

IMPLEMENTASI ALGORITMA GENETIKA UNTUK OPTIMASI NILAI DENSITAS METODE DEMPSTER-SHAFFER PADA DETEKSI PENYAKIT SAPI

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh :
Desak Gede Saras Mutiara Abhisastra
NIM: 115060801111012



TEKNIK INFORMATIKA
PROGRAM TEKNOLOGI INFORMASI DAN ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2015

PENGESAHAN

IMPLEMENTASI ALGORITMA GENETIKA UNTUK OPTIMASI NILAI DENSITAS
METODE DEMPSTER-SHAFFER PADA DETEKSI PENYAKIT SAPI

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :

Desak Gede Saras Mutiara Abhisastra

NIM: 115060801111012

Skripsi ini telah diuji dan dinyatakan lulus pada
7 Januari 2016

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Rekyan Regasari MP., S.T., M.T.

NIK: 770414 06 1 2 0257

Candra Dewi, S. Kom., M. Sc.

NIP: 19771114 200312 2 001

Mengetahui

Ketua Program Studi Informatika/Ilmu Komputer

Drs. Marji, M.T.

NIP: 19670801 199203 1001

PERNYATAAN ORISINALITAS

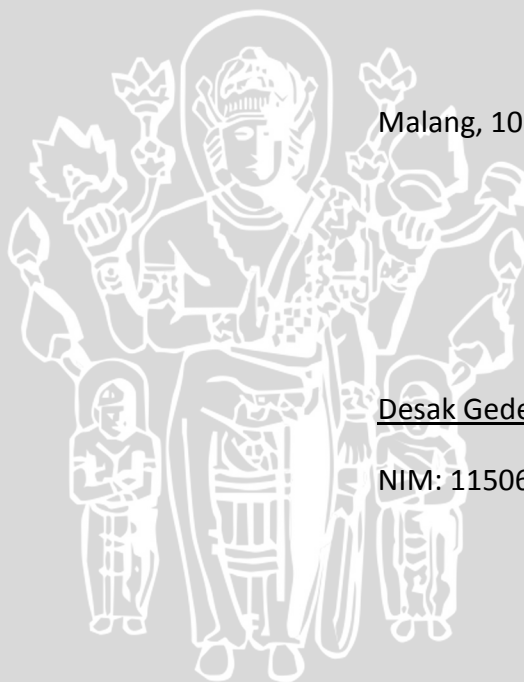
Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan 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, 10 Desember 2015

Desak Gede Saras M. A

NIM: 115060801111012



KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Ida Sang Hyang Widhi Wasa karena atas asung kertha wara nugraha-Nya penulis dapat menyelesaikan skripsi dengan judul “Implementasi Algoritma Genetika Untuk Optimasi Nilai Densitas Metode Dempster-Shafer pada Deteksi Penyakit Sapi”.

Skripsi ini diajukan untuk memenuhi sebagian persyaratan memperoleh gelar Sarjana Komputer.

Melalui kesempatan ini, penulis ingin menyampaikan rasa hormat dan terima kasih yang sebesar-besarnya kepada semua pihak yang telah memberikan bantuan dan dukungan selama penulisan skripsi, diantaranya:

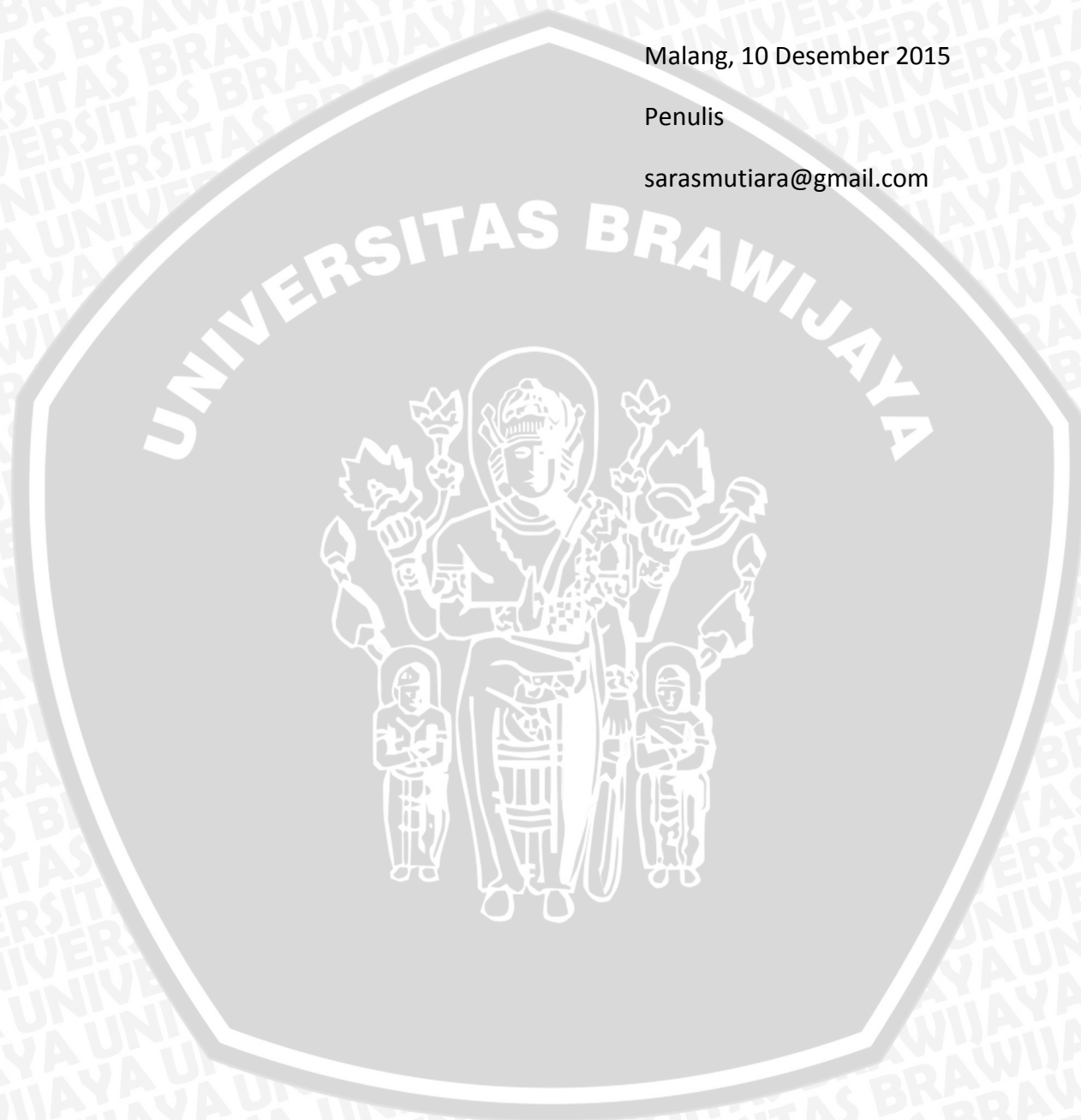
1. Rekyan Regasari MP., S.T., M.T. selaku dosen pembimbing I dan Candra Dewi, S. Kom, M.Sc. selaku dosen pembimbing II yang telah memberikan ilmu dan saran untuk menyelesaikan skripsi ini.
2. Ir. Sutrisno, MT. Sebagai Dekan Fakultas Ilmu Komputer Universitas Brawijaya.
3. Drs. Marji, M.T., selaku Ketua Program Studi Informatika / Ilmu Komputer Fakultas Ilmu Komputer (FILKOM) Universitas Brawijaya.
4. Denny Sagita Rusdianto, S.Kom., M.Kom., selaku dosen pembimbing akademik yang telah banyak memberikan saran dan motivasi selama masa perkuliahan.
5. Bapak Ir. I Dewa Putu Sastradi, Ibu Ir. Desak Made Seriastini, S.Pd., Adik Desak Nyoman Tamiya Bintang Abhisastra dan Kakak Ns. Desak Gede Lilik Eka Cipta Dewi dan seluruh keluarga besar atas nasehat, kasih sayang, perhatian, kesabaran, semangat dan doa yang diberikan kepada penulis.
6. drh. I Ketut Mertanadi selaku pakar yang telah membantu proses pengumpulan data skripsi.
7. Seluruh dosen Teknik Informatika Fakultas Ilmu Komputer Universitas Brawijaya atas kesediaan mendidik dan membagi ilmunya kepada penulis.
8. Seluruh staf dan karyawan Teknik Informatika Fakultas Ilmu Komputer Universitas Brawijaya yang telah membantu kelancaran pengerjaan skripsi.
9. Sahabat-sahabat penulis yaitu Ira Anis Anifa, Disya Nadia Putri, Irma Nur Jayanti, Febria Rafmadhanty dan Kentia Dea Hapsari yang telah memberikan semangat kepada penulis selama menempuh studi di Teknik Informatika Fakultas Ilmu komputer Universitas Brawijaya.
10. Teman-teman kos Watugong 39 yaitu Desak Putu Tantri Swastini, I Gusti Ayu Ade Anggi Wijaya, Gusti Ayu Ade Aprilia Rahardika, Ni Luh Indah Puspayani, Davia Werdiastu, Teva Delani, Made Arny Fariyanti, dan Made Cintia Yanthi atas dukungan dan semangat kepada penulis selama berada di Malang.
11. Semua pihak yang tidak dapat penulis sebutkan satu per satu yang terlibat baik secara langsung maupun yang tidak langsung demi terselenggaranya skripsi ini.

Hanya doa yang bisa penulis berikan semoga Tuhan Yang Maha Esa memberikan pahala serta balasan kebaikan yang berlipat. Penulis menyadari bahwa skripsi ini masih banyak kekurangan. Oleh karena itu, kritik dan saran yang bersifat membangun sangat diharapkan untuk menyempurnakan skripsi ini. Semoga skripsi ini membawa manfaat bagi semua pihak yang menggunakannya.

Malang, 10 Desember 2015

Penulis

sarasmutiara@gmail.com



ABSTRAK

Desak Gede Saras Mutiara Abhisastra, 2015 : Implementasi Algoritma Genetika Untuk Optimalisasi Nilai Densitas Metode Dempster-Shafer pada Deteksi Penyakit Sapi. Skripsi Program Studi Informatika / Ilmu Komputer, Fakultas Ilmu Komputer, Universitas Brawijaya, Malang.

**Dosen Pembimbing : Rekyan Regasari MP., S.T., M.T.;
Candra Dewi, S. Kom, M.Sc.**

Salah satu faktor yang menyebabkan ketidakakuratan dalam mendiagnosa adalah nilai densitas yang kurang optimal. Untuk mendapatkan nilai densitas yang konsisten dan optimal, maka diperlukan optimasi terhadap nilai densitas. Terdapat beberapa metode yang dapat digunakan dalam optimasi, yaitu algoritma genetika, *Ant Colony Algorithm*, dan lain sebagainya. Dari beberapa metode optimasi yang ada, penelitian ini menggunakan metode Algoritma Genetika karena pada beberapa penelitian mampu meningkatkan akurasi setelah melakukan optimasi menggunakan algoritma genetika. Metode yang digunakan dalam Algoritma Genetika yaitu *Extended Intermediate Crossover*, *Random Mutation* dan *Replacement Selection*.

Berdasarkan hasil pengujian dan analisis sistem parameter algoritma genetika yang baik untuk menentukan nilai densitas yaitu probabilitas *crossover* 0,8, probabilitas mutasi 0,8, jumlah generasi 450 dan ukuran populasi 15. Hasil perhitungan akurasi rata-rata sistem dengan menggunakan nilai densitas yang telah dioptimasi pada 18 kasus penyakit sapi menunjukkan akurasi sebesar 93,33%. Hasil akurasi ini mengalami kenaikan sebesar 4,44% dimana pada penelitian sebelumnya akurasi yang didapatkan sebesar 88,89% (Prihastuti, 2014).

Kata kunci : Algoritma Genetika, Optimalisasi Nilai Densitas, Dempster-Shafer, penyakit sapi

ABSTRACT

Desak Gede Saras Mutiara Abhisastra, 2015 : *Implementation of Genetic Algorithm To Optimize Density Dempster - Shafer at Cow Disease Detection. Undergraduate Thesis of Informatic / Computer Science Study Program, Computer Science Faculty, Brawijaya University, Malang*

Advisors : Rekyan Regasari MP., S.T., M.T.; Candra Dewi, S. Kom, M.Sc.

One of the factors that cause inaccuracies in diagnosing is the value of the density less optimal. To get the density consistent and optimal, then required optimize on the density. There are several methods that can be used to optimize, Genetic Algorithm, Ant Colony Algorithm, and others. From the several methods which are exist, this research uses Genetic Algorithm because at some researches its able to improve the accuracy after optimization using Genetic Algorithm. The method that are used in Genetic Algorithm namely extended intermediate crossover, random mutation and replacement selection.

Based on the result of examination and systems analysis of parameter of Genetic Algorithm which is good to determine the value of the density, namely the probability crossover 0,8, the probability of mutation 0,8, of generations 450 and size of the population 15. The results of average accuracy the system to use the density who has optimized on 18 cases of beef show accuracy of 93,33%. The accuracy of this increased by 4,44 % where previous research accuracy obtained by 88,89% (Prihastuti, 2014).

Keyword : Genetic Algorithm, Optimization of Density, Dempster-Shafer, Cow Disease

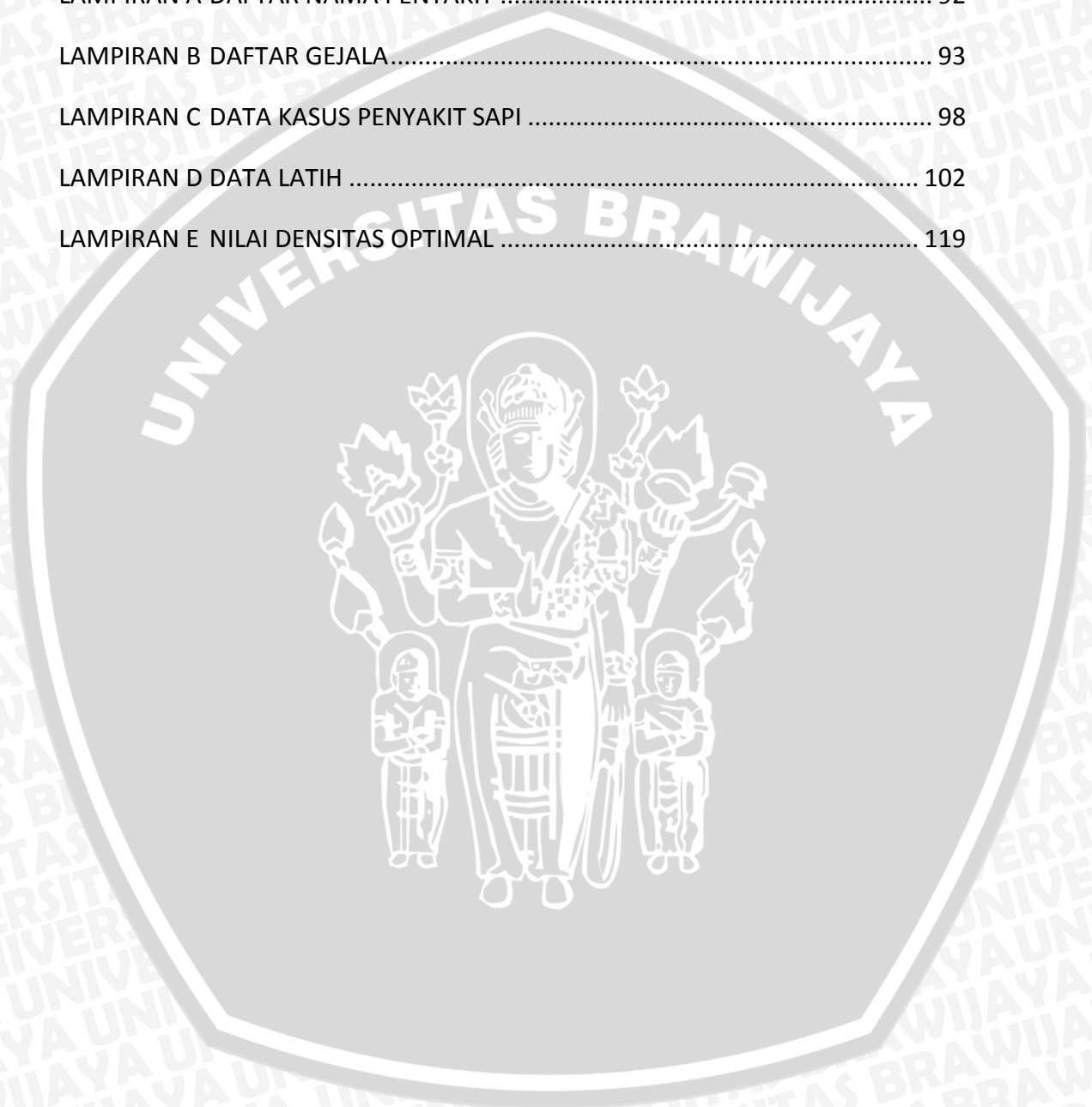
DAFTAR ISI

PENGESAHAN	ii
PERNYATAAN ORISINALITAS	iii
KATA PENGANTAR.....	iv
ABSTRAK	vi
ABSTRACT	vii
DAFTAR ISI	viii
DAFTAR TABEL.....	xii
DAFTAR GAMBAR.....	xiv
DAFTAR <i>SOURCE CODE</i>	xvi
DAFTAR LAMPIRAN	xvii
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan Penelitian.....	2
1.4 Manfaat Penelitian.....	3
1.5 Batasan Masalah	3
1.6 Sistematika Penulisan.....	3
BAB 2 LANDASAN KEPUSTAKAAN.....	4
2.1 Kajian Pustaka.....	4
2.2 Penyakit Sapi	5
2.2.1 Penyakit yang disebabkan oleh virus (Penyakit Viral)	5
2.2.2 Penyakit yang disebabkan oleh bakteri	10
2.2.3 Penyakit yang disebabkan oleh jamur	13
2.2.4 Penyakit yang disebabkan oleh parasit.....	14
2.2.5 Penyakit karena gangguan metabolisme	14

