitungan bobot dengan data yang sudah dinormalisasi dapat dilihat pada Tabel 3.30 Hasil perkalian data normalisasi dan bobot kriteria. Hasil perhitungan secara lebih jelas dapat dilihat pada Lampiran 7 Tabel Hasil Pembobotan.

Data hasil pembobotan digunakan untuk menghitung nilai *kernel*. *Kernel* yang akan digunakan adalah *polynomial* dan *kernel Gaussian RBF* (*Radial Basic Function*). Hasil perhitungan *Kernel polynomial*, dapat dilihat pada Tabel 3.31 Hasil perhitungan data *training* menggunakan *kernel polynomial*. Hasil perhitungan secara lebih jelas dapat dilihat pada Lampiran 8 Tabel Hasil Perhitungan Data *Training* dengan *Kernel Polynomial*.

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

Tabel 3.30 Hasil Perkalian antara data ternormalisasi dan bobot sub kriteria

	K1	K2	K3	K4	K5	K6	...	K27	K28	K30	Class
KEL1	0.008	0.045	0.005	0.003	0.002	0.006	...	0.013	0.002	0.014	-1
KEL2	0.073	0.006	0.005	0.003	0.002	0.006	...	0.002	0.002	0.014	-1
KEL3	0.008	0.006	0.005	0.013	0.002	0.006	...	0.002	0.002	0.002	-1
KEL4	0.008	0.045	0.005	0.003	0.002	0.046	...	0.002	0.006	0.002	-1
KEL5	0.008	0.006	0.005	0.013	0.002	0.046	...	0.002	0.002	0.014	-1
KEL6	0.008	0.045	0.005	0.003	0.002	0.046	...	0.002	0.002	0.002	-1
KEL7	0.008	0.045	0.005	0.003	0.002	0.046	...	0.002	0.006	0.002	-1
KEL8	0.008	0.045	0.005	0.003	0.002	0.046	...	0.002	0.006	0.002	-1
KEL9	0.008	0.006	0.005	0.003	0.004	0.006	...	0.002	0.006	0.002	1
KEL10	0.008	0.006	0.005	0.013	0.002	0.006	...	0.002	0.002	0.002	1
KEL11	0.008	0.006	0.005	0.003	0.004	0.006	...	0.002	0.006	0.002	1
KEL12	0.008	0.006	0.005	0.013	0.002	0.006	...	0.002	0.002	0.002	1
KEL13	0.008	0.006	0.005	0.013	0.002	0.006	...	0.002	0.002	0.002	1
KEL14	0.008	0.006	0.005	0.003	0.004	0.006	...	0.002	0.002	0.014	1
KEL15	0.008	0.006	0.005	0.013	0.002	0.006	...	0.002	0.002	0.002	1
KEL16	0.008	0.045	0.005	0.003	0.002	0.006	...	0.002	0.002	0.002	-1
KEL17	0.008	0.006	0.005	0.013	0.002	0.006	...	0.002	0.002	0.002	-1
KEL18	0.008	0.045	0.005	0.003	0.002	0.046	...	0.002	0.002	0.014	-1
KEL19	0.008	0.006	0.005	0.003	0.004	0.006	...	0.002	0.002	0.002	1
KEL20	0.008	0.006	0.005	0.013	0.002	0.006	...	0.002	0.006	0.002	1

Tabel 3.31 Hasil Perhitungan Data *Training* dengan *Kernel Polynomial*

Data	1	2	3	4	...	10	11
KL_1	0.000024	0.000008	0.000003	0.000018	...	0.000001	0.000001
KL_2	0.000008	0.000061	0.000002	0.000009	...	0.000002	0.000002
KL_3	0.000003	0.000002	0.000002	0.000002	...	0.000000	0.000001
KL_4	0.000018	0.000009	0.000002	0.000037	...	0.000001	0.000001
KL_5	0.000009	0.000011	0.000002	0.000022	...	0.000002	0.000002
KL_6	0.000018	0.000009	0.000002	0.000035	...	0.000001	0.000001
KL_7	0.000019	0.000009	0.000002	0.000037	...	0.000001	0.000002
KL_9	0.000001	0.000002	0.000000	0.000001	...	0.000000	0.000000
KL_10	0.000001	0.000002	0.000001	0.000001	...	0.000000	0.000000
KL_11	0.000001	0.000002	0.000000	0.000001	...	0.000000	0.000000
KL_12	0.000001	0.000002	0.000001	0.000001	...	0.000000	0.000000

Setelah dilakukan perhitungan *kernel*, proses selanjutnya adalah perhitungan dengan metode *sequential training* untuk mendapatkan parameter SVM lainnya. Proses perhitungan menggunakan metode *sequential* dapat dilihat secara bertahap pada Persamaan 2.21 sampai Persamaan 2.24. Berikut adalah tahapan dalam perhitungan *sequential training* SVM.

1. Melakukan inisialisasi dengan nilai $\alpha_i = 0$
2. Menghitung komponen matriks pada Persamaan 2.21.

$D_{ij} = y_i y_j (K(x_i, x_j) + \lambda^2)$, dimana $y_i = 1$ Class pada atribut pertama dan $y_j = 1$ Class pada atribut kedua. (x_i, x_j) adalah *kernel* yang telah dihitung pada proses sebelumnya. Inisialisasi nilai lamda $\lambda = 0.5$.

Tabel 3.32 adalah hasil perhitungan matriks hessian dengan mengimplementasikan Persamaan 2.21.

Tabel 3.32 Hasil Perhitungan Matriks Hessian

Data	1	2	3	4	5	6	7	8	9	10	11
1	0.250	0.250	0.250	0.250	0.250	0.250	0.250	-0.250	-0.250	-0.250	-0.250
2	0.250	0.250	0.250	0.250	0.250	0.250	0.250	-0.250	-0.250	-0.250	-0.250
3	0.250	0.250	0.250	0.250	0.250	0.250	0.250	-0.250	-0.250	-0.250	-0.250
4	0.250	0.250	0.250	0.250	0.250	0.250	0.250	-0.250	-0.250	-0.250	-0.250
5	0.250	0.250	0.250	0.250	0.250	0.250	0.250	-0.250	-0.250	-0.250	-0.250
6	0.250	0.250	0.250	0.250	0.250	0.250	0.250	-0.250	-0.250	-0.250	-0.250
7	0.250	0.250	0.250	0.250	0.250	0.250	0.250	-0.250	-0.250	-0.250	-0.250
8	-0.250	-0.250	-0.250	-0.250	-0.250	-0.250	-0.250	0.250	0.250	0.250	0.250
9	-0.250	-0.250	-0.250	-0.250	-0.250	-0.250	-0.250	0.250	0.250	0.250	0.250
10	-0.250	-0.250	-0.250	-0.250	-0.250	-0.250	-0.250	0.250	0.250	0.250	0.250
11	-0.250	-0.250	-0.250	-0.250	-0.250	-0.250	-0.250	0.250	0.250	0.250	0.250

Setelah diperoleh nilai matriks langkah selanjutnya menggunakan Persamaan 2.22 sampai 2.24.