2.2.6	Penyakit karena gangguan pencernaan	15
2.2.7	Penyakit karena gangguan reproduksi	15
2.3	Dempster-Shafer	16
2.3.1	Aturan Kombinasi Dempster-Shafer	17
2.4	Algoritma Genetika	17
2.4.1	Representasi Kromosom dan Inisialisasi Populasi	18
2.4.2	Evaluasi	18
2.4.3	Tukar Silang (<i>Crossover</i>)	19
2.4.4	Mutasi	20
2.4.5	Seleksi	21
BAB 3	METODOLOGI PENELITIAN	25
3.1	Studi Literatur	25
3.2	Pengumpulan Data	26
3.3	Analisis Kebutuhan	27
3.4	Perancangan Sistem	27
3.5	Implementasi Sistem	28
3.6	Pengujian Sistem	28
3.7	Pengambilan Kesimpulan	31
BAB 4	PERANCANGAN	32
4.1	Analisis Kebutuhan	32
4.2	Perancangan Sistem	32
4.2.1	Inisialisasi Populasi Awal	34
4.2.2	<i>Fitness</i>	39
4.2.3	Tukar Silang (<i>Crossover</i>)	42
4.2.4	Mutasi	44
4.2.5	Seleksi	45
4.3	Perhitungan Manual	48

4.3.1	Inisialisasi Populasi Awal	48
4.3.2	Perhitungan <i>Fitness</i>	50
4.3.3	<i>Crossover</i>	54
4.3.4	Mutasi	57
4.3.5	Seleksi.....	59
4.4	Perancangan Antarmuka	60
BAB 5	IMPLEMENTASI.....	62
5.1	Spesifikasi Sistem.....	62
5.1.1	Spesifikasi Perangkat Keras.....	62
5.1.2	Spesifikasi Perangkat Lunak	62
5.2	Implementasi Sistem	63
5.2.1	Implementasi Inisialisasi Populasi Awal.....	63
5.2.2	Implementasi <i>Fitness</i>	64
5.2.3	Implementasi <i>Crossover</i>	70
5.2.4	Implementasi Mutasi.....	71
5.2.5	Implementasi Seleksi	72
5.3	Implementasi Antarmuka.....	73
5.3.1	Tampilan Proses Algoritma Genetika.....	74
5.3.2	Tampilan Nilai Densitas Optimal.....	75
5.3.3	Tampilan Hasil Akurasi	75
BAB 6	PENGUJIAN	77
6.1	Pengujian Parameter Algoritma Genetika.....	77
6.1.1	Pengujian Probabilitas <i>Crossover</i> dan Probabilitas Mutasi	77
6.1.2	Pengujian Jumlah Generasi	79
6.1.3	Pengujian Ukuran Populasi	80
6.2	Nilai Densitas Optimal	82
6.3	Pengujian Akurasi	84
BAB 7	PENUTUP.....	89

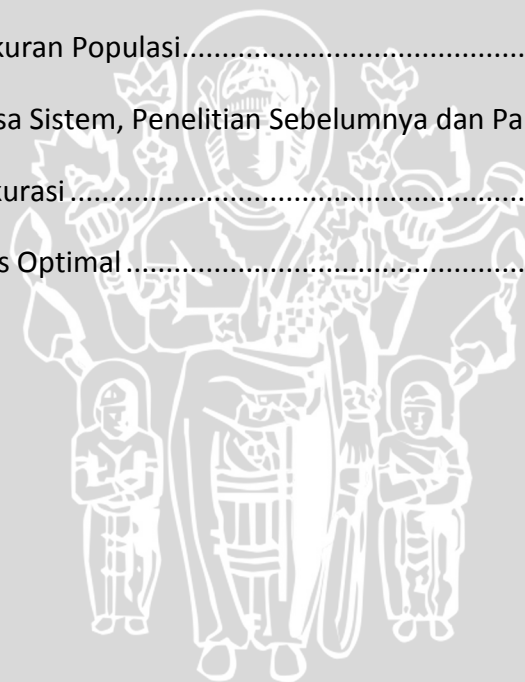
7.1	Kesimpulan	89
7.2	Saran	89
DAFTAR PUSTAKA.....		90
LAMPIRAN A DAFTAR NAMA PENYAKIT		92
LAMPIRAN B DAFTAR GEJALA.....		93
LAMPIRAN C DATA KASUS PENYAKIT SAPI		98
LAMPIRAN D DATA LATIH		102
LAMPIRAN E NILAI DENSITAS OPTIMAL		119



DAFTAR TABEL

Tabel 2.1 Contoh populasi untuk metode <i>Replacement Selection</i>	21
Tabel 2.2 Contoh <i>offspring</i> yang dihasilkan setelah proses <i>crossover</i> dan mutasi	21
Tabel 2.3 Contoh hasil seleksi dengan menggunakan metode <i>Replacement Selection</i>	22
Tabel 2.4 Contoh populasi untuk metode <i>Tournament Selection</i>	23
Tabel 2.5 Contoh hasil seleksi menggunakan metode <i>Tournament Selection</i>	23
Tabel 2.6 Contoh populasi untuk metode <i>Elitism Selection</i>	23
Tabel 2.7 Hasil perankingan semua individu	24
Tabel 2.8 Hasil seleksi menggunakan metode <i>Elitism Selection</i>	24
Tabel 3.1 Bentuk data yang diperoleh	26
Tabel 3.2 Tabel Pengujian Probabilitas <i>Crossover</i> dan Mutasi	29
Tabel 3.3 Tabel Pengujian Jumlah Generasi	29
Tabel 3.4 Tabel Pengujian Ukuran Populasi	30
Tabel 3.5 Tabel Pengujian Akurasi	30
Tabel 4.1 Daftar Kebutuhan Sistem	32
Tabel 4.2 Pembentukan Bilangan <i>Random</i> pada Populasi Awal	36
Tabel 4.3 Individu 1	48
Tabel 4.4 Aturan kombinasi untuk m_3	51
Tabel 4.5 Aturan kombinasi untuk m_5	52
Tabel 4.6 Hasil Akurasi Individu 1	53
Tabel 4.7 Hasil <i>Fitness</i> 5 Individu	53
Tabel 4.8 Tabel Alpha	54
Tabel 4. 9 Offspring Hasil <i>Crossover</i> Individu 1 dan Individu 2	55

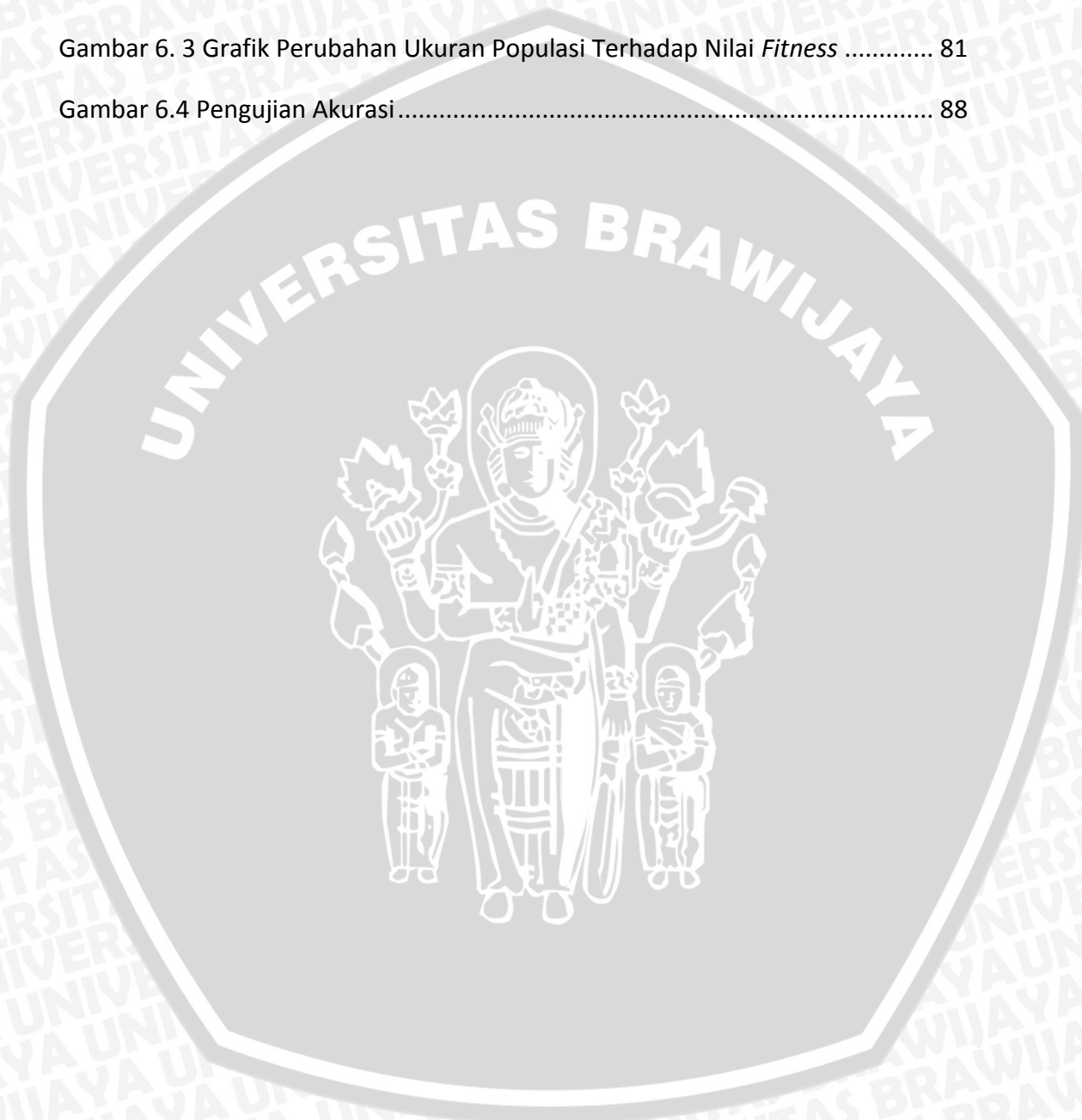
Tabel 4.10 Offspring Hasil Mutasi Individu 3	57
Tabel 4.11 Himpunan Populasi yang Menjadi <i>Parent</i>	59
Tabel 4.12 <i>Offspring</i> Beserta <i>Parent</i> dan <i>Fitness</i> -nya.....	59
Tabel 4.13 Individu yang Masuk ke Generasi Berikutnya	59
Tabel 4.14 Individu pada Generasi Baru	60
Tabel 5.1 Spesifikasi Perangkat Keras	62
Tabel 5.2 Spesifikasi Perangkat Lunak	62
Tabel 6.1 Rata-rata <i>Fitness</i> di Kelima Pengujian	78
Tabel 6.2 Pengujian Jumlah Generasi	79
Tabel 6.3 Pengujian Ukuran Populasi.....	80
Tabel 6.4 Hasil Diagnosa Sistem, Penelitian Sebelumnya dan Pakar.....	85
Tabel 6.5 Pengujian Akurasi	87
Tabel 6.6 Nilai Densitas Optimal	82



DAFTAR GAMBAR

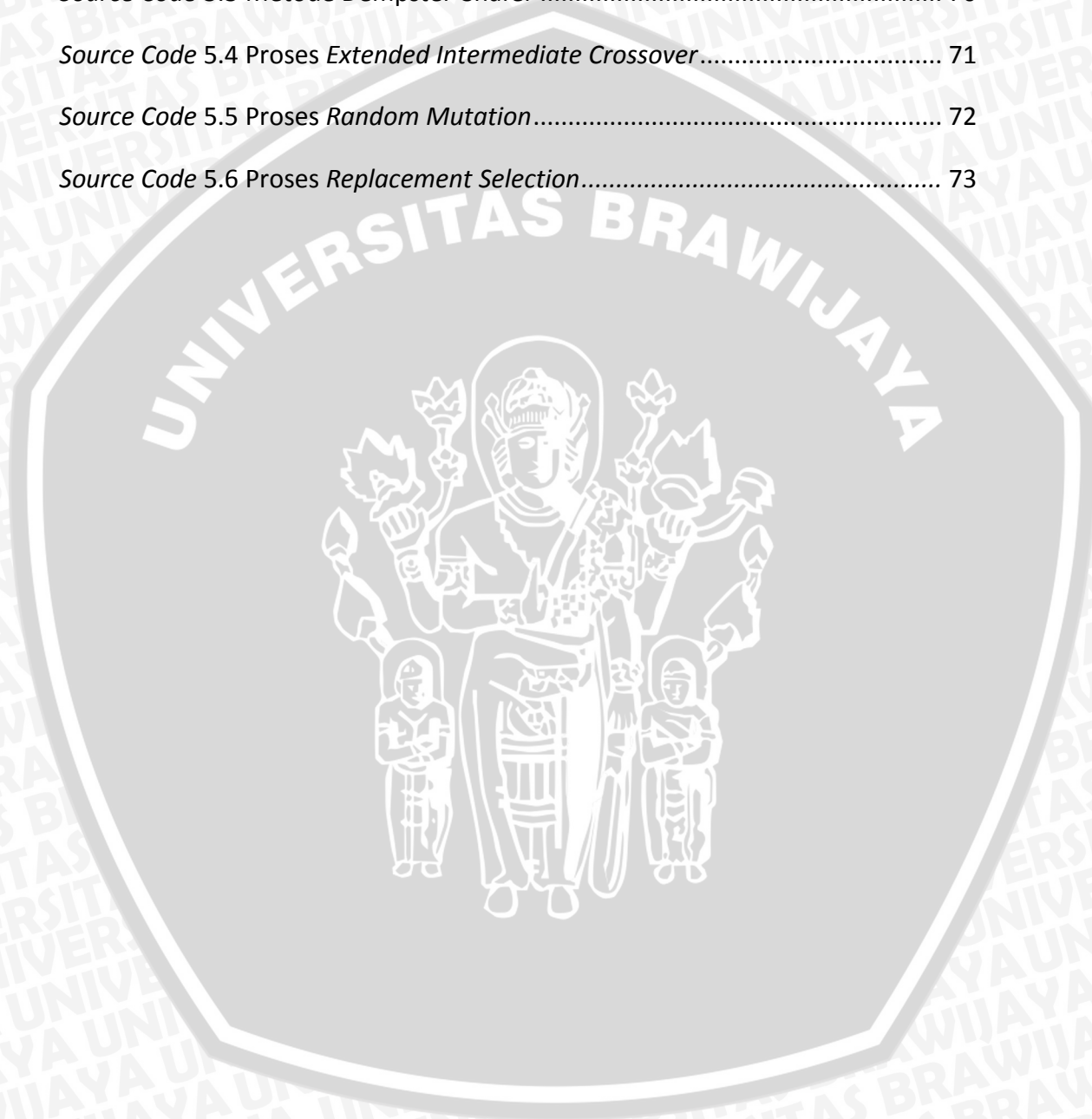
Gambar 2.1 Sapi yang Menderita <i>Malignant Catarrhal Fever</i> tahap awal.....	5
Gambar 2.2 Usus distal pada sapiyang menderita <i>Malignant Catarrhal Fever</i>	6
Gambar 2.3 Sapi yang Menderita Penyakit Mulut dan Kuku.....	6
Gambar 2.4 Sapi yang Menderita <i>Infectious Bovine Rhinotracheitis</i>	7
Gambar 2.5 Kerak yang pengering pada luka di kulit leher sapi	9
Gambar 2.6 Luka pada ambing sapi.....	10
Gambar 2.7 Sapi yang Menderita Penyakit <i>Brucellosis</i>	11
Gambar 2.8 Sapi yang Menderita Pink Eye	12
Gambar 2.9 Sapi yang Menderita Kelumpuhan.....	13
Gambar 3.1 Diagram Blok Metodologi Penelitian	25
Gambar 3.2 Gambar Diagram Blok Desain Arsitektur Sistem.....	28
Gambar 4.1 <i>Flowchart</i> Algoritma Genetika	33
Gambar 4.2 <i>Flowchart</i> Proses Inisialisasi Populasi Awal	35
Gambar 4.3 <i>Flowchart</i> Proses Evaluasi.....	39
Gambar 4.4 <i>Flowchart</i> Perhitungan Dempster Shafer.....	40
Gambar 4.5 <i>Flowchart</i> Proses Tukar Silang (<i>Crossover</i>)	43
Gambar 4.6 <i>Flowchart</i> Proses Mutasi.....	44
Gambar 4.7 <i>Flowchart</i> Proses Seleksi	45
Gambar 4.8 Antarmuka Proses Algoritma Genetika.....	60
Gambar 4.9 Antarmuka Nilai Densitas Optimal	61
Gambar 4.10 Antarmuka Hasil Akurasi	61
Gambar 5.1 Tampilan Proses Algoritma Genetika.....	74
Gambar 5.2 Tampilan Nilai Densitas Optimal.....	75

Gambar 5.3 Tampilan Hasil Akurasi	76
Gambar 6.1 Grafik Pengujian Nilai Probabilitas <i>Crossover</i> dan Probabilitas Mutasi Terhadap Nilai <i>Fitness</i>	78
Gambar 6.2 Grafik Pengujian Jumlah Generasi Terhadap Nilai <i>Fitness</i>	80
Gambar 6. 3 Grafik Perubahan Ukuran Populasi Terhadap Nilai <i>Fitness</i>	81
Gambar 6.4 Pengujian Akurasi	88



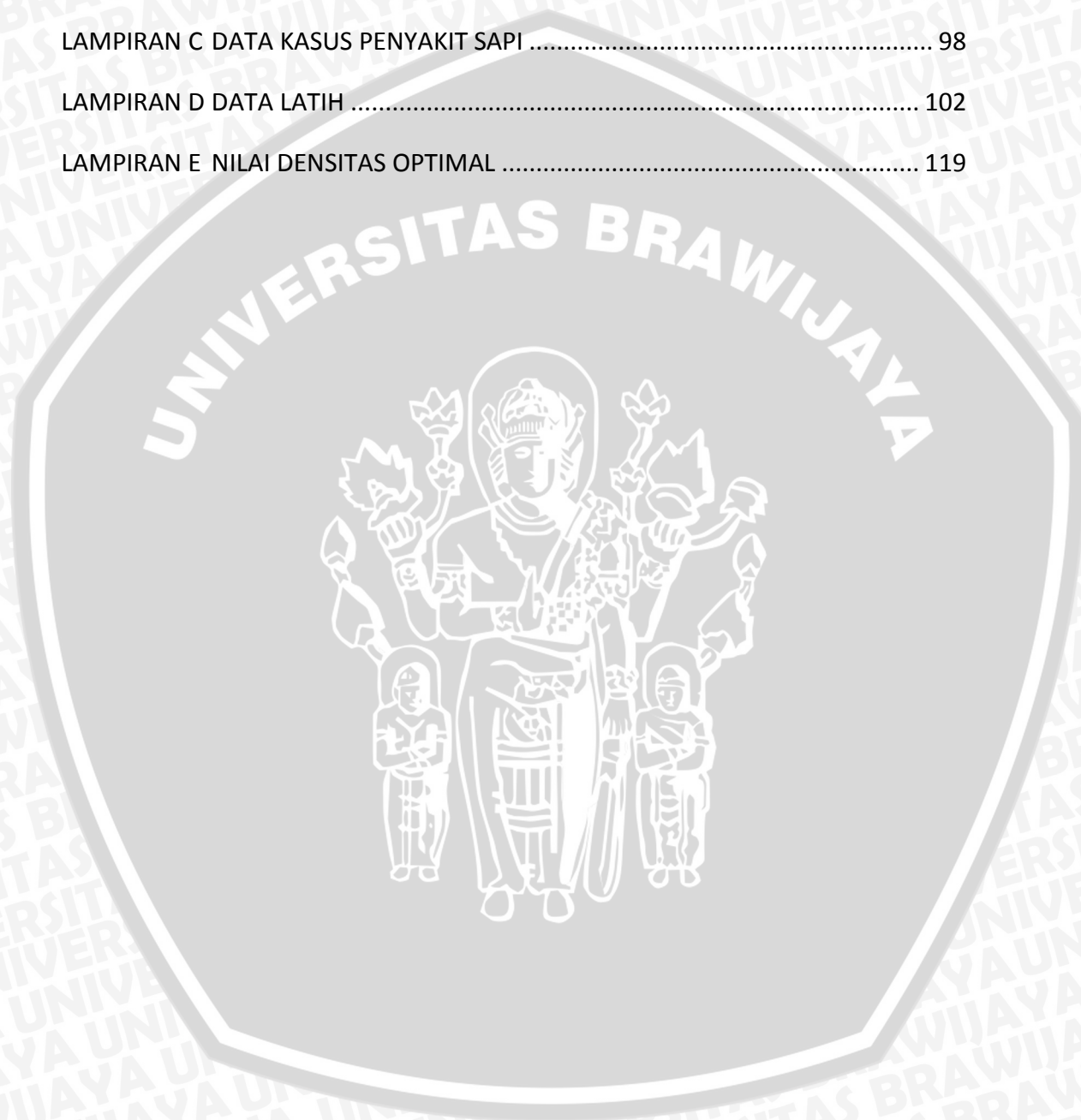
DAFTAR SOURCE CODE

Source Code 5.1 Inisialisasi Populasi Awal	64
Source Code 5.2 Perhitungan <i>Fitenss</i>	65
Source Code 5.3 Metode Dempster-Shafer	70
Source Code 5.4 Proses <i>Extended Intermediate Crossover</i>	71
Source Code 5.5 Proses <i>Random Mutation</i>	72
Source Code 5.6 Proses <i>Replacement Selection</i>	73



DAFTAR LAMPIRAN

LAMPIRAN A DAFTAR NAMA PENYAKIT	92
LAMPIRAN B DAFTAR GEJALA.....	93
LAMPIRAN C DATA KASUS PENYAKIT SAPI	98
LAMPIRAN D DATA LATIH	102
LAMPIRAN E NILAI DENSITAS OPTIMAL	119



BAB 1 PENDAHULUAN

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

1.1 Latar Belakang

Salah satu hewan ternak yang memiliki potensi ekonomi yang cukup tinggi adalah sapi. Peningkatan permintaan akan olahan ternak ini berbanding terbalik dengan produktivitasnya. Hal ini disebabkan dari berbagai faktor, salah satunya adalah pengetahuan terhadap penyakit sapi (Haloho dkk., 2013). Kurangnya pengetahuan menangani penyakit sapi membuat para peternak sapi sangat bergantung pada dokter hewan yang keberadaannya di Indonesia jarang, terutama di daerah pedesaan. Untuk menanganinya diperlukan pengembangan sistem pakar yang mengadopsi pengetahuan dari pakar atau dokter hewan.

Dalam pengembangan sistem pakar terdapat beberapa fakta, diantaranya terdapat beberapa kasus dengan gejala yang berbeda menghasilkan diagnosa yang sama. Fakta lainnya dengan gejala yang sama dan menghasilkan diagnosa yang sama, memiliki nilai setiap gejala yang jauh berbeda. Dalam hal ini diperlukan adanya pembobotan dalam sistem pakar. Salah satu metode sistem pakar yang memiliki pembobotan adalah Metode Dempster-Shafer. Metode ini memiliki pembobotan yang diberikan oleh pakar atau dokter hewan sesuai dengan pengetahuan yang dimilikinya.

Penelitian mengenai sistem pakar pendeteksian penyakit sapi menggunakan metode Dempster-Shafer sebelumnya sudah dilakukan oleh Mustikadewi Prihastuti (2014). Pada penelitian ini dokter hewan memberikan nilai densitas sebagai bobot yang digunakan pada perhitungan dalam mendiagnosa penyakit. Bobot yang diberikan oleh dokter pada penelitian ini masih menghasilkan nilai akurasi yang belum maksimal. Hasil perhitungan akurasi yang diperoleh sebesar 88,89%, dan ketidakakuratan sebesar 11,11% (Prihastuti, 2014). Hal ini disebabkan oleh beberapa faktor, salah satunya yaitu tidak mudah bagi pakar atau dokter hewan dalam memberikan nilai densitas. Dalam menentukan nilai densitas timbul ketidakkonsistenan yang dapat menyebabkan nilai densitas kurang optimal. Untuk mendapatkan nilai densitas yang konsisten dan optimal, maka diperlukan optimasi terhadap nilai densitas.

Ada beberapa metode yang dapat digunakan dalam optimasi, yaitu algoritma genetika (Anggariawan, 2014) (Kusumoputro and Irwanto, 2002), Ant Colony Algorithm (Zhao and Li, 2007), dan lain sebagainya. Dari beberapa metode optimasi yang ada, penelitian ini menggunakan metode algoritma genetika karena pada beberapa penelitian mampu meningkatkan akurasi setelah melakukan optimasi menggunakan algoritma genetika. Salah satu penelitian tersebut yaitu penelitian yang dilakukan Hendry Anggariawan (2014). Penelitiannya ini menerapkan algoritma genetika dalam mengoptimasi fungsi keanggotaan *fuzzy*

inference system model sugeno yang digunakan dalam menghitung angka metabolisme basal (AMB). Hasil pengujian akurasi pada penelitian ini mencapai 94%, dimana pada penelitian sebelumnya perhitungan AMB menggunakan Metode Inferensi Sugeno menghasilkan akurasi 75%. Penelitian lainnya dilakukan oleh Benyamin Kusumoputro dan Ponix Irwanto (2002). Penelitian ini menerapkan algoritma genetika untuk mengoptimasi jaringan syaraf buatan-fuzzy pada sistem penciuman elektronik. Hasil yang didapat dari penelitian ini yaitu derajat pengenalan sistem pada struktur jaringan yang telah dioptimasi meningkat menjadi 85,2%, dimana sebelum dioptimasi didapatkan 70,4%.

Algoritma genetika merupakan teknik pencarian heuristik yang menggunakan gagasan proses evolusi alam dan genetik. Algoritma ini memanfaatkan proses-proses seleksi alam yang juga dikenal dengan nama proses evolusi. Konsep algoritma ini dirancang untuk menirukan proses-proses yang diperlukan untuk evolusi yang dicetuskan oleh Chaerles Darwin. Dalam proses evolusi ini, setiap individu dalam menyesuaikan diri atau beradaptasi dengan lingkungan hidupnya akan terus-menerus mengalami perubahan gen. Algoritma ini telah banyak digunakan dalam memecahkan berbagai masalah diantaranya memecahkan masalah dan pemodelan teknik, bisnis, dan hiburan, termasuk optimasi (Sutojo et al., 2011).

Berdasarkan pemaparan latar belakang tersebut penulis mengambil judul “Implementasi algoritma genetika Untuk Optimasi Nilai Densitas Metode Dempster-Shafer pada Deteksi Penyakit Sapi”. Dengan digunakannya algoritma genetika diharapkan dapat memberikan nilai densitas pada metode Dempster-Shafer yang optimum.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan, maka dapat dirumuskan beberapa permasalahan, diantaranya sebagai berikut.

1. Bagaimana implementasi Algoritma Genetika pada Optimasi Nilai Densitas Metode Dempster-Shafer pada Deteksi Penyakit Sapi?
2. Berapa nilai probabilitas *crossover*, probabilitas mutasi, jumlah generasi dan jumlah populasi agar mendapatkan nilai densitas yang optimal?
3. Bagaimana hasil pengujian akurasi yang dihasilkan oleh nilai densitas yang optimal?

1.3 Tujuan Penelitian

Tujuan dari penelitian ini adalah sebagai berikut:

1. Mengimplementasikan algoritma genetika untuk mengoptimasi nilai densitas metode Dempster-Shafer pada Deteksi penyakit sapi.
2. Mengetahui nilai probabilitas *crossover*, probabilitas mutasi, jumlah generasi dan jumlah populasi agar mendapatkan nilai densitas yang optimal pada sistem optimasi nilai densitas metode Dempster-Shafer pada deteksi penyakit sapi menggunakan metode algoritma genetika.
3. Mengetahui akurasi yang dihasilkan oleh nilai densitas yang optimal.

1.4 Manfaat Penelitian

Manfaat yang diharapkan dari penulisan skripsi ini yaitu dapat memberikan rekomendasi nilai densitas pada metode Dempster-Shafer sehingga menghasilkan akurasi yang lebih tinggi dalam mendeteksi penyakit sapi.

1.5 Batasan Masalah

Penelitian ini dibatasi dalam hal, diantaranya sebagai berikut:

1. Penyakit sapi yang digunakan dalam penelitian ini adalah penyakit yang disebabkan oleh bakteri, virus, jamur, parasit, gangguan metabolisme, gangguan pencernaan, dan gangguan reproduksi.
2. Data uji yang digunakan adalah data yang diambil dari penelitian sebelumnya yaitu penelitian yang dilakukan oleh Mustikadewi Prihastuti (2014).
3. Metode *crossover* yang digunakan adalah *Extended Intermediate Crossover*, metode mutasi yang digunakan adalah *Random Mutation*, metode seleksi yang digunakan adalah *Replacement Selection*.

1.6 Sistematika Penulisan

Sistematika penulisan dalam skripsi ini sebagai berikut.

BAB I Pendahuluan

Bab ini menjelaskan latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian dan sistematika penelitian.

BAB II Tinjauan Pustaka

Pada bab ini akan diuraikan dasar teori yang berhubungan dengan penelitian, serta hasil penelitian terdahulu yang berhubungan dengan penelitian.

BAB III Metodologi Penelitian

Dalam bab ini akan membahas metodologi yang digunakan dalam penelitian, analisis kebutuhan dari dan perancangan sistem.

BAB IV Implementasi

Bab ini membahas tentang hasil dari analisis kebutuhan dan perancangan sistem.

BAB V Pengujian dan Analisi

Pada bab ini akan dijelaskan tentang pengujian dan analisis terhadap sistem yang telah terealisasi.

BAB VI Penutup

Dalam bab ini akan memuat kesimpulan yang diperoleh dari pembuatan dan pengujian sistem, serta saran terhadap penelitian ini.

BAB 2 LANDASAN KEPUSTAKAAN

Bab ini akan membahas mengenai landasan kepastakaan yang digunakan dalam penulisan skripsi “Implementasi Algoritma Genetika Untuk Optimasi Nilai Densitas Metode Dempster-Shafer pada Deteksi Penyakit Sapi”.

2.1 Kajian Pustaka

Pada kajian pustaka akan dijelaskan tentang penelitian-penelitian yang mengimplementasikan algoritma genetika untuk optimasi. Penelitian-penelitian tersebut dilakukan oleh Benyamin Kusumoputro dan Ponix Irwanto (2002), Hendry Anggariawan (2014), Wiharto dkk. (2013), dan Youngjoo Lee dan Jongjin Lee (2015).

Benyamin Kusumoputro dan Ponix Irwanto pada tahun 2002 melakukan optimasi jaringan syaraf tiruan propagasi balik menggunakan algoritma genetika. Optimasi ini bertujuan untuk memperkecil jumlah neuron dalam *hidden layer* atau jumlah bobot. Penelitian ini menghasilkan peningkatan derajat pengenalan sistem dari 70,4 % menjadi 85,2 % setelah dioptimasi (Kusumoputro dan Irwanto, 2002).

Penelitian Hendry Anggariawan pada tahun 2014 menggunakan algoritma genetika dalam mengoptimasi fungsi keanggotaan *Fuzzy Inference System Model Sugeno*. Fungsi keanggotaan ini digunakan pada perhitungan Angka Metabolisme Basal (AMB). Hasil akurasi yang didapat dari penelitian ini mencapai 94% dimana sebelum dioptimasi sebesar 75% (Anggariawan, 2014).

Penelitian Wiharto, Y.S. Palgunadi, dan Muh Aziz Nugroho pada tahun 2013 menggunakan algoritma genetika untuk mengatasi permasalahan pada Jaringan Syaraf Tiruan. Algoritma genetika digunakan untuk menyelesaikan permasalahan yang terjadi dalam pelatihan menggunakan *Least Means Square* (LMS). Rata-rata akurasi yang didapat sebelum dioptimasi yaitu 63,75%, sedangkan setelah dioptimasi menggunakan algoritma genetika yaitu 81,73% (Wiharto dkk., 2013).

Penelitian yang dilakukan Youngjoo Lee dan Jeongjin Lee pada tahun 2015 menggunakan algoritma genetika untuk mengoptimasi pohon biner untuk SVM *multiclass*. Rata-rata akurasi yang didapat setelah optimasi yaitu 86,83%. Penelitian ini juga membandingkan algoritma genetika dengan metode lain seperti One-Againts-All (OAA), One-Againts-One (OAO), Direct Acyclic Graph (DAG), k-means, dan OneAtOnce. Algoritma genetika di 11 dari 18 dataset yang diberikan memperoleh akurasi tertinggi dibandingkan dengan metode lain. Untuk metode OAA dan OneAtOnce hanya di 4 dataset, metode OAO di 2 dataset, sedangkan akurasi DAG dan k-means tidak mendapat posisi terbaik di semua dataset (Lee dan Lee, 2015).

Penelitian ini terfokus pada optimasi nilai densitas pada metode *Dempster-Shafer* dengan menggunakan algoritma genetika dalam mendeteksi penyakit sapi. Referensi yang digunakan berasal dari penelitian sebelumnya yaitu yang

dilakukan Mustikadewi Prihastuti (2014) dimana menggunakan metode *Dempster-Shafer* untuk mendeteksi penyakit sapi.

2.2 Penyakit Sapi

Penyakit sapi dapat digolongkan kedalam beberapa jenis penyakit sesuai dengan penyebabnya dan jenis gangguannya. Penyakit sapi ada yang disebabkan oleh virus, bakteri, jamur, dan parasit. Selain itu penyakit sapi dapat disebabkan karena gangguan metabolisme, gangguan pencernaan dan gangguan reproduksi yaitu :

2.2.1 Penyakit yang disebabkan oleh virus (Penyakit Viral)

Ada beberapa penyakit sapi yang sebagian besar disebabkan oleh bakteri dan virus seperti Malignant Catarrhal Fever, Penyakit Mulut dan Kuku, Bovine Ephemeral Fever, Bovine Viral Diarrhea, Infectious Bovine Rhinotracheitis / Infectious Pustular Vulvovaginitis, Penularan Virus Parainfluenza-3, Infeksi "Respiratory Syncytial Virus", Cacar Sapi (*cowpox*), Paravaksinaria pada Sapi, Stomatitis Papulosa pada Sapi, Lekosis Enzootis pada Sapi, Infeksi Rotavirus dan *Bovine Herpes Virus*. Penjelasan penyakit yang lebih lanjut adalah sebagai berikut:

1. *Malignant Catarrhal Fever*

Malignant catarrhal fever dikenal dengan istilah penyakit ingusan, *Coryza Gangraenosa Bovum*, *Bösartiges Katarrhalfieber der Rinder*, *Snotsiekte*. Penyakit ini merupakan penyakit akut pada sapi dan kerbau yang ditandai dengan demam tinggi, radang luas pada mukosa bagian muka dari alat pernafasan dan alat digesti. Di samping itu terlihat gejala-gejala pada susunan saraf pusat dan kekeruhan kornea yang biasanya dimulai pada periferi tetapi meluas secara sentripetal dan mengakibatkan hewan buta. Pembesaran kelenjar getah bening juga sering terlihat (Ressang, 1986). Sapi yang menderita *Malignant Catarrhal Fever* tahap awal mengalami *opacity* kornea, konjungtivitis dan kemerahan dari kelopak mata seperti yang ditunjukkan pada Gambar 2.1 dan Gambar 2.2.



Gambar 2.1 Sapi yang Menderita *Malignant Catarrhal Fever* tahap awal
Sumber : (Herenda, 1994)



Gambar 2.2 Usus distal pada sapiyang menderita *Malignant Catarrhal Fever*
Sumber : (Herenda, 1994)

2. Penyakit Mulut dan Kuku

Penyakit mulut dan kuku dikenal dengan istilah lain seperti *Foot and Mouth Disease*, *La Fievre Aphteuse*, *Aphtae epizooticae*, *Maul und Klauenseuche*. Penyakit ini merupakan penyakit akut yang disertai demam dan menyerang hewan-hewan berkuku belah. Secara klinis terlihat pembentukan lepuh pada lidah, dinding tongga mulut dan pada tungkai terutama pada tepi kuku. Lepuh juga di dapatkan pada kulit puting dan kelenjar susu, dan juga pada epitel rumen. Pada hewan yang masih muda sering terlihat perubahan-perubahan degenerasi pada jaringan otot, seperti pada otot jantung yang akan menyebabkan hewan mati mendadak (Ressang, 1986).