3. Melakukan iterasi untuk setiap $i, = 1, 2, \dots, n$. Pada perhitungan ini dilakukan iterasi (*itermax*) = 10, $C = 1$, dan γ adalah nilai konstanta, dalam penelitian ini nilai $\gamma = 0.01$. Untuk mendapatkan nilai γ menggunakan rumus konstanta

dibagi dengan nilai max diagonal Matrik Hessian. Sehingga diperoleh nilai

$$\gamma = \frac{0.01}{0.25} = 0.04$$

Iterasi ke 0:

- a) Menghitung Persamaan 2.22. $E_i = \sum_{j=1}^n \alpha_i D_{ij}$. Contoh perhitungan *record*

pertama adalah sebagai berikut.

$$E_1 = \begin{pmatrix} 0 \times 0.25 + 0 \times 0.25 + 0 \times 0.25 + 0 \times 0.25 + \\ 0 \times 0.25 + 0 \times 0.25 + 0 \times 0.25 + 0 \times -0.25 + \\ 0 \times -0.25 + 0 \times -0.25 + 0 \times -0.25 \end{pmatrix} = 0$$

Perhitungan berlaku untuk seluruh data, dan hasil perhitungan dapat dilihat pada Tabel 3.33.

Tabel 3.33 Hasil Perhitungan Iterasi ke 0 pada Persamaan 2.22

$\alpha_i D_{ij}$	1	2	3	4	5	6	7	8	9	10	11	E_i
1	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0

- b) Menghitung Nilai $\delta\alpha_i$ dengan Persamaan 2.23. $\delta\alpha_i = \min \{ \max [\gamma (1 - E_i), -\alpha_i], C - \alpha_i \}$. Contoh perhitungan *record* pertama adalah $\delta\alpha_i = \min \{ \max [0.04 (1 - 0), -0], 1 - 0 \}$. Perhitungan berlaku untuk seluruh data, dan hasil perhitungan dapat dilihat pada Tabel 3.34.

Tabel 3.34 Hasil Perhitungan Iterasi ke 0 pada Persamaan 2.23

	1	2	3	4	5	6	7	8	9	10	11
$\delta\alpha_i$	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04

- c) Memperbarui Nilai α_i dengan Persamaan 2.24. $\alpha_i = \alpha_i + \delta\alpha_i$ Contoh perhitungan record pertama adalah $\alpha_i = 0 + 0.040 = 0.040$. Perhitungan berlaku untuk seluruh data, dan hasil perhitungan dapat dilihat pada Tabel 3.35.

Tabel 3.35 Hasil Perhitungan Iterasi ke 0 pada Persamaan 2.24

	1	2	3	4	5	6	7	8	9	10	11
α_i	0.040	0.040	0.040	0.040	0.040	0.040	0.040	0.040	0.040	0.040	0.040

- d) Proses selanjutnya adalah melakukan pengecekan iterasi, apabila iterasi telah mencapai nilai konvergen ($|\delta\alpha| < \epsilon$), maka iterasi akan dihentikan. Nilai *threshold* (ϵ) yang diperoleh pada hasil perhitungan iterasi ke-0 adalah 0.001. Nilai max dapat dilihat dari nilai keseluruhan data yang terdapat pada Tabel 3.34. Karena nilai max yang diperoleh adalah 0.040 dan kurang dari nilai *threshold* yang ditentukan yaitu 0.001 maka iterasi dapat dilanjutkan. Selain itu iterasi akan dihentikan ketika mencapai maximum iterasi yang ditentukan. Dari hasil perhitungan pada iterasi ke-0 diperoleh nilai α_i yang baru adalah 0.040. Nilai α_i tersebut akan digunakan sebagai perhitungan pada proses iterasi selanjutnya.

Iterasi ke 3:

- a) Menghitung Persamaan 2.22. $E_i = \sum_{j=1}^n \alpha_i D_{ij}$ Contoh perhitungan *record* pertama adalah sebagai berikut.

$$E_3 = \begin{pmatrix} 0.1165 \times 0.25 + 0.1165 \times 0.25 + 0.1165 \times 0.25 + 0.1165 \times 0.25 + \\ 0.1165 \times 0.25 + 0.1165 \times 0.25 + 0.1165 \times 0.25 + 0.1234 \times -0.25 + \\ 0.1234 \times -0.25 + 0.1234 \times -0.25 + 0.1234 \times -0.25 \end{pmatrix} = 0.03$$

Perhitungan berlaku untuk seluruh data, dan hasil perhitungan dapat dilihat pada Tabel 3.36.

Tabel 3.36 Hasil Perhitungan Iterasi ke 3 pada Persamaan 2.22

$\alpha_i D_{ij}$	1	2	3	4	5	6	7	8	9	10	11	E_i
1	0.03	0.03	0.03	0.03	0.03	0.03	0.03	-0.03	-0.03	-0.03	-0.03	0.08
2	0.03	0.03	0.03	0.03	0.03	0.03	0.03	-0.03	-0.03	-0.03	-0.03	0.08
3	0.03	0.03	0.03	0.03	0.03	0.03	0.03	-0.03	-0.03	-0.03	-0.03	0.08
4	0.03	0.03	0.03	0.03	0.03	0.03	0.03	-0.03	-0.03	-0.03	-0.03	0.08
5	0.03	0.03	0.03	0.03	0.03	0.03	0.03	-0.03	-0.03	-0.03	-0.03	0.08
6	0.03	0.03	0.03	0.03	0.03	0.03	0.03	-0.03	-0.03	-0.03	-0.03	0.08
7	0.03	0.03	0.03	0.03	0.03	0.03	0.03	-0.03	-0.03	-0.03	-0.03	0.08
8	-0.03	-0.03	-0.03	-0.03	-0.03	-0.03	-0.03	0.03	0.03	0.03	0.03	-0.08
9	-0.03	-0.03	-0.03	-0.03	-0.03	-0.03	-0.03	0.03	0.03	0.03	0.03	-0.08
10	-0.03	-0.03	-0.03	-0.03	-0.03	-0.03	-0.03	0.03	0.03	0.03	0.03	-0.08
11	-0.03	-0.03	-0.03	-0.03	-0.03	-0.03	-0.03	0.03	0.03	0.03	0.03	-0.08

- b) Menghitung Nilai $\delta\alpha_i$ dengan Persamaan 2.23. $\delta\alpha_i = \min \{ \max [\gamma (1 - E_i), -\alpha_i], C - \alpha_i \}$. Contoh perhitungan record pertama adalah $\delta\alpha_i = \min \{ \max [0.04 (1 - 0.08), -0.1165], 1 - 0.1165 \}$. Perhitungan berlaku untuk seluruh data, dan hasil perhitungan dapat dilihat pada Tabel 3.37.

Tabel 3.37 Hasil Perhitungan Iterasi ke 3 pada Persamaan 2.23

	1	2	3	4	5	6	7	8	9	10	11
$\delta\alpha_i$	0.036	0.036	0.036	0.036	0.036	0.036	0.036	0.043	0.043	0.043	0.043

- c) Memperbarui Nilai α_i dengan Persamaan 2.24. $\alpha_i = \alpha_i + \delta\alpha_i$. Contoh perhitungan record pertama adalah $\alpha_i = 0.1165 + 0.0368 = 0.1533$. Perhitungan berlaku untuk seluruh data, dan hasil perhitungan dapat dilihat pada Tabel 3.38.