Gambar 2.3 Sapi yang Menderita Penyakit Mulut dan Kuku
Sumber : (Herenda, 1994)

3. *Bovine Ephemeral Fever* (Demam 3 Hari)

Bovine Ephemeral Fever merupakan penyakit jenis ringan pada sapi yang ditandai dengan demam dan kepincangan. Penyakit ini berlangsung singkat, biasanya berakhir dalam waktu tidak lebih dari 3 hari, dengan morbiditas tinggi tetapi mortalitas rendah. Penyakit ini dapat menimbulkan gangguan terhadap produksi susu pada sapi perah dan sapi yang digunakan untuk membajak. Agens penyebabnya ialah rhabdovirus yang ditransmisikan oleh arthropoda (Ressang, 1986).

4. *Bovine Viral Diarrhoea* (Diare Ganas)

Bovine Viral Diarrhoea atau dikenal dengan istilah *Mucosal Disease*, yang disebabkan oleh virus bovine viral diarrhea (BVD), virus mucosal disease (MD) atau virus BVD-MD. Pada sapi, infeksi oleh virus BVD biasanya menyebabkan infeksi yang tidak tampak (*inapparent*), meskipun bentuk akut dan kronis penyakit ini telah diketahui. Apabila sudah terjangkit penyakit ini, hampir selalu menyebabkan kematian. Infeksi pada sapi bunting dapat mengakibatkan berbagai macam abnormalitas fetus meskipun tidak menimbulkan gangguan pada induknya. Infeksi pada sapi terjadi lewat kontak dan virus yang dikeluarkan dalam berbagai getah sekresi dan ekskresi (Ressang, 1986).

5. *Infectious Bovine Rhinotracheitis / Infectious Pustular Vulvovaginitis*

Infectious bovine rhinotracheitis (IBR) merupakan penyakit saluran pernafasan bagian muka pada sapi biasanya disertai suhu badan tinggi. Sapi sakit juga memperlihatkan konjunktivitis. *Infectious pustular vulvovaginitis* (IPV) yang juga dikenal dengan nama *exanthema coitale* atau balanopostitis menular pada sapi dan pemacet. Virus IBR/IPV dapat juga menyebabkan infeksi umum dan menyebabkan abortus (Ressang, 1986).



Gambar 2.4 Sapi yang Menderita *Infectious Bovine Rhinotracheitis*
Sumber : (Herenda, 1994)

6. Penularan Virus Parainfluenza-3 pada Sapi

Infeksi yang disebabkan oleh Paramyxovirus menimbulkan penyakit yang gejala-gejala klinisnya relative ringan, kecuali bila ada komplikasi yang disebabkan oleh agen infeksi lain seperti misalnya mikoplasma atau bakteri. Tempat predileksi virus ini adalah saluran pernafasan meskipun fetus sekali-kali ditularinya (Ressang, 1986).

7. Infeksi "*Respiratory Syncytial Virus*" pada Sapi

Infeksi yang disebabkan oleh virus tractus respiratorius ini menyebabkan pembentukan sel-sel sinsicia. Virus ini terkadang menjadi penyebab utama penyakit pernafasan pada sapi Virus saluran respiratorius dan sinsisial (RS) merupakan virus RNA yang berbentuk pleomorf dan mempunyai selubung. Gejala-gejala klinis yang utamanya antara lain rhinitis, faringitis dan batuk-batuk. Banyak bayi berumur kurang dari enam bulan mengalami bronkhitis,

bronchiolitis dan bronkhopnemoni yang disebabkan karena terinfeksi virus ini. Kavia dapat tertular secara intranasal dan intraperitoneal, jenis hewan laboratorium ini sering dipergunakan untuk pembuatan serum imun (Ressang, 1986).

8. Cacar Sapi (*cowpox*)

Cacar sapi atau *cowpox* merupakan penyakit kulit ambing dan puting susu pada sapi yang disertai pembentukan cacar. Pada tahun 1796 Jenner mempergunakan virus cacar sapi untuk memvaksinasi seorang anak yang menderita variola (*pox*). Ia memakai paterial seorang pemerah susu yang tertular setelah pemerah sapi yang terinfeksi. Agen yang dipergunakan untuk memvaksinasi manusia baru dapat didiferensiasi pada tahun 1950an dari virus *cowpox* asli. Virus cacar sapi asli ini ditemukan hanya secara sporadis (Ressang, 1986).

9. *Paravakcinia* pada Sapi

Paravakcinia yang juga dinamakan "*Pseudo cowpox*" merupakan penyakit kulit puting susu dan ambing pada sapi yang menyerupai penyakit cacar. Induk semang yang dapat terinfeksi virus paravakcinia adalah sapi. Virus ini bersifat pathogen untuk manusia. Infeksi ditandai dengan timbulnya warna kemerahan dan terdapat satu atau beberapa benjolan-benjolan kecil yang keras yang berwarna kelabu pada tangan (*milker's nodules*). Orang yang terinfeksi biasanya tidak menunjukkan gejala sakit seperti biasanya akan tetapi terjadi pembengkakan kelenjar limfe regional yang sekali-kali terlihat. Umumnya perubahan-perubahan kulit akan menghilang dengan sendirinya dalam beberapa minggu (Ressang, 1986).

10. *Stomatitis Papulosa* pada Sapi

Penyakit *Stomatitis Papulosa* yang juga dikenal dengan nama "*Bovine popular stomatitis*". Penyakit ini merupakan infeksi tenang pada selaput lendir mulut dan sekali-kali pada cermin hidung anak sapi dan sapi dewasa. Infeksi tersebar luas dan mungkin ditemukan di mana saja sapi dipelihara. Virus stomatitis papulose termasuk genus *Parapoxvirus*. Dan virus ini tidak dapat dibedakan dengan virus paravakcinia. Sapi yang terinfeksi dapat mengekskresi virus dengan air liur dalam jangka waktu lama. Setelah infeksi lokal pada selaput lendir mulut maka dalam beberapa hari hingga seminggu akan timbul bercak-bercak merah kelabu. Beberapa hari kemudian akan terbentuk plak berwarna merah coklat dan menonjol di atas jaringan sekitarnya. Penonjolan-penonjolan ini berdiameter sekitar 1 - 2 cm. Pada bagian tengahnya akan memperlihatkan nekrosa. Infeksi ini mengakibatkan kerusakan-kerusakan pada gusi, langit-langit, bibir lidah dan cermin hidung. Hewan yang terjangkit penyakit *Stomatitis Papulosa* biasanya tidak memperlihatkan gejala-gejala sakit. Infeksi sekunder dengan kuman-kuman dapat menyebabkan penyakit berlangsung berbulan-bulan. Pada hewan yang lemah karena sebab lain penyakit dapat berjangkit menahun (Ressang, 1986).

11. *Lekosis Enzoosis* pada Sapi

Lekosis enzoosis bovina (EBL) yang juga dikenal dengan istilah *Enzoosis Bovine Leucosis*, *Bovine Lymphosarcoma*, *Bovine Leukemia* merupakan penyakit ganas yang menyerang susunan kelenjar-kelenjar limfe. Penyakit ini mempunyai jangka inkubasi yang cukup lama. Gejala-gejala klinis penyakit ini yaitu lymphadenopathia, penambahan jumlah darah putih dan umumnya hanya dilihat pada hewan yang lebih dari 2 tahun umurnya. Pada anak sapi dan sapi berumur sampai 1,5 tahun dikenal juga bentuk lekosis yang sporadic (*leucosis sporadica juvenilis*) dan tidak bersifat menular. Pada bentuk lekosis juvenile ini, tidak didapatkan virus dan tidak terbentuk antibody terhadap virus EBL (Ressang, 1986).

12. Infeksi *Rotavirus* pada Sapi

Infeksi *Rotavirus* menyebabkan infeksi usus terutama pada anak sapi yang baru lahir. Penyakit yang ditimbulkannya dinamakan juga *Neonatal calf diarrhea* atau *Nebraska calf diarrhea*. *Rotavirus* menimbulkan diare pada anak sapi berumur beberapa hari hingga kira-kira 6 minggu. Anak sapi tiba-tiba jatuh sakit, memperlihatkan depresi, tidak minum dan tinja encer seperti air dan berwarna kuning hingga kuning hijau. Diare itu hanya berlangsung 12-24 jam dan hewan yang terkena menyembuh dalam beberapa hari bila tidak ada komplikasi oleh infeksi kuman-kuman (*E. coli*) (Ressang, 1986).

13. *Bovine Herpes Virus*

Penyakit *bovine herpes virus* yang sering disebut dengan IBR terbagi menjadi tiga sub tipe utama yaitu BHV-1.1, BHV-1.2 dan BHV-1.3. Masa inkubasi hanya dalam beberapa hari sampai satu minggu. Setelah masa inkubasi awal, mulai timbul beberapa gejala yaitu demam tinggi, batuk, gangguan pernafasan, leleran hidung, konjungtivitis dan aborsi (Syibli, 2014).



Gambar 2.5 Kerak yang pengering pada luka di kulit leher sapi
Sumber : (Herenda, 1994)



Gambar 2.6 Luka pada ambing sapi
Sumber : (Herenda, 1994)

2.2.2 Penyakit yang disebabkan oleh bakteri

Ada beberapa penyakit yang disebabkan oleh bakteri seperti Antraks, *Brucellosis*, *Septisemia Epizootica*, *Mastitis*, TBC, Radang Paha, *Pink Eye*, *Johne' Disease*, *Salmonellosis*, *Botulisme*, Radang Kuku dan *Vibriosis / Campylobacteriosis*.

1. Antraks

Penyakit antraks atau radang limpa disebabkan oleh bakteri *Bacillus anthracis*. Bakteri ini berbentuk panjang dan terbungkus kapsul. Gejala-gejala yang dialami sapi jika terserang penyakit ini adalah demam, gelisah, lemah, paha gemetar, nafsu makan hilang dan menyebabkan sapi ambruk. Gejala awal yang dialami adalah sulit buang kotoran, diare, kotoran bercampur air atau darah. Sapi sangat cepat mengalami antraks akut hanya dalam 1-3 hari saja (Syibli, 2014).

2. *Brucellosis*

Penyakit *brucellosis* yang sering disebut dengan kluron menular atau keguguran menular. Penyakit ini dapat menyebabkan rusaknya alat reproduksi pada sapi betina, dan dapat menakibatkan sapi mengalami kemandulan. Gejala yang dialami sapi yang terserang penyakit ini yaitu radang alat kelamin. Beberapa sapi yang terjangkit penyakit ini masih dapat melahirkan, namun anak sapi yang dilahirkan memiliki kondisi yang tidak sehat (Syibli, 2014).



Gambar 2.7 Sapi yang Menderita Penyakit *Brucellosis*
Sumber : (Herenda, 1994)

3. *Septisemia Epizootica* (Penyakit Sapi Ngorok)

Penyakit *Septisemia Epizootica* atau yang dikenal dengan penyakit sapi ngorok merupakan penyakit yang disebabkan oleh bakteri *Pasturella multocida*. Penyakit ini menyebar melalui makanan dan minuman yang tercemar bakteri. Sapi yang terjangkit bakteri *Pasturella multocida* mengalami peradangan sehingga mengalami kesulitan bernafas dan terdengar seperti sedang mengorok. Penyebab lain dari penyakit ini adalah kebersihan kandang atau sistem sanitasi kandang yang tidak dijaga dengan baik (Syibli, 2014).

4. *Mastitis*

Penyakit mastitis atau radang ambing merupakan penyakit yang sering menyerang ternak sapi dan menyebabkan produksi susu menurun. Sapi yang terserang penyakit ini akan mengalami pembengkakan ambing dalam skala tidak normal, radang berupa kebengkakan, panas dalam rabaan, rasa sakit, ambing yang kemerahan dan terganggunya fungsi ambing. Selain itu air susu 'pecah', bercampur endapan, konsistensi air susu lebih encer dan warnanya menjadi agak kebiruan, atau putih yang pucat (Syibli, 2014).

5. TBC (*Tuberculosis*)

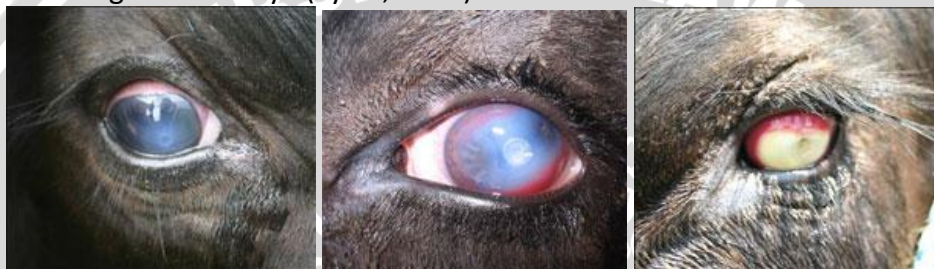
Penyakit *Tuberculosis* atau TBC merupakan penyakit radang paru-paru dan sangat berbahaya, karena ternak dapat menularkannya ke manusia atau sebaliknya. Gejala awal dari penyakit ini adalah pembentukan tuberkel pada alat-alat yang digunakan sebagai tempat kuman-kuman TBC berkembangbiak. Sapi yang terjangkit kuman TBC akan mengalami hilangnya nafsu makan, badan menjadi kurus, bulu kusam, kering dan tidak mengkilat, sulit bernafas, batuk-batuk dan mengeluarkan lendir campur darah (Syibli, 2014).

6. Radang Paha (*Backleg*)

Penyakit radang paha menyebabkan nekrosis otot kaki pada sapi muda, berwarna merah kehitaman dengan gelembung. Penyakit ini mengakibatkan infeksi local akut, keracunan darah, dan dapat menyebabkan kematian. Penyakit ini dapat terjadi saat cuaca panas, lembap, atau setelah tiba-tiba cuaca dingin (Syibli, 2014).

7. *Pink Eye*

Penyakit *pink eye* dikenal dengan istilah *infectious bovine keratoconjunctivitis* (IBK). Penyakit ini mudah menular dan dapat menyebabkan radang/inflamasi pada kornea hingga menyebabkan kebutaan. Penyakit ini bersifat endemic, karena sapi yang telah terinfeksi penyakit ini dapat terjangkit kembali setiap tahunnya. Namun penyakit ini termasuk penyakit yang dapat sembuh dengan sendirinya (Syibli, 2014).



Gambar 2.8 Sapi yang Menderita Pink Eye
Sumber : (Whittier dkk., 2009)

8. *Johne' Disease*

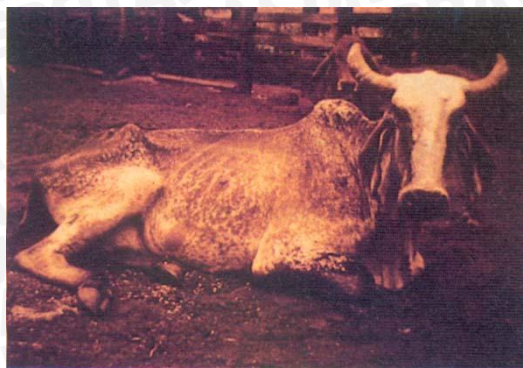
Gejala penyakit *johne' disease* yang ditunjukkan pada sapi jarang dapat ditemukan sebelum sapi berumur dua tahun atau lebih, meskipun infeksi terjadi sejak anak sapi dilahirkan. Pada stadium akhir gejala yang ditunjukkan dapat berupa diare kronis dan kehilangan berat badan (Syibli, 2014).

9. *Salmonellosis*

Kasus penyakit *Salmonella typhimurium* yang paling banyak ditemukan yaitu diare akut. Sapi dewasa yang telah terinfeksi dapat sebagai *carrier* bagi sapi-sapi lainnya. Gejala yang ditunjukkan jika terinfeksi penyakit ini demam, lesu, kurang nafsu makan, produksi susu menurun pada sapi perah, diare bercampur darah dan abortus pada sapi bunting (Syibli, 2014).

10. *Botulisme*

Penyakit *botulisme* dapat melumpuhkan sapi, dimana sapi akan menderita kelumpuhan progresif dan gagal bernafas karena kelumpuhan otot-otot pernafasan, sapi cenderung memiliki gaya berjalan kaku dan mengeluarkan air liur. Sapi yang sudah terjangkit biasanya ditemukan duduk, tidak mampu berdiri dan nafasnya berat. Untuk memudahkan sapi bernafas, sapi akan membenkokkan kakinya kebelakang seperti posisi kaki katak (Tjahajati dan Husniyati, 2012).



Gambar 2.9 Sapi yang Menderita Kelumpuhan

Sumber : (Herenda, 1994)

11. *Radang Kuku*

Penyakit radang kuku atau sering disebut kuku busuk kondisi kandang yang basah, lembab dan kotor dapat menyebabkan sapi terserang penyakit ini. Gejala yang sering dialami oleh sapi yaitu sekitar celah kuku bengkak dan mengeluarkan cairan putih keruh, kulit kuku mengelupas, kuku membusuk berwarna merah kehitman dan kuku sapi busuk. Selain itu kadang-kadang tumbuh benjolan yang menimbulkan rasa sakit hingga akhirnya lumpuh (Tjahajati dan Husniyati, 2012).

12. *Vibriosis / Campylobacteriosis*

Penyakit *Vibriosis* atau dikenal juga dengan istilah *Campylobacteriosis* merupakan penyakit yang disebabkan oleh bakteri *campylobacteriosis*. Bakteri ini yang merupakan penyebab utama kegagalan reproduksi pada sapi yang disebarkan melalui perkawinan dan ditandaioleh fertilitas. Biasanya kematian dini embrio dan abortus terjadi pada bulan keempat sampai akhir masa kehamilan pada sapi (Tjahajati dan Husniyati, 2012).

2.2.3 Penyakit yang disebabkan oleh jamur

Penyakit sapi yang disebabkan oleh fungi atau jamur yaitu *Ringworm* dan *Aspergilosis*.

1. *Ringworm*

Penyakit *ringworm* merupakan penyakit yang menyerang permukaan kulit dan disebabkan oleh jamur. Penyakit ini adalah salah satu penyakit menular. Meskipun tingkat kematian sapi akibat penyakit ini rendah, tetapi kerugian ekonomi yang ditimbulkan sangat besar. Penyakit ini menyebabkan mutu kulit menurun dan berat badan menurun karena sapi terlalu gelisah (Syibli, 2014).

2. *Aspergilosis*

Penyakit *aspergilosis* merupakan radang pada ari-ari yang mengakibatkan keguguran. Keguguran ini sering dianggap sebagai penyebab dari penyakit lain, seperti *brucellosis*, *vibriosis* atau *leptospirosis* (Syibli, 2014).

2.2.4 Penyakit yang disebabkan oleh parasit

Beberapa penyakit sapi yang disebabkan oleh parasit yaitu Surra, Cacingan, Kudisan Menular dan Belatungan.

1. Surra

Penyakit surra atau sering disebut dengan sapi gila merupakan penyakit yang menyebabkan sapi mengamuk, berputar-putar tanpa arah dan susah dikendalikan. Penyakit ini disebabkan oleh parasit darah yang menyerang otak dan saraf, kemudian terjadi inkordinasi ke seluruh tubuh. Karena membahayakan pemilik sapi biasanya membiarkan sapi terlepas, akibatnya sering kali sapi ditemukan sudah tidak bernyawa (Syibli, 2014).

2. Cacingan

Penyakit cacingan biasanya menyerang sapi yang dipelihara secara tradisional. Variasi ukuran sapi yang terjangkit penyakit ini tergantung pada jenis cacing, jumlah cacing yang menyerang, umur sapi yang terserang dan kondisi pakan (Syibli, 2014).

3. *Demodectosis* (Kudisan Menular)

Penyakit kudisan ini ditularkan melalui perantara lalat rumah. Penyakit ini bersifat zoonosis dan dapat menular ke manusia. Awalnya lalat hinggap di dekat mata dan mengenai air mata sapi yang terinfeksi, kemudian lalat tersebut akan menginfeksi sapi lain. Penularan penyakit ini melalui kontak langsung antara sapi yang terjangkit dengan sapi yang sehat atau melalui peralatan kandang yang tercemar oleh tungai (Syibli, 2014).

4. Belatungan

Belatungan merupakan invasi belatung pada jaringan tubuh dan menyebabkan kerusakan pada jaringan tersebut. Myasis berawal dari trauma terbuka dan didukung oleh lingkungan yang kotor sehingga mempermudah lalat untuk kontak langsung dengan luka (Syibli, 2014).

2.2.5 Penyakit karena gangguan metabolisme

Penyakit yang diakibatkan karena adanya gangguan metabolisme yaitu kelumpuhan dan sindrom sapi ambruk.

1. Kelumpuhan

Kelumpuhan merupakan salah satu penyakit metabolik yang sering dialami oleh sapi perah. Sapi perah biasanya terserang penyakit ini saat 48-72 jam setelah melahirkan, pada masa laktasi ketiga atau pada sapi yang sudah tua dan produksi susunya tinggi (Tjahajati dan Husniyati, 2012).

2. Sindrom Sapi Ambruk

Sindrom sapi ambruk merupakan salah satu gangguan fungsional yang bersifat kompleks. Penyakit ini sering ditemukan pada sapi perah. Gejala yang

dialami sapi adalah *milk fever*, sapi tidak mampu bangun meskipun sudah diberikan pengobatan (Tjahajati dan Husniyati, 2012).

2.2.6 Penyakit karena gangguan pencernaan

Penyakit yang disebabkan oleh adanya gangguan pencernaan antara lain Indigesti Sederhana, Kembung Sapi dan Asidosis Rumen.

1. Indigesti Sederhana

Indigesti sederhana adalah sindrom gangguan pencernaan yang bersumber dari reticulum atau rumen. Gejala penyakit ini ditandai dengan penurunan atau hilangnya gerak rumen. Sindrom ini biasanya berlangsung dalam beberapa jam hingga dua hari. Sindrom ini merupakan gejala awal dari penyakit organik lainnya (Tjahajati dan Husniyati, 2012).

2. Kembung Sapi

Kembung pada sapi merupakan suatu keadaan rumen yang membesar secara tidak normal. Hal ini dikarenakan akumulasi kelebihan gas yang tidak keluar. Rumen merupakan perut yang paling besar dengan kapasitas 100-230 L dan terletak di sebelah kiri. Oleh karena itu, perut sebelah kiri akan tampak menonjol, karena udara pada rumen yang mengembang mendorong ke segala arah (Tjahajati dan Husniyati, 2012).

3. Asidosis Rumen

Asidosis rumen disebabkan karena sapi mengonsumsi karbohidrat yang cepat difermentasi dalam alat pencernaan sapi dalam jumlah yang banyak. Hal ini mengakibatkan akumulasi nonfisiologi dan penurunan pH (Tjahajati dan Husniyati, 2012).

2.2.7 Penyakit karena gangguan reproduksi

Penyakit yang akibat adanya gangguan reproduksi pada sapi yaitu Retensi Plasenta dan Distokia.

1. Retensi Plasenta

Retensi plasenta sapi merupakan suatu kegagalan penularan plasenta fetalis dari alat kelamin induk ke anaknya. Penularan ini berlangsung dalam waktu 1-2 jam setelah kelahiran. Penyakit ini terjadi pada kasus abortus setelah bulan kelima. Gejala yang dialami sapi yang terjangkit yaitu kesulitan melahirkan, rahim berputar, rahim berisi cairan, kekurangan kalsium, eksitasi waktu melahirkan, kelahiran yang dipaksakan, kegemukan dan defisiensi vitamin A, E dan selenium (Tjahajati dan Husniyati, 2012).

2. Distokia

Distokia atau kesulitan melahirkan adalah suatu kondisi stadium kelahiran pertama dan kedua yang lebih lama. Hal ini membuat induk sulit dan tidak mungkin lagi untuk mengeluarkan fetus (Tjahajati dan Husniyati, 2012).

2.3 Dempster-Shafer

Teori Dempster-Shafer pertama kali diperkenalkan oleh Dempster dan kemudian diperpanjang oleh Shafer, tetapi jenis penalaran penggunaan teori sudah ditemukan pada abad ketujuh belas. Teori ini sebenarnya merupakan perpanjangan pemodelan ketidakpastian probabilistik klasik. Nama teori Dempster-Shafer diciptakan oleh JA Barnett dalam sebuah artikel yang menandai masuknya fungsi kepercayaan ke dalam literatur kecerdasan buatan (Maseleno dan Hasan, 2014).

Dalam ruang diskrit, teori Dempster-Shafer dapat diartikan sebagai generalisasi dari teori probabilitas. Dalam teori probabilitas tradisional, bukti dikaitkan dengan satu kemungkinan. Dalam teori ini, fakta dapat diartikan dengan beberapa peristiwa yang mungkin atau rangkaian peristiwa. Dimana fakta-fakta yang cukup digunakan untuk memungkinkan penetapan probabilitas. Model ini dirancang untuk mengatasi berbagai tingkat presisi mengenai informasi dan tidak ada asumsi lebih lanjut yang diperlukan untuk mewakili informasi (Senz dan Ferson, 2002).

Teori Dempster-Shafer memiliki tiga fungsi penting, yaitu fungsi *basic probability assignment* (bpa atau m), fungsi Kepercayaan (Bel), dan fungsi Plausibility (Pl) (Senz dan Ferson, 2002).

Basic probability assignment (bpa) diwakili oleh m , mendefinisikan pemetaan himpunan kuasa untuk interval antara 0 dan 1, dimana bpa himpunan kosong adalah 0 dan penjumlahan dari semua bpa himpunan bagian dari himpunan kuasa adalah 1. Penjelasan dari bpa dapat diwakili dengan tiga persamaan berikut (Senz dan Ferson, 2002):

$$m: P(X) \rightarrow [0,1] \quad (2.1)$$

$$m(\emptyset) = 0 \quad (2.2)$$

$$\sum_{A \in P(X)} m(A) = 1 \quad (2.3)$$

Dimana $P(X)$ merupakan himpunan kuasa X , $\emptyset$ adalah himpunan kosong, dan A adalah satu himpunan di himpunan kuasa, $A \in P(X)$. Dari *basic probability assignment* dapat mendefinisikan batas atas dan bawah dari interval. Interval ini dibatasi oleh dua ukuran yang bersifat kontinyu nonadditive yang disebut *Belief* (Kepercayaan) dan *Plausibility*. Batas bawah atau *belief* untuk satu himpunan didefinisikan sebagai jumlah dari semua *basic probability assignment* himpunan bagian (B) dari himpunan (A) ($B \subseteq A$). Batas atas atau *Plausibility* adalah jumlah dari semua *basic probability assignment* himpunan (B) yang beririsan dengan himpunan himpunan (A) dimana $B \cap A \neq \emptyset$. Untuk semua himpunan A yang merupakan elemen dari himpunan kuasa ($A \in P(X)$) (Senz dan Ferson, 2002).

$$Bel(A) = \sum_{B|B \subseteq A} m(B) \quad (2.4)$$

$$Pl(A) = \sum_{B|B \cap A \neq \emptyset} m(B) \quad (2.5)$$

Kedua ukuran, *belief* dan *plausibility* adalah *nonadditive*. Hal ini dapat diartikan sebagai tidak diharuskan untuk jumlah semua *belief* sama dengan 1 dan begitu juga untuk jumlah *plausibility*. Menurut (Kusumadewi, 2003) *plausibility* dapat dinotasikan dengan:

$$Pl(s) = 1 - Bel(-s) \quad (2.6)$$

2.3.1 Aturan Kombinasi Dempster-Shafer

Aturan kombinasi Dempster adalah murni operasi penghubung (dan/and/∧). Secara khusus, kombinasi (m_{12}) dihitung dari gabungan kedua bpa m_1 dan m_2 dengan cara sebagai berikut (Senz dan Ferson, 2002):

$$m_i(C) = \frac{\sum_{A \cap B = C} m_1(A)m_2(B)}{1 - K}, \text{ dimana } m_{1,2} \neq \emptyset \quad (2.7)$$

$$m_{12}(\emptyset) = 0$$

$$\text{dimana } K = \sum_{A \cap B = \emptyset} m_1(A)m_2(B) \quad (2.8)$$

Penyebut dalam aturan Dempster ini, $1-K$, merupakan faktor normalisasi. K merupakan massa *basic probability* yang merupakan jumlah hasil dari bpa dari semua himpunan dimana irisannya sama dengan null (menghasilkan himpunan kosong) (Senz dan Ferson, 2002).

2.4 Algoritma Genetika

Algoritma genetika merupakan salah satu teknik pencarian heuristik dimana teknik ini menirukan gagasan seleksi alam dan genetik pada proses evolusi. Algoritma genetik mengadopsi prinsip genetika dan seleksi alam dari teori evolusi Darwin. Algoritma ini ditemukan oleh John Holland dalam sebuah penelitian yang selanjutnya dipopulerkan oleh David Goldberg yang merupakan murid dari John Holland. Penelitian ini dilakukan pada tahun 1975 di Universitas Michigan, Amerika Serikat. Algoritma genetika ini dirancang untuk menirukan proses sistem alam yang diperlukan untuk evolusi, khususnya teori evolusi Charles Darwin, yaitu *survival of fittest*. Berdasarkan teori yang dicetuskan oleh Darwin, makhluk yang akan bertahan diantara persaingan untuk bertahan hidup adalah makhluk yang lebih kuat dari yang lain (Sutojo dkk., 2011).

Terdapat beberapa pengertian dasar dari algoritma genetika (Sutojo dkk., 2011).

1. Gen (Genotype) merupakan variabel dasar pembentuk suatu kromosom.
2. Allele merupakan nilai dari suatu gen, dapat berupa biner, float, integer maupun karakter.

3. Kromosom merupakan gabungan dari beberapa gen yang membentuk arti tertentu. Berikut ini beberapa bentuk kromosom, yaitu:
 - Kromosom biner, adalah kromosom yang bernilai biner.
 - Kromosom float, adalah kromosom yang bernilai pecahan, termasuk gen yang bernilai bulat. Kromosom ini merupakan model untuk jumlah parameternya banyak.
 - Kromosom string, yaitu kromosom yang bernilai string
 - Kromosom kombinatorial, yaitu kromosom yang gennya disusun dengan nilai berdasarkan urutannya.
4. Individu merupakan kumpulan gen, individu dapat dikatakan sama dengan kromosom. Individu menyatakan kemungkinan solusi dari suatu masalah.
5. Populasi merupakan kumpulan individu yang diproses secara bersama-sama dalam satu siklus evolusi
6. Generasi menyatakan satu satuan siklus evolusi
7. Nilai *fitness* merupakan nilai yang menyatakan seberapa baik individu atau solusi yang didapatkan dan menjadi acuan untuk mencapai nilai optimal. Tujuan dari algoritma genetika yaitu untuk mencari individu yang memiliki nilai *fitness* yang paling optimal (maksimum atau minimum, tergantung pada kebutuhan).

Dalam memecahkan masalah menggunakan algoritma genetika akan melewati tahapan-tahapan berikut.

2.4.1 Representasi Kromosom dan Inisialisasi Populasi

Inisialisasi populasi menghasilkan populasi awal secara acak sebanyak n individu yang disebut kromosom. Kromosom dilambangkan dengan string biner, integer ataupun real. Setiap posisi sting kromosom disebut gen dan menunjukkan karakteristik (variabel input) tertentu (Haupt dan Haupt, 2004).

2.4.2 Evaluasi

Fungsi fitness adalah untuk menentukan individu terbaik di mana setiap individu harus mewakili solusi untuk masalah (Hidalgo dkk., 2012). Algoritma genetika memiliki tujuan untuk memaksimalkan nilai fitness. Jika yang dicari adalah nilai *fitness* yang maksimal, maka nilai *fitness* sama dengan nilai dari fungsi itu sendiri (Sutojo, dkk., 2011). Untuk pencarian nilai maksimum, nilai *fitness* dapat diperoleh dengan menjadikan nilai hasil perhitungan suatu fungsi $f(x)$ menjadi *fitness* (Mahmudy, 2013).

$$fitness = f(x) \quad (2.9)$$

Sedangkan untuk mencari nilai minimum dapat menggunakan persamaan berikut (Mahmudy, 2013).

$$fitness = C - f(x) \quad (2.10)$$

atau

$$fitness = \frac{C}{f(x)} \quad (2.11)$$

Untuk persamaan 2.11, pembagi tidak boleh sama dengan nol. Pada penelitian ini menggunakan persamaan 2.12. Persamaan ini juga digunakan pada penelitian (Wiharto dkk., 2013).

$$fitness = \frac{correct_classification}{size_of_evaluation_set} \quad (2.12)$$

2.4.3 Tukar Silang (Crossover)

Pada proses seleksi alamiah individu akan melalui proses perubahan gen melalui proses perkembangbiakan. Proses perkembangbiakan pada individu menggunakan analogi perilaku kromosom dan struktur genetik (Sutojo dkk., 2011).

Dalam crossover terdapat istilah jumlah *offspring* (jumlah keturunan yang dihasilkan). Jumlah *offspring* didapat dari perkalian *crossover rate* dengan ukuran populasi. Jika terdapat 10 populasi dan *crossover rate*-nya 0,5, maka didapat 5 *offspring*. Ada beberapa metode *crossover* diantaranya (Mahmudy, 2013):

- *Extended Intermediate Crossover*

Metode ini membangkitkan *offspring* menggunakan persamaan yang ditunjukkan pada persamaan 2.13 dan 2.14 (Mahmudy, 2013).

$$C_1 = P_1 + \alpha(P_2 - P_1) \quad (2.13)$$

$$C_2 = P_2 + \alpha(P_1 - P_2) \quad (2.14)$$

dimana α dipilih secara acak antara -0,25 sampai dengan 1,25.

Misalkan *parent* yang terpilih $P_1 = [0.35, 0.65]$ dan $P_2 = [0.55, 0.78]$ dengan $\alpha = [0.11, 1.23]$, maka akan menghasilkan dua *offspring* sebagai berikut

$$C_1: \quad x_1 = 0.35 + 0.11(0.55 - 0.35) = 0.37$$

$$x_2 = 0.65 + 1.23(0.78 - 0.65) = 0.81$$

$$C_2: \quad x_1 = 0.55 + 0.11(0.35 - 0.55) = 0.53$$

$$x_2 = 0.78 + 1.23(0.65 - 0.78) = 0.62$$

Anak yang dihasilkan melalui perhitungan *offspring* adalah $C_1 = [0.37, 0.81]$ dan $C_2 = [0.53, 0.62]$.

- *One-Cut-Point Crossover*

Metode ini membagi susunan gen menjadi dua bagian, anak pertama akan mendapatkan susunan gen pertama dari induk pertama dan susunan gen kedua dari induk kedua, sebaliknya anak kedua mendapatkan susunan gen kedua dari

induk pertama dan susunan gen kedua dari induk kedua (Mahmudy, 2013). Contohnya sebagai berikut:

$$\begin{array}{lcl} P_1 & [& 1 \ 0 \mid 0 \ 1 \\ P_2 & [& 0 \ 1 \mid 1 \ 0 \\ C_1 & [& 1 \ 0 \mid 1 \ 0 \\ C_2 & [& 0 \ 1 \mid 0 \ 1 \end{array}$$

2.4.4 Mutasi

Selain *crossover* operator genetik lain yang dapat dilakukan adalah mutasi. Sama seperti *crossover*, pada mutasi juga terdapat *mutation rate* yang nantinya akan dikalikan dengan ukuran populasi yang menghasilkan jumlah *offspring*. Berikut ini beberapa metode yang ada pada mutasi (Mahmudy, 2013):

- *Random Mutation*

Metode ini membangkitkan gen kromosom anak dengan menggunakan persamaan yang ditunjukkan oleh persamaan 2.15 berikut (Mahmudy, 2013):

$$x'_i = x'_i + r(\max_i - \min_j) \quad (2.15)$$

Keterangan

- x'_i merupakan gen induk yang dipilih untuk nantinya dapat dikurangi atau ditambahkan bilangan random
- r merupakan bilangan random
- $[\min_j, \max_j]$ domain variabel x_j

Misalkan *parent* yang terpilih $P = [0.45, 0.73]$ dengan $r = -0.06$, maka akan menghasilkan dua *offspring* sebagai berikut:

$$\begin{array}{lcl} C: & x_1 = 0.45 - 0.06(0.53 - 0.37) = 0.44 \\ & x_2 = 0.73 \text{ (tetap)} \end{array}$$

Anak yang dihasilkan berdasarkan perhitungan *Random Mutation* adalah $C = [0.44, 0.73]$.