Tabel 3.38 Hasil Perhitungan Iterasi ke 3 pada Persamaan 2.24

	1	2	3	4	5	6	7	8	9	10	11
α_i	0.153	0.153	0.153	0.153	0.153	0.153	0.153	0.166	0.166	0.166	0.166

- d) Proses selanjutnya adalah melakukan pengecekan iterasi, apabila iterasi telah mencapai nilai konvergen ($|\delta\alpha| < \epsilon$), maka iterasi akan dihentikan. Nilai *threshold* (ϵ) yang diperoleh pada hasil perhitungan iterasi ke-3 adalah 0.001. Nilai max dapat dilihat dari nilai keseluruhan data pada Tabel 3.37.

Karena nilai max yang diperoleh adalah 0.0432 dan kurang dari nilai *threshold* yang ditentukan yaitu 0.001 maka iterasi dapat dilanjutkan. Selain itu iterasi akan dihentikan ketika mencapai maximum iterasi yang ditentukan. Dari hasil perhitungan pada iterasi ke-0 diperoleh nilai α_i yang baru adalah 0.153. Nilai α_i tersebut akan digunakan sebagai perhitungan pada proses iterasi selanjutnya. Dalam proses perhitungan ini iterasi berhenti pada iterasi ke 10 sesuai dengan nilai *itermax* yang ditentukan. Berikut adalah perhitungan pada iterasi ke 10.

Iterasi ke 10:

- a) Menghitung Persamaan 2.22. $E_i = \sum_{j=1}^n \alpha_i D_{ij}$. Contoh perhitungan *record* pertama adalah sebagai berikut.

$$E_{10} = \begin{pmatrix} 0.3591 \times 0.25 + 0.3591 \times 0.25 + 0.3591 \times 0.25 + 0.3591 \times 0.25 + \\ 0.3591 \times 0.25 + 0.3591 \times 0.25 + 0.3591 \times 0.25 + 0.4882 \times -0.25 + \\ 0.4882 \times -0.25 + 0.4882 \times -0.25 + 0.4882 \times -0.25 \end{pmatrix} = 0.09$$

Perhitungan berlaku untuk seluruh data, dan hasil perhitungan dapat dilihat pada Tabel 3.39.

Tabel 3.39 Hasil Perhitungan Iterasi ke 10 pada Persamaan 2.22

$\alpha_i D_{ij}$	1	2	3	4	5	6	7	8	9	10	11	E_i
1	0.09	0.09	0.09	0.09	0.09	0.09	0.09	-0.11	-0.11	-0.11	-0.11	0.19
2	0.09	0.09	0.09	0.09	0.09	0.09	0.09	-0.11	-0.11	-0.11	-0.11	0.19
3	0.09	0.09	0.09	0.09	0.09	0.09	0.09	-0.11	-0.11	-0.11	-0.11	0.19
4	0.09	0.09	0.09	0.09	0.09	0.09	0.09	-0.11	-0.11	-0.11	-0.11	0.19
5	0.09	0.09	0.09	0.09	0.09	0.09	0.09	-0.11	-0.11	-0.11	-0.11	0.19
6	0.09	0.09	0.09	0.09	0.09	0.09	0.09	-0.11	-0.11	-0.11	-0.11	0.19
7	0.09	0.09	0.09	0.09	0.09	0.09	0.09	-0.11	-0.11	-0.11	-0.11	0.19
8	-0.09	-0.09	-0.09	-0.09	-0.09	-0.09	-0.09	0.11	0.11	0.11	0.11	-0.19
9	-0.09	-0.09	-0.09	-0.09	-0.09	-0.09	-0.09	0.11	0.11	0.11	0.11	-0.19
10	-0.09	-0.09	-0.09	-0.09	-0.09	-0.09	-0.09	0.11	0.11	0.11	0.11	-0.19
11	-0.09	-0.09	-0.09	-0.09	-0.09	-0.09	-0.09	0.11	0.11	0.11	0.11	-0.19

- b) Menghitung Nilai $\delta\alpha_i$ dengan Persamaan 2.23. $\delta\alpha_i = \min \{ \max [\gamma (1 - E_i), -\alpha_i], C - \alpha_i \}$. Contoh perhitungan record pertama adalah $\delta\alpha_i = \min \{ \max [0.04 (1 - 0.19), -0.3591], 1 - 0.3591 \}$. Perhitungan berlaku untuk seluruh data, dan hasil perhitungan dapat dilihat pada Tabel 3.40.

Tabel 3.40 Hasil Perhitungan Iterasi ke 10 pada Persamaan 2.23

	1	2	3	4	5	6	7	8	9	10	11
$\delta\alpha_i$	0.032	0.032	0.032	0.032	0.032	0.032	0.032	0.047	0.047	0.047	0.047

- c) Memperbarui Nilai α_i dengan Persamaan 2.24. $\alpha_i = \alpha_i + \delta\alpha_i$. Contoh perhitungan record pertama adalah $\alpha_i = 0.3591 + 0.0325 = 0.3916$. Perhitungan berlaku untuk seluruh data, dan hasil perhitungan dapat dilihat pada Tabel 3.41.

Tabel 3.41 Hasil Perhitungan Iterasi ke 10 pada Persamaan 2.24

	1	2	3	4	5	6	7	8	9	10	11
α_i	0.391	0.391	0.391	0.391	0.391	0.391	0.391	0.488	0.488	0.488	0.488

- d) Proses selanjutnya adalah melakukan pengecekan iterasi. Nilai max dapat dilihat dari nilai keseluruhan data pada Tabel 3.40. Karena nilai max yang diperoleh adalah 0.0475 dan kurang dari nilai *threshold* yang ditentukan yaitu 0.001 maka iterasi dapat dilanjutkan. Nilai *itermax* yaitu 10 sudah terpenuhi, maka seluruh proses dihentikan.

Setelah dilakukan perhitungan *training* menggunakan *sequential training*, proses selanjutnya adalah mencari nilai w (*weight*) dan b (posisi bidang relatif terhadap pusat koordinat) dengan menggunakan proses *sequential training*. Nilai $w \cdot x^+$ adalah nilai dari salah satu *support vector* dari *class positive*. x^+ memiliki nilai bobot dan nilai awal lebih besar. Sedangkan $w \cdot x^-$ adalah nilai dari salah satu *support vector* dari *class negative*. x^- memiliki nilai bobot dan nilai awal lebih kecil. Data awal setelah dinormalisasi dikali dengan data yang mempunyai nilai *support vector positive* atau *negative*.