- *Reciprocal Exchange Mutation*

Metode ini menukarkan posisi gen *parent* (exchange point / XP) yang dipilih secara random (Mahmudy, 2013). Contohnya sebagai berikut:

$$\begin{array}{lcl} P_1 & [& 2 \ 4 \ 1 \ 5 \ 3 \\ C_1 & [& 2 \ 5 \ 1 \ 4 \ 3 \end{array}$$

- *Insertion Mutation*

Metode *Insertion Mutation* bekerja dengan menyisipkan satu posisi yang dipilih secara random sehingga menggeser posisi yang lainnya (Mahmudy, 2013). Contohnya adalah sebagai berikut:

P₁ [2 4 1 5 3]
 C1 [1 2 4 5 3]

2.4.5 Seleksi

Setelah operator genetik dilakukan, tahapan berikutnya adalah seleksi individu, generasi baru dievaluasi dan digunakan dalam iterasi berikutnya. Iterasi dalam algoritma sampai jumlah maksimum generasi atau solusi yang memuaskan tercapai (Correa dkk., 2011), atau jika keduanya tidak menghasilkan solusi yang baik maka dapat menentukan batas waktu maksimum (Mahmudy, 2013). Terdapat beberapa jenis metode seleksi, yaitu (Mahmudy, 2013):

- *Replacement Selection*

Metode ini memiliki beberapa ketentuan diantaranya sebagai berikut (Mahmudy, 2013):

- Jika *offspring* yang dihasilkan dari proses *crossover* memiliki *fitness* yang lebih baik dari induknya, maka *offspring* akan menggantikan induk yang memiliki *fitness* terkecil.
- Jika *offspring* yang dihasilkan dari proses mutasi memiliki *fitness* yang lebih baik dari induknya, maka *offspring* akan menggantikan induknya.

Dimisalkan terdapat populasi induk seperti yang ditunjukkan pada Tabel 2.1:

Tabel 2.1 Contoh populasi untuk metode *Replacement Selection*

Individu	<i>Fitness</i>
P ₁	13
P ₂	11
P ₃	15
P ₄	16
P ₅	19

Dari Tabel 2.1 dianggap menghasilkan *offspring* seperti yang ditunjukkan pada Tabel 2.2.

Tabel 2.2 Contoh *offspring* yang dihasilkan setelah proses *crossover* dan mutasi

Individu	<i>Parents</i>	<i>Fitness</i>
C ₁	P ₂ dan P ₃	18
C ₂	P ₄ dan P ₅	16
C ₃	P ₁	12

Berdasarkan data yang ditunjukkan pada Tabel 2.1 dan Tabel 2.2, C₁ menghasilkan *fitness* yang lebih baik dari induknya, karena C₁ merupakan

offspring yang dihasilkan dari proses *crossover* dari induk P_2 dan P_3 , dan P_2 merupakan *parent* yang memiliki *fitness* yang lebih rendah dibandingkan P_3 maka C_1 akan menggantikan P_2 . Hasil seleksi dengan menggunakan metode *Replacement Selection* ditunjukkan pada Tabel 2.3.

Tabel 2.3 Contoh hasil seleksi dengan menggunakan metode *Replacement Selection*

Individu	Asal	<i>Fitness</i>
P_1	P_1	13
P_2	C_1	11
P_3	P_3	15
P_4	P_4	16
P_5	P_5	19

- *Roulette Wheel*

Langkah-langkah yang dilakukan pada metode seleksi *Roulette Wheel* ini adalah sebagai berikut (Mahmudy, 2013) :

- Menghitung total *fitness* dengan menggunakan persamaan 2.16

$$\text{totalFitness} = \sum_{k=1}^{\text{popSize}} \text{fitness}(\text{Parent}_k) \quad (2.16)$$

- Menghitung probabilitas seleksi (*prob*) dengan menggunakan persamaan 2.17

$$\text{prob}_k = \frac{\text{fitness}(\text{Parent}_k)}{\text{totalFitness}} \quad (2.17)$$

- Menghitung nilai probabilitas kumulatif (*probCum*) dengan menggunakan persamaan 2.18

$$\text{probCum}_k = \sum_{j=1}^k \text{prob}_j \quad (2.18)$$

- *Tournament Selection*

Metode ini mengambil beberapa individu, yang nantinya individu dengan nilai terbesar akan menjadi individu baru pada generasi baru, proses ini dilakukan sebanyak ukuran populasi yang ditentukan (Mahmudy, 2013). Misalnya terdapat populasi seperti pada Tabel 2.4.

Tabel 2.4 Contoh populasi untuk metode *Tournament Selection*

P(t+1)	<i>Fitness</i>
P ₁	24,56
P ₂	26,73
P ₃	30,54
P ₄	31,22
P ₅	23,55
C ₁	29,23
C ₂	33,02
C ₃	30.11

Berdasarkan populasi yang ditunjukkan pada Tabel 2.4, selanjutnya akan menyeleksi individu, hasil seleksi yang didapatkan ditunjukkan pada Tabel 2.5.

Tabel 2.5 Contoh hasil seleksi menggunakan metode *Tournament Selection*

P(t+1)	Individu Pertama	Individu Kedua	Individu Terpilih	<i>Fitness</i>
P ₁	P ₁	C ₂	C ₂	33,02
P ₂	C ₃	C ₁	C ₃	30.11
P ₃	P ₂	P ₄	P ₄	31,22
P ₄	P ₅	C ₃	C ₃	30.11
P ₅	P ₃	P ₁	P ₃	30,54

- ***Elitism Selection***

Pada metode ini akan dilakukan pengumpulan individu-individu yang ada pada populasi (*parents* dan *offspring*). Semua individu akan diurutkan dari *fitness* terbesar sampai terkecil. Individu sejumlah ukuran populasi teratas akan menjadi generasi baru (Mahmudy, 2013). Misalkan terdapat populasi seperti yang ditunjukkan pada Tabel 2.6.

Tabel 2.6 Contoh populasi untuk metode *Elitism Selection*

P(t+1)	<i>Fitness</i>
P ₁	24,56
P ₂	26,73
P ₃	30,54
P ₄	31,22
P ₅	23,55
C ₁	29,23
C ₂	33,02
C ₃	30.11

Selanjutnya populasi diurutkan berdasarkan nilai *fitnessnya*, dan hasilnya adalah seperti yang ditunjukkan pada Tabel 2.7.

Tabel 2.7 Hasil perankingan semua individu

Rank	P(t+1)	<i>Fitness</i>
1	C ₂	33,02
2	P ₄	31,22
3	P ₃	30,54
4	C ₃	30,11
5	C ₁	29,23
6	P ₂	26,73
7	P ₁	24,56
8	P ₅	23,55

Ukuran populasi yang ditentukan adalah 5, sehingga hasil seleksi dengan menggunakan metode *Elitism Selection* ditunjukkan pada Tabel 2.8.

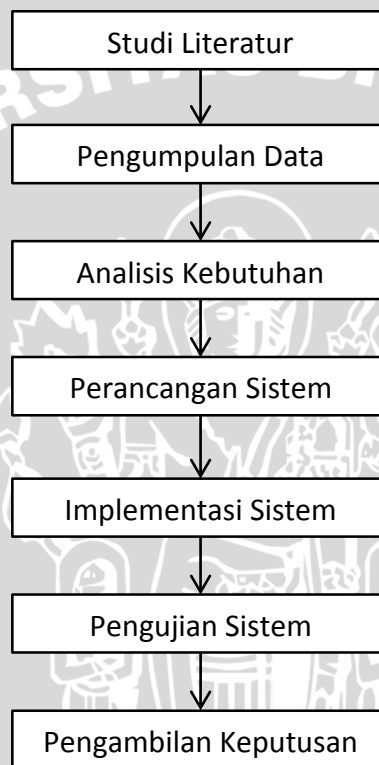
Tabel 2.8 Hasil seleksi menggunakan metode *Elitism Selection*

Individu	Asal	<i>Fitness</i>
P ₁	C ₂	33,02
P ₂	P ₄	31,22
P ₃	P ₃	30,54
P ₄	C ₃	30,11
P ₅	C ₁	29,23

BAB 3 METODOLOGI PENELITIAN

Bab ini menjelaskan tentang metode dan perancangan dalam penelitian Implementasi Algoritma Genetika Untuk Optimasi Nilai Densitas Metode Dempster-Shafer pada Deteksi Penyakit Sapi.

Metodologi penelitian yang digunakan memiliki beberapa tahapan, yaitu studi literatur, pengumpulan data, analisa kebutuhan, perancangan sistem, implementasi sistem, pengujian sistem dan pengambilan keputusan. Tahapan-tahapan tersebut digambarkan dengan diagram blok seperti pada Gambar 3.1.



Gambar 3.1 Diagram Blok Metodologi Penelitian

3.1 Studi Literatur

Tahapan studi literatur ini mempelajari tentang literatur-literatur dari berbagai sumber yang berhubungan dengan optimasi nilai densitas metode Dempster-Shafer untuk deteksi penyakit sapi menggunakan algoritma genetika. Literatur-literatur yang dipelajari berisi tentang:

- Berbagai penyakit sapi dan gejala yang dialami sapi yang terjangkit
- Metode Dempster-Shafer
- Algoritma genetika

Teori-teori tersebut diperoleh dari buku, *e-book*, jurnal, laporan penelitian sejenis dan sumber lainnya.

3.2 Pengumpulan Data

Data yang diperlukan pada penelitian ini adalah data latih dan data uji. Penelitian ini merupakan pengembangan penelitian sebelumnya yaitu penelitian Mustikadewi Prihastuti (2014). Dalam mengoptimasi nilai densitas diperlukan data latih. Karena data yang terdapat pada penelitian sebelumnya hanya 18 data, sementara penyakit yang terdapat pada pengetahuan 30 maka diperlukan data latih yang banyaknya dapat mewakili semua penyakit. Sehingga pada pelatihan menggunakan data latih yang merupakan data *dummy*. Data *dummy* merupakan data yang hampir mirip dengan data kasus nyata dan pembuatannya dikonsultasikan kepada pakar.

Data latih yang digunakan sebanyak 100 data, sedangkan data uji yang digunakan sebanyak 18 data yang diambil dari penelitian sebelumnya. Data latih dan data uji terdapat pada Lampiran D dan Lampiran C. Tabel 3.1 menunjukkan beberapa bentuk data yang diberikan oleh pakar.

Tabel 3.1 Bentuk data yang diperoleh

No.	Penyakit	Gejala
1.	Demam Tiga Hari (<i>Bovine Ephemeral Fever</i>)	1. Demam / Panas ($>40^{\circ}\text{C}$) selama tiga hari 2. Tubuh lemah dan lesu 3. Kekuatan anggota gerak sampai pincang 4. Duduk / Tidak bisa berdiri / Kelemahan anggota gerak 5. Susah bernafas 6. Keluar air liur berlebihan 7. Keluar sedikit cairan dari mata dan hidung 8. Produksi susu menurun pada sapi yang laktasi 9. Gemetar
2.	Diare Ganas Menahun (<i>Bovine Viral Diarrhoea</i>)	1. Demam ringan atau tinggi 2. Nafsu makan hilang / berkurang 3. Depresi 4. Diare berlendir dan ada bercak darah 5. Air liur berlebihan 6. Keluar leleran dari lubang hidung 7. Erosi pada selaput lendir hidung, lidah bibir, gusi dan bagian belakang maksila
3.	<i>Maligant Chartaral Fever</i>	1. Demam / Panas (40°C-41°C) 2. Kadang terjadi radang kulit (penebalan dan pengelupasan kulit) 3. Nafsu makan hilang / berkurang 4. Otot-otot gemetar / sempoyongan

No.	Penyakit	Gejala
		5. Kelumpuhan sebelum mati 6. Mata dan hidung mengeluarkan leleran kental dan mukopurulon 7. Selaput lendir mulut kemerahan / peradangan mulut dan lepuhan dipermukaan lidah 8. Air liur yang berlebihan 9. Tubuh kurus dan lemah 10. Moncong kering dan pecah-pecah terisi nanah 11. Susah bernafas (hidung tersumbat kerak) 12. Kornea mata keruh dan keputihan (jika parah menjadi buta)

3.3 Analisis Kebutuhan

Analisis kebutuhan bertujuan untuk menganalisis dan mendapatkan semua kebutuhan yang diperlukan dalam membentangkan sistem agar tidak menyimpang dari permasalahan dan tujuan penelitian. Analisis kebutuhan dilakukan dengan mengidentifikasi kebutuhan sistem antara lain kebutuhan masukan, kebutuhan proses dan kebutuhan keluaran. Kebutuhan ini akan dimodelkan dalam *use case diagram*, *sequence diagram* dan *class diagram*. Hal ini bertujuan untuk memudahkan dalam memahami sistem.

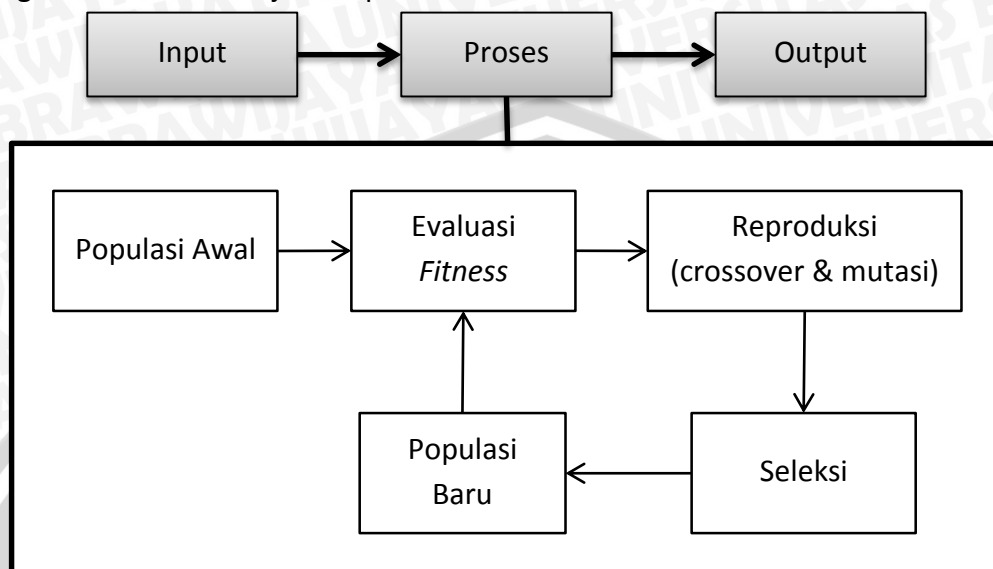
3.4 Perancangan Sistem

Tahap perancangan sistem bertujuan untuk mengubah informasi awal yang diperoleh pada tahapan analisis menjadi model yang akan digunakan dalam pengimplementasian sistem. Pada tahapan ini akan menjelaskan perancangan pada setiap proses pembentukan nilai densitas menggunakan algoritma genetika dan proses untuk menguji nilai densitas tersebut menggunakan metode Dempster-Shafer. Perancangan algoritma sistem menggunakan notasi diagram alir (*flowchart*).

Pada penelitian ini menggunakan 30 penyakit dengan 179 gejala, sehingga tidak memungkinkan untuk mengoptimasi biasa tanpa menggunakan metode optimasi. Tujuan optimasi ini untuk mendapatkan nilai densitas yang konsisten dan optimal. Ketidakkonsistenan terjadi diakibatkan oleh fakta-fakta yang kompleks dan faktor-faktor lain yang mempengaruhi pakar dari dalam menentukan nilai densitas pakar. Metode optimasi yang digunakan yaitu algoritma genetika, hal ini didukung oleh penelitian-penelitian sebelumnya yang menunjukkan bahwa algoritma genetika dapat dengan baik dalam mengoptimasi dan menghasilkan akurasi yang lebih baik dari penelitian sebelumnya.

Langkah-langkah yang ada pada sistem yaitu mengambil masukan berupa parameter-parameter algoritma genetika dan nilai densitas yang diberikan oleh

pakar, melakukan proses optimasi menggunakan algoritma genetika dan terakhir menghasilkan output berupa nilai densitas yang telah dioptimasi. Langkah-langkah tersebut ditunjukkan pada Gambar 3.2.



Gambar 3.2 Diagram blok desain arsitektur sistem

3.5 Implementasi Sistem

Implementasi perangkat lunak dilakukan dengan mengacu kepada perancangan aplikasi. Pada penelitian ini akan menerapkan algoritma genetika untuk mengoptimnasi nilai densitas pada metode Dempster-Shafer. Bahasa pemrograman yang digunakan adalah C#. Tahapan ini meliputi pembuatan antarmuka, perhitungan algoritma genetika yang memiliki beberapa proses diantaranya pembentukan populasi, perhitungan *fitness*, proses *crossover* dan mutasi, proses seleksi, dan pembentukan generasi baru, proses ini kembali berulang dari perhitungan *fitness* untuk generasi baru hingga kondisi berhenti yang ditentukan tercapai atau hingga nilai densitas yang optimum didapatkan.

3.6 Pengujian Sistem

Pada tahapan ini dilakukan pengujian keberhasilan sistem yang telah dibuat pada tahapan implementasi. Pengujian sistem ini bertujuan untuk mengetahui apakah sistem yang dibuat sudah sesuai dengan yang diharapkan. Pengujian yang dilakukan meliputi pengujian parameter algoritma genetika dan pengujian akurasi.

1. Pengujian parameter algoritma genetika yaitu pengujian nilai probabilitas *crossover* dan mutasi, ukuran populasi dan jumlah generasi.
 - a. Pengujian probabilitas *crossover* dan mutasi dilakukan dengan melihat pengaruh perubahan probabilitas *crossover* dan mutasi terhadap nilai *fitness* yang dihasilkan. Pengujian akan dilakukan terhadap 9 nilai probabilitas *crossover* dan mutasi yaitu 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8 dan 0.9. Pada setiap pengujian satu kombinasi akan dilakukan sebanyak 5 kali percobaan,

lalu akan diambil nilai rata-rata *fitness*-nya yang nantinya akan dilakukan perbandingan terhadap *fitness* kombinasi yang lain. Tabel pengujian probabilitas *crossover* dan probabilitas mutasi ditunjukkan oleh Tabel 3.2.