Berikut adalah salah satu contoh perhitungan untuk $K(x_i, x^+)$, dan untuk perhitungan $K(x_i, x^-)$ dilakukan dengan contoh yang sama. Data nilai x_i untuk class $x^+ = 0.008$, dan x_i untuk class $x^- = 0.008$, maka hasil perhitungan adalah sebagai berikut.

$$K(x_i, x^+) = \left(\begin{array}{l} 0.008 \times 0.008 + 0.006 \times 0.045 + 0.005 \times 0.005 + 0.003 \times 0.003 \\ + 0.004 \times 0.002 + 0.006 \times 0.006 + 0.004 \times 0.026 + 0.004 \times 0.004 \\ + 0.009 \times 0.009 + 0.004 \times 0.004 + 0.006 \times 0.037 + 0.004 \times 0.004 \\ + 0.004 \times 0.002 + 0.006 \times 0.006 + 0.002 \times 0.002 + 0.005 \times 0.005 \\ + 0.003 \times 0.003 + 0.002 \times 0.002 + 0.002 \times 0.012 + 0.002 \times 0.013 \\ + 0.006 \times 0.002 + 0.002 \times 0.014 \end{array} \right)^2 = 0.0000011$$

Data nilai x_i untuk class $x^- = 0.008$, dan x_i untuk class $x^- = 0.008$, maka hasil perhitungan adalah sebagai berikut.

$$K(x_i, x^-) = \left(\begin{array}{l} 0.008 \times 0.008 + 0.045 \times 0.045 + 0.005 \times 0.005 + 0.003 \times 0.003 \\ + 0.002 \times 0.002 + 0.006 \times 0.006 + 0.026 \times 0.026 + 0.004 \times 0.004 \\ + 0.009 \times 0.009 + 0.004 \times 0.004 + 0.037 \times 0.037 + 0.004 \times 0.004 \\ + 0.002 \times 0.002 + 0.006 \times 0.006 + 0.002 \times 0.002 + 0.005 \times 0.005 \\ + 0.003 \times 0.003 + 0.002 \times 0.002 + 0.012 \times 0.012 + 0.013 \times 0.013 \\ + 0.002 \times 0.002 + 0.014 \times 0.014 \end{array} \right)^2 = 0.0000236$$

Untuk hasil perhitungan secara keseluruhan untuk mendapatkan nilai $K(x_i, x^+)$ dan nilai $K(x_i, x^-)$ dapat dilihat pada Tabel 3.42.

Tabel 3.42 Hasil Perhitungan $K(x_i, x^+)$ dan $K(x_i, x^-)$

Id_SV	$K(x_i, x^+)$	$K(x_i, x^-)$
KL_1	0.0000011	0.0000236
KL_2	0.0000016	0.0000084
KL_3	0.0000003	0.0000025
KL_4	0.0000014	0.0000183
KL_5	0.0000018	0.0000094
KL_6	0.0000014	0.0000177
KL_7	0.0000015	0.0000011
KL_9	0.0000003	0.0000011
KL_10	0.0000003	0.0000011
KL_11	0.0000002	0.0000011
KL_12	0.0000003	0.0000011

Setelah diperoleh nilai dari masing-masing *class* langkah selanjutnya adalah menghitung nilai w . Untuk proses perhitungan salah satu data adalah dengan Persamaan 2.10.

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

Data nilai $\alpha_i = 0.3916$, $y_i = 1$, $K(x_i, x^+) = 0.0000011$, maka hasil perhitungan adalah

$$\Sigma w.x^+ = \left(\begin{array}{l} 1 \times 0.3916 \times 0.0000011 + 1 \times 0.3916 \times 0.0000016 \\ + 1 \times 0.3916 \times 0.0000003 + 1 \times 0.3916 \times 0.0000014 \\ + 1 \times 0.3916 \times 0.0000018 + 1 \times 0.3916 \times 0.0000014 \\ + 1 \times 0.3916 \times 0.0000015 + -1 \times 0.4882 \times 0.0000003 \\ + -1 \times 0.4882 \times 0.0000003 + -1 \times 0.4882 \times 0.0000002 \\ + -1 \times 0.4882 \times 0.0000003 \end{array} \right) = 0.0000031$$

Data nilai $\alpha_i = 0.3916$, $y_i = 1$, $K(x_i, x^-) = 0.0000236$, maka hasil perhitungan adalah sebagai berikut.

$$\Sigma w.x^- = \left(\begin{array}{l} 1 \times 0.3916 \times 0.0000236 + 1 \times 0.3916 \times 0.0000084 \\ + 1 \times 0.3916 \times 0.0000025 + 1 \times 0.3916 \times 0.0000183 \\ + 1 \times 0.3916 \times 0.0000094 + 1 \times 0.3916 \times 0.0000177 \\ + 1 \times 0.3916 \times 0.0000186 + -1 \times 0.4882 \times 0.0000011 \\ + -1 \times 0.4882 \times 0.0000011 + -1 \times 0.4882 \times 0.0000011 \\ + -1 \times 0.4882 \times 0.0000011 \end{array} \right) = 0.0000364$$

Hasil perhitungan nilai $w.x^+$ dan $w.x^-$ dapat dilihat pada Tabel 3.43.

Tabel 3.43 Hasil Perhitungan $w.x^+$ dan $w.x^-$

Id_SV	$w.x^+$	$w.x^-$
KL_1	0.0000004	0.0000092
KL_2	0.0000006	0.0000033
KL_3	0.0000001	0.0000010
KL_4	0.0000006	0.0000072
KL_5	0.0000007	0.0000037
KL_6	0.0000006	0.0000069
KL_7	0.0000006	0.0000073
KL_9	-0.0000001	-0.0000005
KL_10	-0.0000001	-0.0000006
KL_11	-0.0000001	-0.0000005
KL_12	-0.0000001	-0.0000006
Σ	0.0000031	0.0000364

Langkah selanjutnya adalah mencari nilai b dengan menggunakan Persamaan 2.10.

$$b = -\frac{1}{2}(w \cdot x^+ + w \cdot x^-)$$

$$= -\frac{1}{2}((0.0000031) + (0.0000364)) = -0.0000197775$$

3.4.4 Perhitungan Manual Proses *Testing* SVM

Setelah dilakukan proses *training* langkah selanjutnya adalah melakukan proses *testing* dengan menggunakan data *testing*. Data *testing* sebelumnya dihitung menggunakan *kernel* dan hasil perhitungan dapat dilihat pada Tabel 3.31. Tahap proses *testing* SVM dilakukan untuk mendapatkan nilai $f(x)$ dengan mengimplementasikan Persamaan 2.12. Hasil klasifikasi dapat dihitung dengan Persamaan 2.13.

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

Untuk proses perhitungan salah satu data dengan Persamaan 2.21 adalah sebagai berikut. Untuk langkah awal adalah menghitung *kernel* data *testing*. Pada bagian ini contoh perhitungan data *testing* dilakukan untuk data pada KL_8 dan Kel 15.

a. Perhitungan *testing* untuk KL_8.