Tabel 3.2 Tabel Pengujian Probabilitas *Crossover* dan Mutasi

Kombinasi	Percobaan ke- <i>i</i>					Rata-rata
	I	II	III	IV	V	
[0.1, 0.1]						
[0.1, 0.2]						
[0.1, 0.3]						
[0.1, 0.4]						
[0.1, 0.5]						
[0.1, 0.6]						
[0.1, 0.7]						
[0.1, 0.8]						
[0.1, 0.9]						
[0.2, 0.1]						
⋮						
[0.9, 0.9]						

- b. Sama seperti pengujian yang lain, pengujian terhadap jumlah generasi juga akan melihat pengaruh perubahan jumlah generasi yang digunakan terhadap nilai *fitness* yang dihasilkan. Pengujian ini akan dilakukan dengan mengubah jumlah generasi dengan nilai 10, 50, 100, 150, 200, 250, 300, 350, 400, 450, 500, 550, 600, 650 dan 700. Pada setiap pengujian satu jumlah generasi akan dilakukan sebanyak 5 kali percobaan, lalu akan diambil nilai rata-rata *fitness*-nya yang nantinya akan dilakukan perbandingan terhadap *fitness* jumlah generasi yang lain. Tabel pengujian jumlah generasi ditunjukkan pada Tabel 3.3.

Tabel 3.3 Tabel Pengujian Jumlah Generasi

Jumlah Generasi	Percobaan ke- <i>i</i>					Fitess Rata-rata
	1	2	3	4	5	
10						
50						
100						
150						
200						
250						
300						
350						
⋮						
700						

- c. Pengujian terhadap ukuran populasi dilakukan dengan melihat pengaruh perubahan ukuran populasi yang digunakan terhadap nilai *fitness* yang dihasilkan. Pengujian ini akan mengubah ukuran populasi dengan nilai 10, 15, 20, 25 dan 30. Pada setiap pengujian satu ukuran populasi akan dilakukan sebanyak 5 kali percobaan, lalu akan diambil nilai rata-rata *fitness*-nya yang nantinya akan dilakukan perbandingan terhadap *fitness* ukuran populasi yang lain. Tabel pengujian ukuran populasi ditunjukkan pada Tabel 3.4.

Tabel 3.4 Tabel Pengujian Ukuran Populasi

Ukuran Populasi	Percobaan ke- <i>i</i>					Rata-rata
	I	II	III	IV	V	
10						
15						
20						
25						
30						

2. Pengujian akurasi dilakukan dengan menggunakan nilai densitas yang mempunyai *fitness* paling tinggi. Nilai ini diujikan terhadap 18 data kasus penyakit sapi. Pada pengujian ini hasil diagnosa sistem akan dibandingkan dengan diagnosa pakar. Jumlah hasil sistem yang cocok dengan pakar dibagi dengan jumlah data yang digunakan yaitu 18, kemudian dikalikan dengan 100%. Pengujian akurasi dilakukan sebanyak 10 kali, kemudian dirata-rata. Tabel pengujian akurasi ditunjukkan pada Tabel 3.5.

Tabel 3.5 Tabel Pengujian Akurasi

Percobaan ke- <i>i</i>	<i>Fitness</i>
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
Rata-rata	

3.7 Pengambilan Kesimpulan

Penarikan kesimpulan dilakukan setelah semua tahapan analisis, perancangan, implementasi dan pengujian telah selesai dilakukan. Kesimpulan diambil dari hasil pengujian dan analisis metode yang diterapkan. Selain penarikan kesimpulan, tahap terakhir dari penulisan yaitu saran yang dimaksudkan untuk memberikan masukan dan pertimbangan untuk pengembangan selanjutnya.



BAB 4 PERANCANGAN

Pada bab ini akan membahas mengenai perancangan sistem yang meliputi analisa kebutuhan perangkat lunak, perancangan sistem, perhitungan manual dan perancangan antarmuka.

4.1 Analisis Kebutuhan

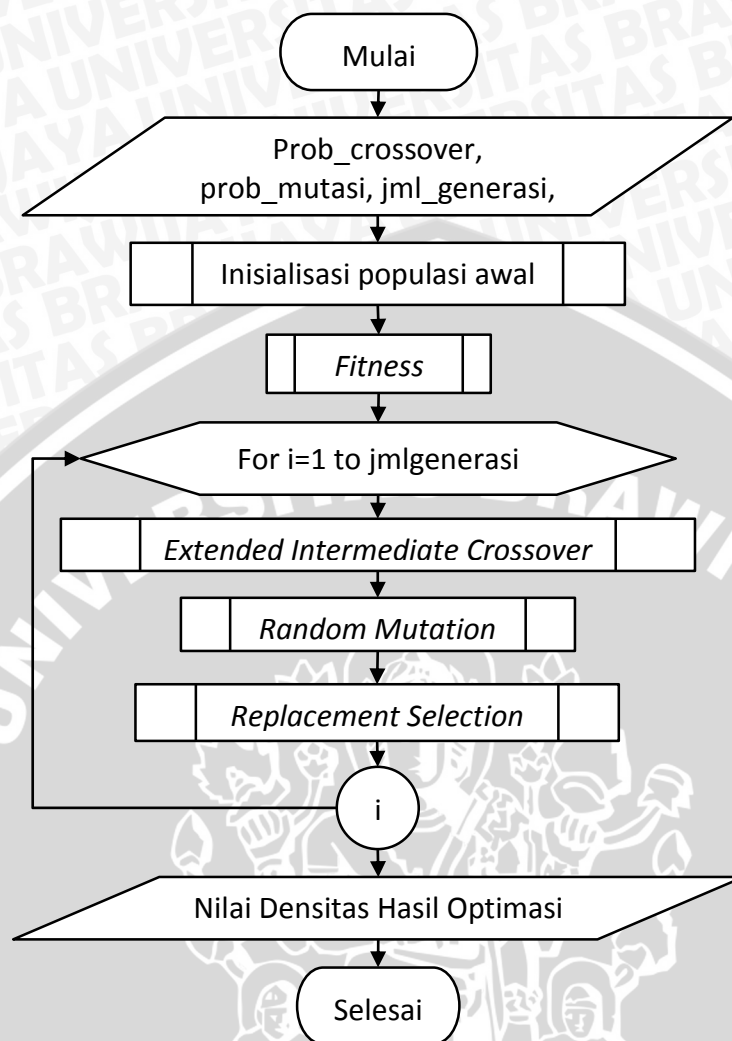
Pada tahapan analisis kebutuhan akan dilakukan analisis kebutuhan perangkat lunak yang meliputi analisis spesifikasi perangkat lunak. Hal ini dimaksudkan agar mempermudah dalam memahami sistem yang akan dibangun.

Tabel 4.1 Daftar Kebutuhan Sistem

Kebutuhan	Aktor	Proses
Sistem menyediakan antarmuka untuk form input probabilitas <i>crossover</i> , probabilitas mutasi, jumlah generasi dan ukuran populasi dan antarmuka untuk melihat proses-proses yang ada di dalam algoritma genetika dalam mengoptimasi nilai densitas	User	Membentuk Nilai Densitas
Sistem menyediakan antarmuka untuk melihat hasil akurasi sistem	User	Menguji Akurasi Sistem

4.2 Perancangan Sistem

Perancangan sistem menggunakan flowchart untuk menggambarkan proses-proses pada sistem. Proses optimasi pada sistem ini menggunakan algoritma genetika. Algoritma genetika terdiri dari beberapa proses, diantaranya inisialisasi populasi awal, evaluasi *fitness*, proses reproduksi (*crossover* dan mutasi) dan seleksi. Proses-proses tersebut ditunjukkan pada Gambar 4.1.



Gambar 4.1 Flowchart Algoritma Genetika

Proses pada sistem diawali dengan memasukkan parameter-parameter algoritma genetika yang diperlukan sistem, diantaranya probabilitas *crossover* (pc), probabilitas mutasi (pm), jumlah generasi dan ukuran populasi (popSize). Setelah memasukkan parameter, akan dilakukan inisialisasi populasi awal dan dilanjutkan dengan menghitung nilai *fitness* tiap individu. Langkah selanjutnya adalah proses reproduksi yang terdiri dari *crossover* dan mutasi. Setelah mendapatkan individu baru (*offspring/child*), proses selanjutnya adalah proses seleksi individu. Pada proses seleksi ini akan didapatkan generasi baru yang akan mengulang proses dari proses reproduksi.

4.2.1 Inisialisasi Populasi Awal

4.2.1.1 Disain Kromosom

Nilai densitas pada kromosom menunjukkan nilai masing-masing gejala terhadap penyakit. Nilai ini berbentuk *float* dengan *range* 0-1. Ukuran kromosom pada penelitian ini adalah 179x30, 179 untuk gejala dan 30 untuk penyakit. Penyakit pada penelitian ini yaitu Antrak (A), Brucellosis (BR), Mastitis (M), Tuberkulosis (TB), Blackleg (BL), Pink Eye (PE), Johne Disease (JD), Salmonelosis (S), Botulisme (BT), Radang Kuku (RK), Vibriosis (V), Ngorok (N), Kelumpuhan (K), Sindrom Sapi Ambruk (SA), Demam 3 Hari (DT), Diare Ganas Menahun (BV), Malignan Chataral Fever (MC), Penyakit Mulut dan Kuku (MK), Bovine Herpes Virus (HV), Ringworm (R), Aspergilosis (AS), Retensi Plasenta (RP), Distokia (D), Indigesti Sederhana (IS), Kembung (KB), Asidosis Rumen (AR), Surra (SR), Cacingan (C), Kudisan Menular (KM) dan Belatungan (B). Gejala untuk penelitian ini ditunjukkan pada Lampiran B. Bentuk kromosom ditunjukkan pada Tabel 4.2

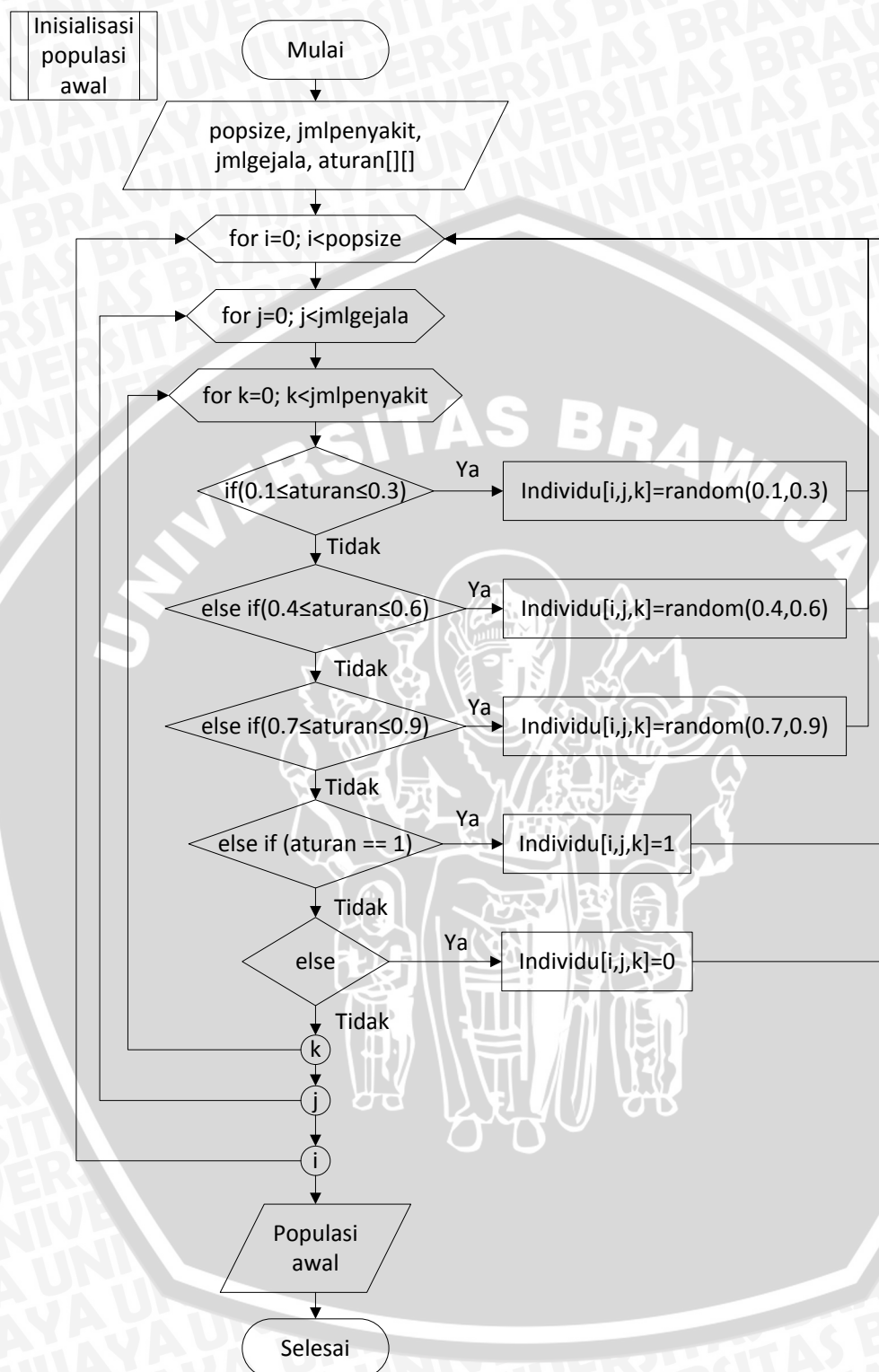
Tabel 4.2 Bentuk Kromosom

	A	BR	M	TB	BL	...	B
1	a_{11}	a_{12}	a_{13}	a_{14}	a_{15}		a_{1n}
2	a_{21}	a_{22}	a_{23}	a_{24}	a_{25}		a_{2n}
3	a_{31}	a_{32}	a_{33}	a_{34}	a_{35}		a_{3n}
4	a_{41}	a_{42}	a_{43}	a_{44}	a_{45}		a_{4n}
5	a_{51}	a_{52}	a_{53}	a_{54}	a_{55}		a_{5n}
6	a_{61}	a_{62}	a_{63}	a_{64}	a_{65}		a_{6n}
7	a_{71}	a_{72}	a_{73}	a_{74}	a_{75}		a_{7n}
8	a_{81}	a_{82}	a_{83}	a_{84}	a_{85}		a_{8n}
9	a_{91}	a_{92}	a_{93}	a_{94}	a_{95}		a_{9n}
10	a_{101}	a_{102}	a_{103}	a_{104}	a_{105}		a_{10n}
⋮							
179	a_{m1}	a_{m2}	a_{m3}	a_{m4}	a_{m5}		a_{mn}

Pada Tabel 4.2, nilai a_{11} menunjukkan nilai densitas untuk gejala 1 terhadap penyakit Antraks (A).

4.2.1.2 Perancangan Inisialisasi Populasi Awal

Pembentukan populasi awal dilakukan dengan membangkitkan bilangan random. Setelah sistem menerima ukuran populasi (*popSize*), populasi akan dibentuk sejumlah *popSize*. Proses pembentukan populasi awal ditunjukkan pada Gambar 4.2.



Gambar 4.2 Flowchart Proses Inisialisasi Populasi Awal

Proses inisialisasi populasi awal memerlukan sejumlah variabel, diantaranya ukuran populasi (popSize), jumlah penyakit dan jumlah gejala. Pembangkitan

bilangan secara acak berulang sebanyak jumlah gejala yang berulang sebanyak jumlah penyakit. Sehingga bilangan yang dibangkitkan berbentuk array berukuran 179x30. Pembangkitan individu ini berulang sebanyak ukuran populasi. Bayak penyakit yang digunakan pada penelitian ini yaitu 30 penyakit dan total gejala yang digunakan adalah 179 gejala. Daftar penyakit dan gejala ditunjukkan pada Lampiran A dan Lampiran B.

Pembentukan populasi awal dilakukan dengan membangkitkan bilangan random dengan menggunakan *range* tertentu. *Range* tersebut memiliki aturan yaitu:

- jika densitas yang diberikan oleh pakar berada pada range 0,1-0,3, maka gen diacak di antara range 0,1-0,3,
 - jika densitas yang diberikan oleh pakar berada pada range 0,4-0,6, maka gen diacak di antara range 0,4-0,6,
 - jika densitas yang diberikan oleh pakar berada pada range 0,7-0,9, maka gen diacak di antara range 0,7-0,9,
 - jika densitas yang diberikan adalah 1 maka gen ditetapkan 1, dan
 - jika densitas yang diberikan adalah 0, maka gen akan ditetapkan bernilai 0.
- Pembagian range ditunjukkan pada Tabel 4.3.