Jika nilai pada *record* pertama KL_1 hasil normalisasi data *training* = 0,008 dan nilai pada *record* pertama hasil normalisasi data *testing* = 0,008, maka hasil perhitungan *kernel* pada *record* KL_1 data *training* adalah sebagai berikut.

$$K(x_i, x) = \left(\begin{array}{l} 0.008 \times 0.008 + 0.045 \times 0.045 + 0.005 \times 0.005 \\ + 0.003 \times 0.003 + 0.002 \times 0.002 + 0.046 \times 0.046 \\ + 0.004 \times 0.026 + 0.004 \times 0.004 + 0.009 \times 0.009 \\ + 0.015 \times 0.004 + 0.037 \times 0.037 + 0.004 \times 0.004 \\ + 0.002 \times 0.002 + 0.006 \times 0.006 + 0.009 \times 0.002 \\ + 0.005 \times 0.005 + 0.003 \times 0.003 + 0.002 \times 0.002 \\ + 0.012 \times 0.012 + 0.002 \times 0.013 + 0.002 \times 0.002 \\ + 0.002 \times 0.014 \end{array} \right)^2 = 0.0000186$$

Jika nilai pada *record* kedua KL_2 hasil normalisasi data *training* = 0,073 dan nilai pada *record* pertama KL_8 data *testing* = 0,008, maka hasil perhitungan *kernel* pada *record* KL_2 data *training* adalah sebagai berikut.

$$K(x_i, x) = \left(\begin{array}{l} 0.008 \times 0.073 + 0.045 \times 0.006 + 0.005 \times 0.005 \\ + 0.003 \times 0.003 + 0.002 \times 0.002 + 0.046 \times 0.006 \\ + 0.004 \times 0.004 + 0.004 \times 0.015 + 0.009 \times 0.009 \\ + 0.015 \times 0.004 + 0.037 \times 0.037 + 0.004 \times 0.004 \\ + 0.002 \times 0.002 + 0.006 \times 0.006 + 0.009 \times 0.009 \\ + 0.005 \times 0.005 + 0.003 \times 0.003 + 0.002 \times 0.019 \\ + 0.012 \times 0.002 + 0.002 \times 0.002 + 0.002 \times 0.002 \\ + 0.002 \times 0.014 \end{array} \right)^2 = 0.0000088$$

Tahap selanjutnya mencari nilai $\alpha_i y_i K(x_i, x)$. Diketahui data nilai $\alpha_i = 0.3916$, dan nilai $y_i = 1$, maka hasil perhitungan adalah sebagai berikut.

Perhitungan untuk data *training* KL_1 = $0.3916 \times (1) 0.0000186 = 0.0000073$

Perhitungan untuk data *training* KL_2 = $0.3916 \times (1) 0.0000088 = 0.0000034$

Hasil perhitungan KL_8, untuk mendapatkan nilai *kernel* data *testing* dan mencari nilai $\alpha_i y_i K(x_i, x)$ dapat dilihat pada Tabel 3.44.

Tabel 3.44 Hasil Perhitungan $K(x_i, x)$ dan $\alpha_i y_i K(x_i, x)$

Id_SV	$K(x_i, x)$	$\alpha_i y_i K(x_i, x)$
KL_1	0.0000186	0.0000073
KL_2	0.0000088	0.0000034
KL_3	0.0000017	0.0000007
KL_4	0.0000367	0.0000144
KL_5	0.0000222	0.0000087
KL_6	0.0000374	0.0000146
KL_7	0.0000387	0.0000152
KL_9	0.0000014	-0.0000007
KL_10	0.0000015	-0.0000007
KL_11	0.0000014	-0.0000007
KL_12	0.0000015	-0.0000007
Σ		0.0000614

Proses selanjutnya adalah menghitung nilai $f(x)$ dengan menggunakan Persamaan 2.12. Kemudian menghitung fungsi klasifikasi dengan menggunakan Persamaan 2.13.

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

$$= 0.0000614 + (-0.0000197775) = 0.0000416$$

$$\text{Fungsi Klasifikasi} = \text{sign}(0.0000416) = 1$$

Tahap ini dilakukan untuk seluruh data *testing*, dan hasil akhir dari seluruh data adalah seperti pada Tabel 3.46.

b. Perhitungan *testing* untuk KL₁₅.

Jika nilai pada *record* pertama KL₁ hasil normalisasi data *training* = 0,008 dan nilai pada *record* pertama hasil normalisasi data *testing* = 0,008, maka hasil perhitungan *kernel* pada *record* pertama KL₁ data *training* adalah sebagai berikut.

$$K(x_i, x) = \left(\begin{array}{l} 0.008 \times 0.008 + 0.006 \times 0.045 + 0.005 \times 0.005 \\ + 0.013 \times 0.003 + 0.002 \times 0.002 + 0.006 \times 0.006 \\ + 0.004 \times 0.035 + 0.004 \times 0.009 + 0.004 \times 0.003 \\ + 0.006 \times 0.004 + 0.004 \times 0.004 + 0.004 \times 0.002 \\ + 0.006 \times 0.006 + 0.002 \times 0.002 + 0.005 \times 0.005 \\ + 0.003 \times 0.002 + 0.002 \times 0.002 + 0.002 \times 0.012 \\ + 0.002 \times 0.013 + 0.002 \times 0.013 + 0.002 \times 0.002 \\ + 0.002 \times 0.014 \end{array} \right)^2 = 0.0000011$$

Jika nilai pada *record* kedua KL₂ hasil normalisasi data *training* = 0,073 dan nilai pada *record* pertama KL₁₅ data *testing* = 0,008, maka hasil perhitungan *kernel* pada *record* pertama KL₁ adalah sebagai berikut.

$$K(x_i, x) = \left(\begin{array}{l} 0.008 \times 0.073 + 0.006 \times 0.006 + 0.005 \times 0.005 \\ + 0.013 \times 0.003 + 0.002 \times 0.002 + 0.006 \times 0.006 \\ + 0.004 \times 0.004 + 0.004 \times 0.005 + 0.004 \times 0.009 \\ + 0.006 \times 0.004 + 0.004 \times 0.004 + 0.004 \times 0.004 \\ + 0.006 \times 0.002 + 0.002 \times 0.006 + 0.005 \times 0.009 \\ + 0.003 \times 0.005 + 0.002 \times 0.003 + 0.002 \times 0.019 \\ + 0.001 \times 0.002 + 0.002 \times 0.002 + 0.002 \times 0.002 \\ + 0.002 \times 0.014 \end{array} \right)^2 = 0.0000017$$

Tahap selanjutnya mencari nilai $\alpha_i y_i K(x_i, x)$. Diketahui data nilai $\alpha_i = 0.3916$, dan nilai $y_i = 1$. maka hasil perhitungan adalah sebagai berikut.

Perhitungan untuk data KL_1 = $0.3916 \times (1)0.0000011 = 0.0000004$

Perhitungan untuk data KL_2 = $0.3916 \times (1)0.0000017 = 0.0000007$

Hasil perhitungan KL_15, untuk mendapatkan nilai *kernel* data *testing* dan mencari nilai $\alpha_i y_i K(x_i, x)$ dapat dilihat pada Tabel 3.45.

Tabel 3.45 Hasil Perhitungan $K(x_i, x)$ dan $\alpha_i y_i K(x_i, x)$

Id_SV	$K(x_i, x)$	$\alpha_i y_i K(x_i, x)$
KL_1	0.0000011	0.0000004
KL_2	0.0000017	0.0000007
KL_3	0.0000005	0.0000002
KL_4	0.0000014	0.0000006
KL_5	0.0000021	0.0000008
KL_6	0.0000015	0.0000006
KL_7	0.0000015	0.0000006
KL_9	0.0000002	-0.0000001
KL_10	0.0000004	-0.0000002
KL_11	0.0000002	-0.0000001
KL_12	0.0000004	-0.0000002
Σ		0.0000032

Proses selanjutnya adalah menghitung nilai *sign* dengan Persamaan 2.12. Kemudian menghitung fungsi klasifikasi dengan menggunakan Persamaan 2.13.

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

$$= 0.000032 + (-0.0000197775) = -0.0000165$$

$$\text{Hasil Klasifikasi} = \text{sign}(-0.0000165) = -1$$

Proses perhitungan untuk mendapatkan hasil klasifikasi dilakukan untuk keseluruhan data *testing* yang telah terpilih. Hasil klasifikasi dengan nilai 1 adalah termasuk dalam kelompok masyarakat yang layak untuk menerima RASKIN. Sedangkan hasil klasifikasi dengan nilai -1 termasuk dalam kelompok masyarakat yang tidak layak untuk menerima RASKIN. Pada Tabel 3.46 menjelaskan tentang *class actual* dan hasil klasifikasi penerima RASKIN dari perhitungan data *testing*.

Tabel 3.46 Data hasil Perhitungan $f(x)$ dan Klasifikasi

Id_Data	Nilai $f(x)$	Hasil Klasifikasi	Actual Class
KL_8	0.000042	1	1
KL_16	0.000013	1	1
KL_17	-0.000001	-1	1
KL_13	-0.000017	-1	-1
KL_14	-0.000001	-1	-1
KL_15	-0.0000165	-1	-1
KL_18	-0.0000176	1	1
KL_19	-0.0000167	-1	-1
KL_20	-0.0000165	-1	-1

Hasil akhir dari seluruh perhitungan data *testing* dan hasil klasifikasi dapat dilihat pada Lampiran 9 yaitu Tabel Hasil perhitungan data *testing* dan hasil klasifikasi *class*.

3.4.5 Perhitungan Manual Nilai Akurasi

Mengukur performansi dan mengukur tingkat keberhasilan atau membandingkan hasil dari proses implementasi yang telah dilakukan dengan standar yang telah ditetapkan dilakukan proses evaluasi atau pengujian sistem. Proses pengujian dilakukan untuk menentukan nilai akurasi sistem dari nilai yang diklasifikasikan secara tepat. Salah satu model evaluasi adalah dengan membuat tabel *confusion matriks* seperti pada Tabel 3.47.

Tabel 3.47 Model Evaluasi *confusion matriks*

	Predicated Class		
	Class 1	Class -1	Jumlah
Actual Class 1	3	1	4
Actual Class -1	0	5	5
Jumlah	3	6	9

Untuk menghitung nilai akurasi sistem dilakukan perhitungan dengan menggunakan Persamaan 2.26.

$$\text{Clasification accuracy (\%)} = \frac{TP + TN}{TP + FP + FN + TN} \times 100\%$$

Maka, nilai akurasi yang diperoleh adalah

$$\text{Akurasi} = \frac{3 + 5}{3 + 0 + 5 + 1} \times 100\% = 88,89\%$$

3.5 Perancangan Antarmuka (*Interface*)

Antarmuka (*Interface*) yang dirancang terdiri dari beberapa bagian yaitu untuk menghitung bobot kriteria dan mengetahui nilai konsistensi, bagian kedua untuk menghitung bobot dari masing-masing sub kriteria dan kriteria terpilih, kemudian bagian load data, proses normalisasi data, bagian perhitungan kernel, bagian proses *training* SVM, bagian *testing* SVM dan bagian pengujian akurasi sistem. Berikut adalah penjelasan rancangan antarmuka (*interface*) dari sistem yang akan ditunjukkan pada Gambar 3.11 sampai Gambar 3.19.

1. Proses Perhitungan bobot kriteria dan konsistensi matriks.

Pada bagian ini terdapat tombol *choose file* untuk memilih lokasi file matriks perbandingan, kemudian terdapat tombol *Load Data* untuk menampilkan matriks kriteria yang telah dipilih berdasarkan lokasi dan nama sheet sebelumnya. Terdapat pilihan load data yang akan digunakan dalam pengujian. Selanjutnya untuk tombol Hitung Proses AHP dilakukan proses perhitungan matriks, bobot hingga cek konsistensi. Rancangan antarmuka proses ini dapat dilihat pada Gambar 3.12.

Gambar 3.12 Form Perhitungan bobot kriteria dan konsistensi matriks

2. Proses Perhitungan bobot masing-masing sub kriteria.

Pada bagian ini akan menampilkan hasil perhitungan data bobot sub kriteria. Tombol *Load* dan proses perhitungan untuk memulai proses perhitungan sub kriteria menggunakan AHP. Pada proses ini Matriks Sub Kriteria akan ditampilkan beserta hasil perhitungan bobot sub kriteria. Proses selanjutnya adalah memasukkan nilai *threshold* yang akan digunakan sebagai batasan nilai bobot sub kriteria yang akan digunakan, kemudian pada tombol *sorting* dilakukan proses berdasarkan batasan nilai yang telah dimasukkan sebelumnya sehingga diperoleh bobot sub kriteria terpilih. Pada bagian ini akan ditampilkan sub kriteria terpilih yaitu nilai diatas *threshold* dan nilai dibawah *threshold*. Nilai sub kriteria yang terpilih akan digunakan untuk proses selanjutnya. Rancangan antarmuka proses ini dapat dilihat pada Gambar 3.13.

Gambar 3.13 Form Perhitungan bobot sub kriteria

3. Proses *load Dataset* Awal

Pada bagian ini adalah untuk melakukan *load dataset* awal penerima RASKIN. Pada proses ini terdapat dua bagian proses. Bagian pertama *load data training* dan *testing* akan ditampilkan seluruh data warga penerima RASKIN di daerah Ronggomulyo Kabupaten Tuban. Proses ini dilakukan secara otomatis setelah menekan tombol Proses Selanjutnya pada bagian setelah dilakukan *sorting* nilai *threshold*. Tombol *Load Dataset* akan menampilkan dataset yang digunakan pada proses pengujian berdasarkan nama sheet. Bagian kedua akan ditampilkan *dataset* berdasarkan sub kriteria yang telah terpilih. Proses ini dilakukan dengan menekan tombol proses *dataset*. Sehingga semua *dataset* akan ditampilkan dari sub kriteria terpilih melalui nilai *threshold* tertentu sebagai batasannya. Tombol Proses pengujian menunjukkan proses tahapan pengujian data *training* dan data *testing* SVM. Rancangan antarmuka proses ini dapat dilihat pada Gambar 3.14.

Project AHP-SVM

Implementasi Metode AHP-SVM
Klasifikasi Penerima Beras untuk Masyarakat Miskin (RASKIN)
Studi Kasus Kelurahan Ronggomulyo Kabupaten Tuban

Data Matrix Kriteria | Perhitungan Sub Bobot Kriteria AHP | Load Data Awal | Pengujian

Load Data Training dan Testing

Nama Sheet Dataset: Dataset Load Dataset

Dataset Berdasarkan Sub Kriteria Terpilih

Proses Dataset Proses Pengujian

Gambar 3.14 Form *Load Dataset* Awal

4. Proses Pengujian

Pada bagian ini terdiri dari beberapa proses untuk data *training* dan data *testing*. Proses pertama adalah load *dataset training* dan *dataset testing*. Pada form ini jumlah data *training* dan data *testing* yang akan digunakan dapat disesuaikan berdasarkan rasio perbandingan yang dapat dipilih. Terdapat dua pilihan yaitu pilih data secara random dan load data dari file berdasarkan jumlah rasio perbandingan yang ditentukan. Apabila data yang akan digunakan berdasarkan data yang telah tersimpan dapat memasukkan tempat penyimpanan data dan nama sheet data *training* dan data *testing*. Tombol load akan menampilkan data *training* dan data *testing* secara terpisah pada bagian yang telah disediakan. Jumlah data *training* dan data *testing* yang digunakan berdasarkan rasio perbandingan akan diketahui pada label yang telah tersedia. Kemudian tombol Normalisasi adalah untuk menunjukkan hasil perhitungan normalisasi data yang akan ditampilkan pada proses berikutnya. Rancangan antarmuka proses ini dapat dilihat pada Gambar 3.15.

Gambar 3.15 Form *Load Dataset* dan Normalisasi

5. Proses Normalisasi *Dataset*

Pada bagian ini akan menampilkan hasil proses normalisasi *dataset* awal berdasarkan sub kriteria terpilih. Proses normalisasi dihitung sesuai dengan Persamaan 2.3. Normalisasi dilakukan untuk seluruh data, baik data *training* maupun data *testing*. Normalisasi dilakukan pada nilai setiap atribut data dengan skala data input sesuai dengan *range* fungsi penilaian yang digunakan. Pada bagian ini terdapat tombol Proses perkalian bobot yang menunjukkan bahwa proses selanjutnya adalah perkalian data normalisasi dan data bobot sub kriteria. Proses secara rinci akan dijelaskan pada bagian selanjutnya. Rancangan antarmuka proses ini dapat dilihat pada Gambar 3.16.

Project AHP-SVM

Implementasi Metode AHP-SVM
Klasifikasi Penerima Beras untuk Masyarakat Miskin (RASKIN)
Studi Kasus Kelurahan Ronggomulyo Kabupaten Tuban

Data Matrix Kriteria | Perhitungan Sub Bobot Kriteria AHP | Load Data Awal | Pengujian

Dataset | Normalisasi Dataset | Bobot setelah Normalisasi | Sekuensial Training | Hasil Training dan Nilai SV | Hasil Proses Testing

Dataset Setelah Normalisasi

Proses Perkalian Bobot

Gambar 3.16 Form Hasil Normalisasi

6. Proses Perhitungan Bobot setelah dilakukan Normalisasi

Pada bagian ini adalah untuk menunjukkan hasil perhitungan bobot sub kriteria. Hasil perhitungan dari perkalian antara data hasil perhitungan normalisasi dengan nilai bobot sub kriteria. Pada bagian ini dapat dipilih jenis *kernel* yang akan digunakan dalam proses perhitungan selanjutnya. *Kernel* yang dapat digunakan adalah *polynomial*, dan *kernel Gaussian RBF*. Selain itu juga dapat memasukkan nilai λ (*lamda*), nilai σ (*sigma*), dan nilai d (*degree*). Nilai λ (*lamda*) digunakan dalam proses perhitungan *sequential training*. Sedangkan nilai σ (*sigma*) digunakan untuk proses perhitungan pada *kernel Gaussian RBF*. Tombol proses *training* menunjukkan proses selanjutnya untuk Perhitungan *sequential training SVM*. Rancangan antarmuka proses ini dapat dilihat pada Gambar 3.17.

Gambar 3.17 Form Perhitungan Bobot Setelah Normalisasi

7. Proses *Sequential Training* SVM

Pada bagian ini adalah untuk menunjukkan hasil perhitungan data *training* dengan metode *sequential training* SVM. Terdapat beberapa *textbox* untuk menginputkan parameter yang digunakan dalam proses *training* SVM. Parameter yang harus dimasukkan antara lain nilai γ (*Gamma*), nilai ϵ (*Epsilon*), nilai C (*Complexity*), dan nilai *itermax*. Tombol *Proses Training* SVM menunjukkan proses perhitungan data *training* yaitu dengan menggunakan *kernel* yang telah dipilih pada bagian sebelumnya dan hasil perhitungan Matriks Hessian. Tombol *Proses Training* menunjukkan proses *sequential training*. Rancangan antarmuka proses ini dapat dilihat pada Gambar 3.18.

Project AHP-SVM

Implementasi Metode AHP-SVM
Klasifikasi Penerima Beras untuk Masyarakat Miskin (RASKIN)
Studi Kasus Kelurahan Ronggomulyo Kabupaten Tuban

Data Matrix Kriteria | Perhitungan Sub Bobot Kriteria AHP | Load Data Awal | Pengujian

Dataset | Normalisasi Dataset | Bobot setelah Normalisasi | Sekuensial Training | Hasil Training dan Nilai SV | Hasil Proses Testing

Parameter SVM

γ (Gamma): 0.01 ϵ (Epsilon): 0.001 itermax: 10

C: 1 Nilai ci: 0 Hasil Proses Training

Perhitungan Kernel

Matrix Heissian

Gambar 3.18 Form Perhitungan *Sequential Training* SVM

8. Hasil *Training SVM* dan perhitungan nilai *Support Vector*

Pada bagian ini adalah untuk menunjukkan hasil proses *sequential training SVM*. Terdapat beberapa *datagridview* untuk menunjukkan hasil proses iterasi. Iterasi akan berhenti jika salah satu kondisi terpenuhi, yaitu apabila iterasi telah mencapai nilai konvergen ($|\delta\alpha_i| < \epsilon$) atau mencapai nilai *itermax*. Pada label iterasi berhenti akan ditunjukkan iterasi berhenti pada iterasi ke berapa. Proses *sequential training* terdiri dari tiga proses yaitu menentukan nilai E_i , $\delta\alpha_i$, dan α_i . Pada bagian ini juga dilakukan perhitungan untuk mencari nilai *Support Vector positive* dan *negative*. Tombol proses *testing SVM* menunjukkan proses selanjutnya untuk perhitungan *dataset testing SVM* sampai menemukan nilai klasifikasi dan hasil akurasi sistem. Rancangan antarmuka proses ini dapat dilihat pada Gambar 3.19.

Project AHP-SVM

Implementasi Metode AHP-SVM
Klasifikasi Penerima Beras untuk Masyarakat Miskin (RASKIN)
Studi Kasus Kelurahan Ronggomulyo Kabupaten Tuban

Data Matrix Kriteria | Perhitungan Sub Bobot Kriteria AHP | Load Data Awal | Pengujian

Dataset | Normalisasi Dataset | Bobot setelah Normalisasi | Sekuensial Training | Hasil Training dan Nilai SV | Hasil Proses Testing

Iterasi Berhenti Proses Testing

Hasil Nilai E_i

Hasil Nilai $\delta\alpha_i$

Hasil Nilai α_i

Proses Support Vector (+) dan Support Vector (-)

Gambar 3.19 Form Proses *Training SVM*

9. Hasil proses *dataset testing* dan akurasi sistem

Pada bagian ini adalah menunjukkan proses perhitungan *testing* SVM. Hasil ini diperoleh berdasarkan hasil perhitungan yang dilakukan pada bagian sebelumnya. Terdapat label nilai b yang menunjukkan nilai yang diperoleh dari hasil perhitungan sebelumnya. Perhitungan data *testing* dan hasil klasifikasi akan ditampilkan pada bagian yang tersedia. Pada bagian bawah terdapat Hasil Akhir akurasi sistem dan jumlah data benar. Hasil akhir akurasi sistem menunjukkan evaluasi tingkat keberhasilan penerapan metode pada sistem. Tombol Simpan Hasil akan menyimpan data *training* dan data *testing* dari setiap pengujian yang dilakukan dengan akurasi yang terbaik. Data yang disimpan kemudian akan digunakan sebagai proses pengujian selanjutnya. Rancangan antarmuka proses ini dapat dilihat pada Gambar 3.20.

Project AHP-SVM

Implementasi Metode AHP-SVM
Klasifikasi Penerima Beras untuk Masyarakat Miskin (RASKIN)
Studi Kasus Kelurahan Ronggomulyo Kabupaten Tuban

Data Matrix Kriteria | Perhitungan Sub Bobot Kriteria AHP | Load Data Awal | Pengujian

Dataset | Normalisasi Dataset | Bobot setelah Normalisasi | Sekuensial Training | Hasil Training dan Nilai SV | Hasil Proses Testing

Proses Testing
Nilai b (bias)

Hasil Akurasi Sistem : **Nilai** Jumlah Data Benar : **Nilai** Simpan Hasil

Gambar 3.20 Form Hasil Akhir Proses *Testing* SVM dan Akurasi Sistem

3.6 Pengujian Sistem

Setelah perancangan sistem dibuat, tahap selanjutnya adalah proses pengujian sistem. Pengujian sistem dilakukan untuk mengetahui tingkat akurasi dari penerapan metode AHP-SVM dalam sistem. Pada data penerima RASKIN yang asli sudah diketahui klasifikasi *Actual Class* dari masing-masing data. Data tersebut dapat digunakan sebagai pembandingan dengan data hasil perhitungan menggunakan metode AHP-SVM. Proses pengujian dilakukan menggunakan dengan *dataset* penerima RASKIN sebanyak 100 data yang terpilih secara acak.

Skenario pengujian yang akan adalah sebagai berikut.

1. Pengujian berdasarkan data *Training* dan data *Testing* terbaik yang diambil secara acak berdasarkan rasio perbandingan. Pada Tabel 3.48 ditunjukkan contoh tabel pengujian rasio data *Training* dan data *Testing*.

Tabel 3.48 Pengujian Rasio Data *Training* dengan Data *Testing*

Skenario Pengujian	Rasio Perbandingan Data <i>Training</i> dan Data <i>Testing</i>				
	90%:10%	80%:20%	70%:30%	60%:40%	50%:50%
1					
2					
3					
4					
5					
Rata-Rata Akurasi					

2. Pengujian akurasi berdasarkan data matriks perbandingan. Data matriks perbandingan yang digunakan akan dipilih berdasarkan data matriks terbaik yang kemudian akan digunakan untuk proses pengujian. Pada Tabel 3.49 ditunjukkan contoh tabel pengujian berdasarkan data matriks perbandingan.

Tabel 3.49 Pengujian berdasarkan matriks perbandingan

Skenario Pengujian	Data Matriks Perbandingan				
	Data 1	Data 2	Data 3	Data 4	Data 5
1					
2					
3					
4					
5					
Rata-rata Akurasi					

3. Pengujian terhadap data *Training* dan data *Testing* yang menunjukkan nilai akurasi terbaik, kemudian dihitung berdasarkan nilai *Threshold* yang berbeda. Pada Tabel 3.50 ditunjukkan contoh tabel pengujian data *Training* dan data *Testing* berdasarkan nilai *Threshold*.

Tabel 3.50 Pengujian berdasarkan nilai *Threshold*

Skenario Pengujian	Nilai <i>Threshold</i>				
	0.01	0.02	0.03	0.04	0.05
1					
2					
3					
4					
5					
Rata-rata Akurasi					

4. Pengujian akurasi berdasarkan *Kernel Polynomial*. Pada Tabel 3.51 ditunjukkan contoh tabel pengujian berdasarkan parameter d pada *Kernel Polynomial*.

Tabel 3.51 Pengujian Berdasarkan *Kernel Polynomial*

Skenario Pengujian	Nilai d (degree)				
	0.8	0.9	1	2	3
1					
2					
3					
4					
5					
Rata-rata Akurasi					

5. Pengujian akurasi berdasarkan *Kernel Gaussian RBF*. Pada Tabel 3.52 ditunjukkan contoh tabel pengujian berdasarkan *Kernel Gaussian RBF*.

Tabel 3.52 Pengujian Berdasarkan *Kernel Gaussian RBF*

Skenario Pengujian	Nilai σ (sigma)				
	0.8	0.9	1	2	3
1					
2					
3					
4					
5					
Rata-rata Akurasi					

6. Pengujian berdasarkan nilai parameter *Sequential Training SVM*, yaitu λ (*lamda*), nilai parameter γ (*gamma*), nilai parameter ϵ (*epsilon*), nilai parameter Maximum Iterasi (*itermax*), dan nilai parameter C (*Complexity*). Pada Tabel 3.53 sampai dengan Tabel 3.57 ditunjukkan tabel pengujian parameter proses *Sequential Training SVM*.

Tabel 3.53 Pengujian Berdasarkan Parameter λ *Sequential Training SVM*

Skenario Pengujian	Nilai λ (<i>lamda</i>)				
	0.1	0.5	1	1.5	2
1					
2					
3					
4					
5					
Rata-rata Akurasi					

Tabel 3.54 Pengujian Berdasarkan Parameter γ *Sequential Training SVM*

Skenario Pengujian	Nilai γ (<i>gamma</i>)				
	0.1	0.01	0.001	0.0001	0.00001
1					
2					
3					
4					
5					
Rata-rata Akurasi					

Tabel 3.55 Pengujian Berdasarkan Parameter ϵ (*epsilon*) *Sequential Training SVM*

Skenario Pengujian	Nilai ϵ (<i>epsilon</i>)				
	0.1	0.01	0.001	0.0001	0.00001
1					
2					
3					
4					
5					
Rata-rata Akurasi					

Tabel 3.56 Pengujian Berdasarkan Parameter *itermax Sequential Training SVM*

Skenario Pengujian	Maximum Iterasi (<i>itermax</i>)				
	10	100	300	500	1000
1					
2					
3					
4					
5					
Rata-rata Akurasi					

Tabel 3.57 Pengujian Berdasarkan Parameter C (*Complexity*) *Sequential Training SVM*

Skenario Pengujian	Nilai C (<i>Complexity</i>)				
	0.1	0.5	1	1.5	2
1					
2					
3					
4					
5					
Rata-rata Akurasi					