Tabel 4.3 Pembentukan Bilangan *Random* pada Populasi Awal

No.	Penyakit	ID Gejala yang dibangkitkan secara acak			
		Range (0,1-0,3)	Range (0,4-0,6)	Range (0,7-0,9)	1
1	A	43, 166	15, 16, 17, 18, 19, 35, 38, 58, 76, 103, 118, 145	39, 79, 91, 115, 121, 165	
2	BR	15, 22	23, 32, 40, 115, 120, 138, 162	75, 88, 130, 132	
3	M		22, 38, 40, 61, 101, 109, 115, 138, 140, 172	8, 119, 168	4,5, 6, 7
4	TB		138, 172	12, 21, 165	69
5	BL		127	26, 34, 108, 151	
6	PE	159	63, 125, 152	67, 175, 176	
7	JD		49, 127, 138, 171, 172	46	
8	S			38, 44, 103, 115, 138	

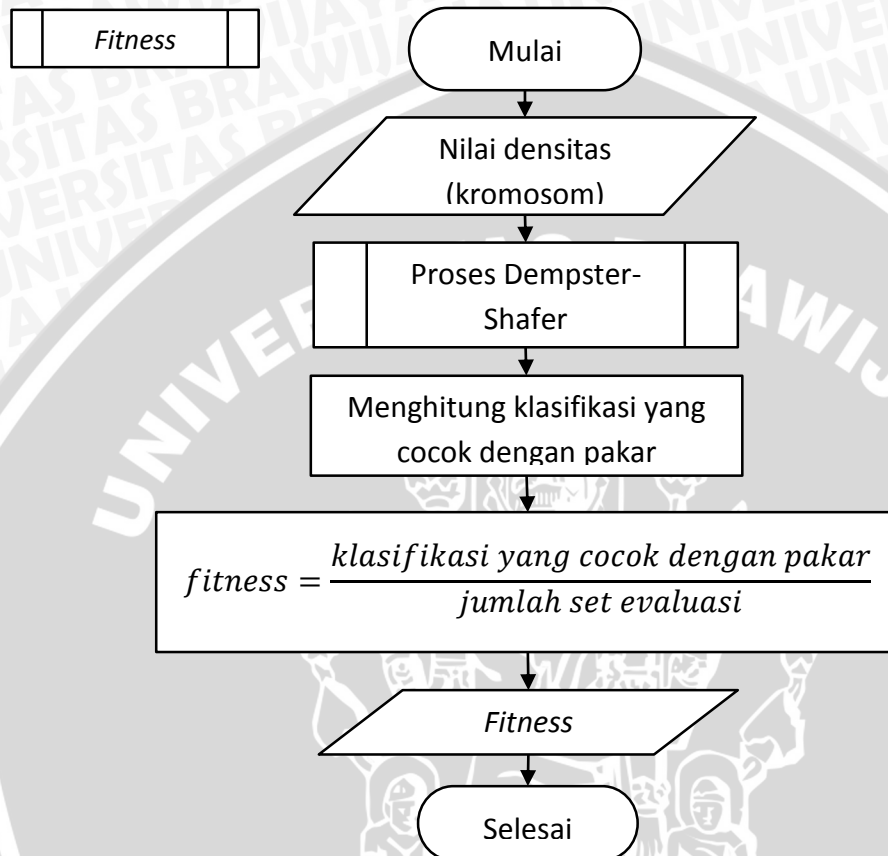
No.	Penyakit	ID Gejala yang dibangkitkan secara acak			
		Range (0,1-0,3)	Range (0,4-0,6)	Range (0,7-0,9)	1
9	BT		47, 48, 57, 112, 114	85, 111, 165	
10	RK			102, 173	33, 99, 100, 153, 155
11	V		126	2, 90	
12	N	158	12, 59, 103, 115, 172	24, 28, 30, 31, 39, 81, 165	
13	K		36, 40, 58, 115, 148	25, 47, 86	
14	SA		40, 89, 115, 169	47, 71, 86, 145, 161, 167, 170	
15	DT		112, 115, 134, 140, 165	37, 38, 47, 59, 77, 82, 83, 103, 138	
16	BV		58, 83, 112, 115	38, 40, 41, 44, 45, 50	
17	MC		39, 70, 103, 115, 172	82, 87, 98, 110, 112, 113, 116, 129, 158, 165	
18	MK		109, 115	38, 102, 104, 105, 128, 137	
19	HV		115	3, 12, 20, 38, 39, 80, 83, 84, 94, 131, 160, 165, 177, 178	
20	R		115, 172	51, 124, 133, 140, 141	
21	AS			1, 13, 38, 83, 134	
22	RD		38, 58, 115, 134, 138, 144, 156, 178	14, 97	
23	D			72, 97, 163,	

No.	Penyakit	ID Gejala yang dibangkitkan secara acak			
		Range (0,1-0,3)	Range (0,4-0,6)	Range (0,7-0,9)	1
				164, 176	
24	IS		10, 140	38, 40, 43, 53, 54, 65, 96, 101, 103, 109, 115, 136, 146, 147	
25	KB		10, 47, 103, 115, 139	55, 58, 61, 135, 136, 145, 149, 160, 165	
26	AR			36, 38, 42, 43, 54, 64, 103, 109, 115, 134, 143, 146, 148, 150	
27	SR		39, 103, 115, 140, 172	9, 27, 68, 82, 157	
28	C		10, 38, 40, 43, 52	9, 11, 29, 36, 58, 78, 103, 107, 115, 117, 122, 139, 140, 142, 152, 154, 172	
29	KM			60, 92, 106, 123, 140	
30	B		58, 115, 127, 138	9, 38, 66, 93, 179	

Penelitian ini tidak menggunakan random 0 - 1 karena penelitian ini memiliki ukuran kromosom yang besar yaitu 179 x 30, sehingga memerlukan waktu yang cukup lama untuk mendapatkan individu yang optimal jika menggunakan random 0 - 1.

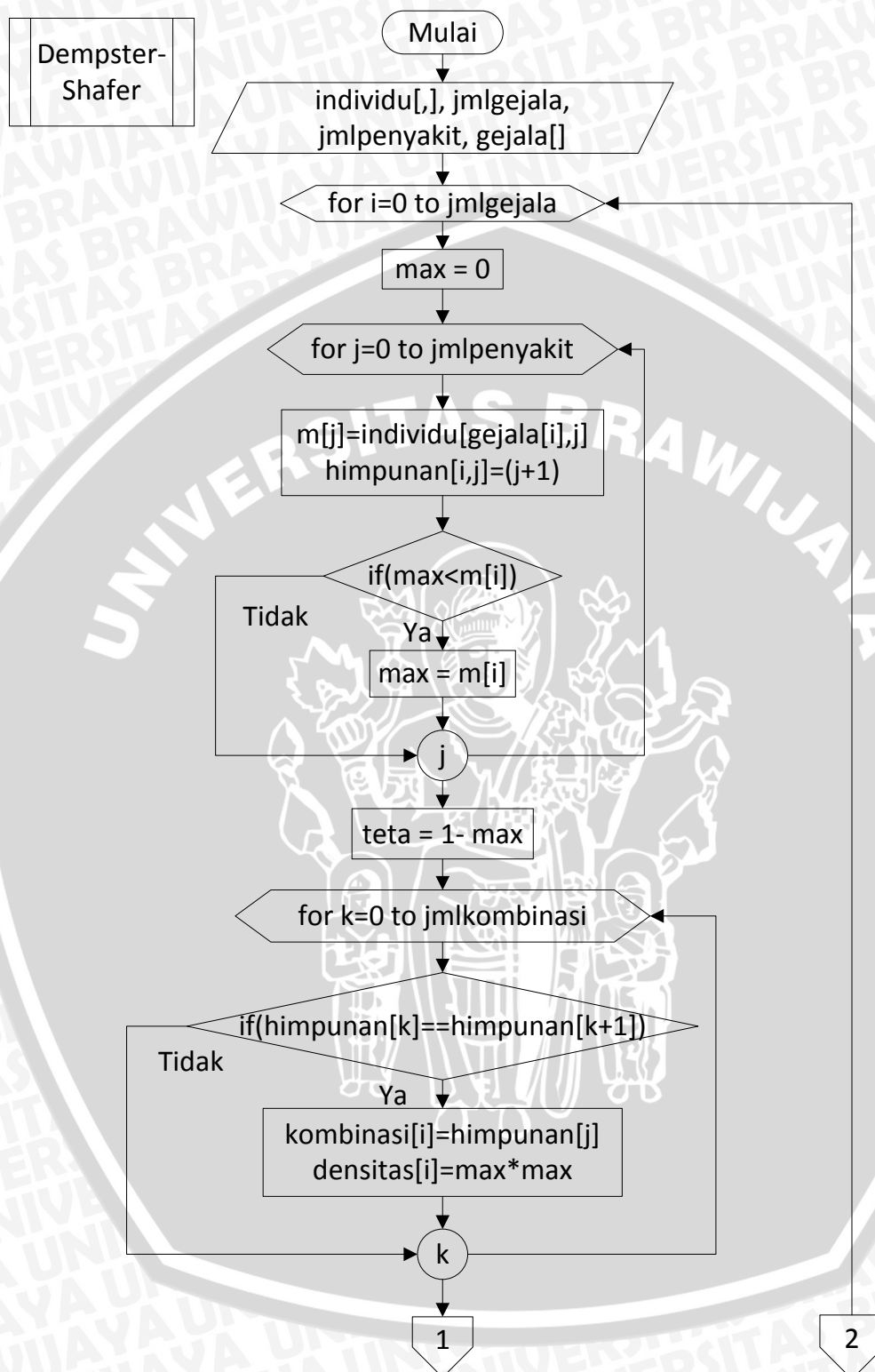
4.2.2 Fitness

Setelah membangkitkan nilai densitas secara acak, proses berikutnya adalah menghitung nilai *fitness*. Perhitungan *fitness* menggunakan perhitungan akurasi. Langkah-langkah dalam proses evaluasi digambarkan pada Gambar 4.3.

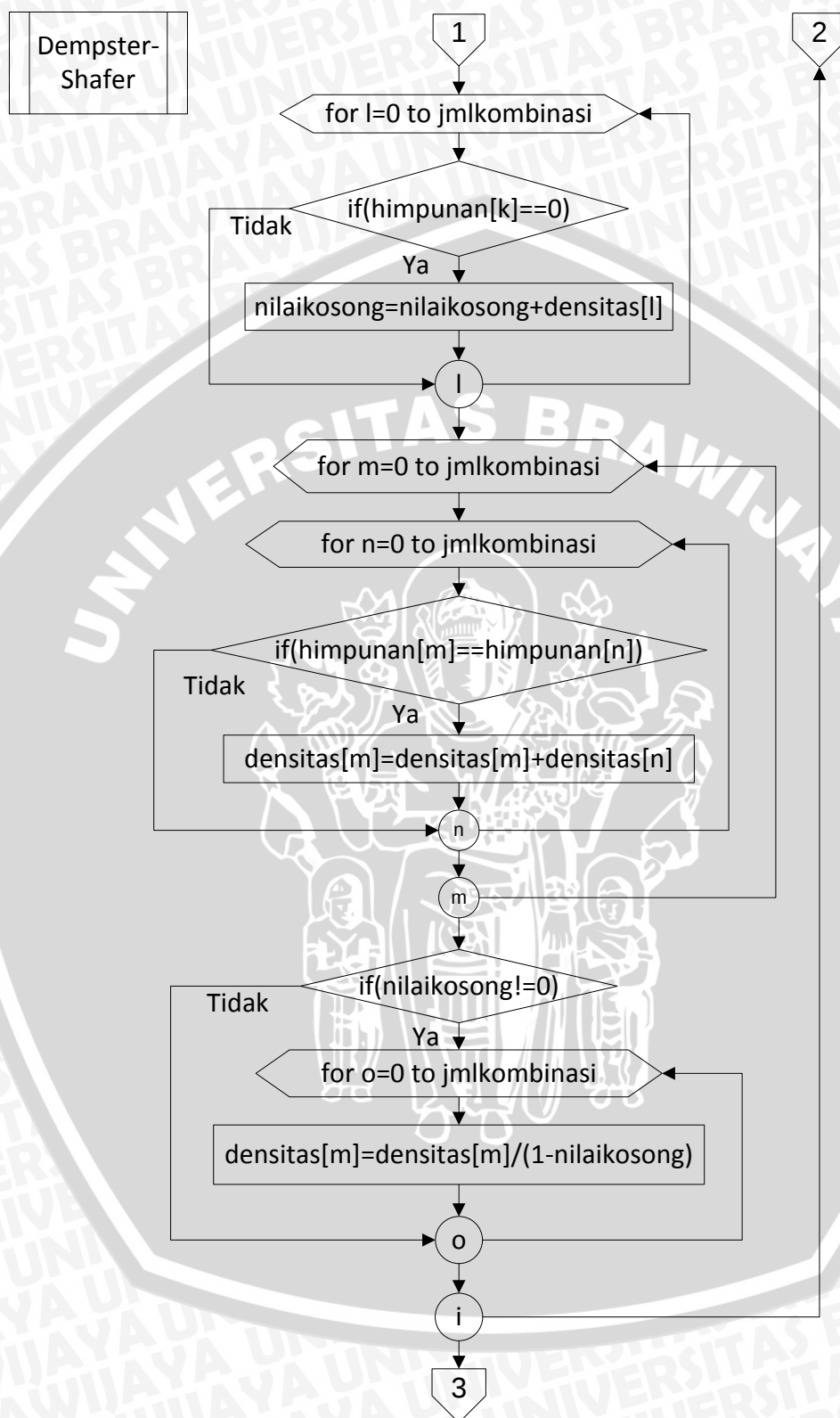


Gambar 4.3 Flowchart Proses Evaluasi

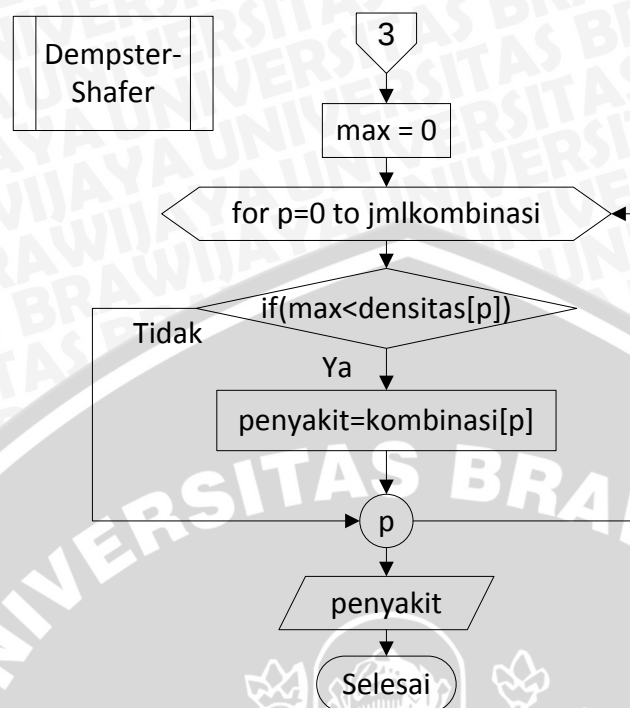
Dalam menghitung *fitness* diperlukan nilai densitas yang sudah dibangkitkan secara acak pada proses sebelumnya. Nilai densitas ini akan digunakan pada perhitungan metode Dempster-Shafer. Dari perhitungan ini akan menghasilkan diagnosa-diagnosa penyakit. selanjutnya diagnosa sistem dicocokkan dengan diagnosa pakar. Langkah terakhir adalah menghitung *fitness* dengan membagi klasifikasi diagnosa yang cocok dengan jumlah data kasus yang digunakan. Perhitungan metode Dempster-Shafer terdiri dari beberapa tahap. Tahapan-tahapan pada metode Dempster-Shafer digambarkan pada Gambar 4.4.



Gambar 4.4 Flowchart Perhitungan Dempster Shafer



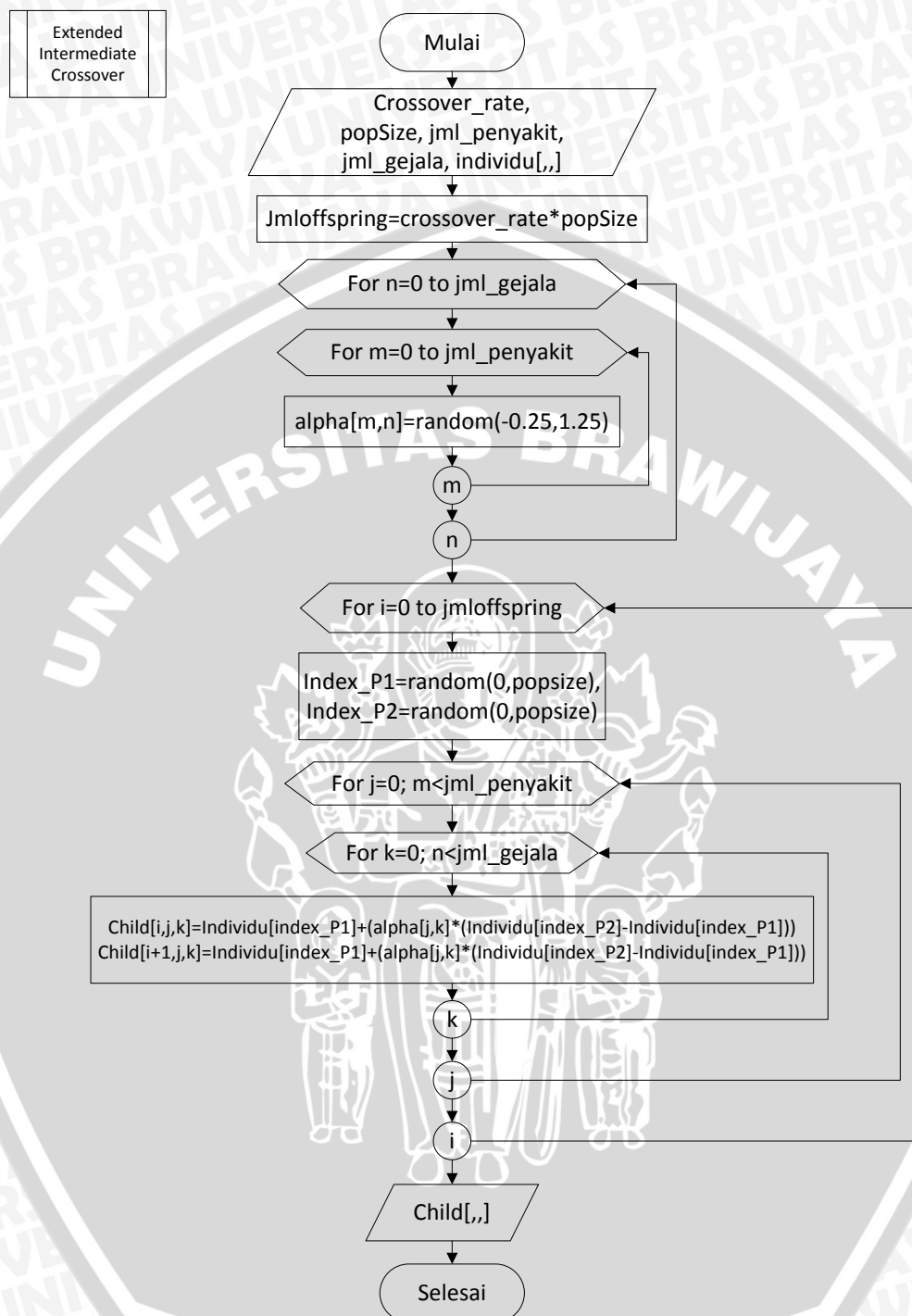
Gambar 4.4 Flowchart Perhitungan Dempster Shafer (lanjutan)



Gambar 4.4 Flowchart Perhitungan Dempster Shafer (lanjutan)

4.2.3 Tukar Silang (Crossover)

Setelah melalui proses evaluasi, individu-individu yang telah dibentuk akan melalui proses reproduksi. Proses reproduksi pertama yaitu *crossover*. Metode *crossover* yang digunakan pada penelitian ini adalah *Extended Intermediate Crossover*. Proses pada metode *Extended Intermediate Crossover* ditunjukkan pada Gambar 4.5.



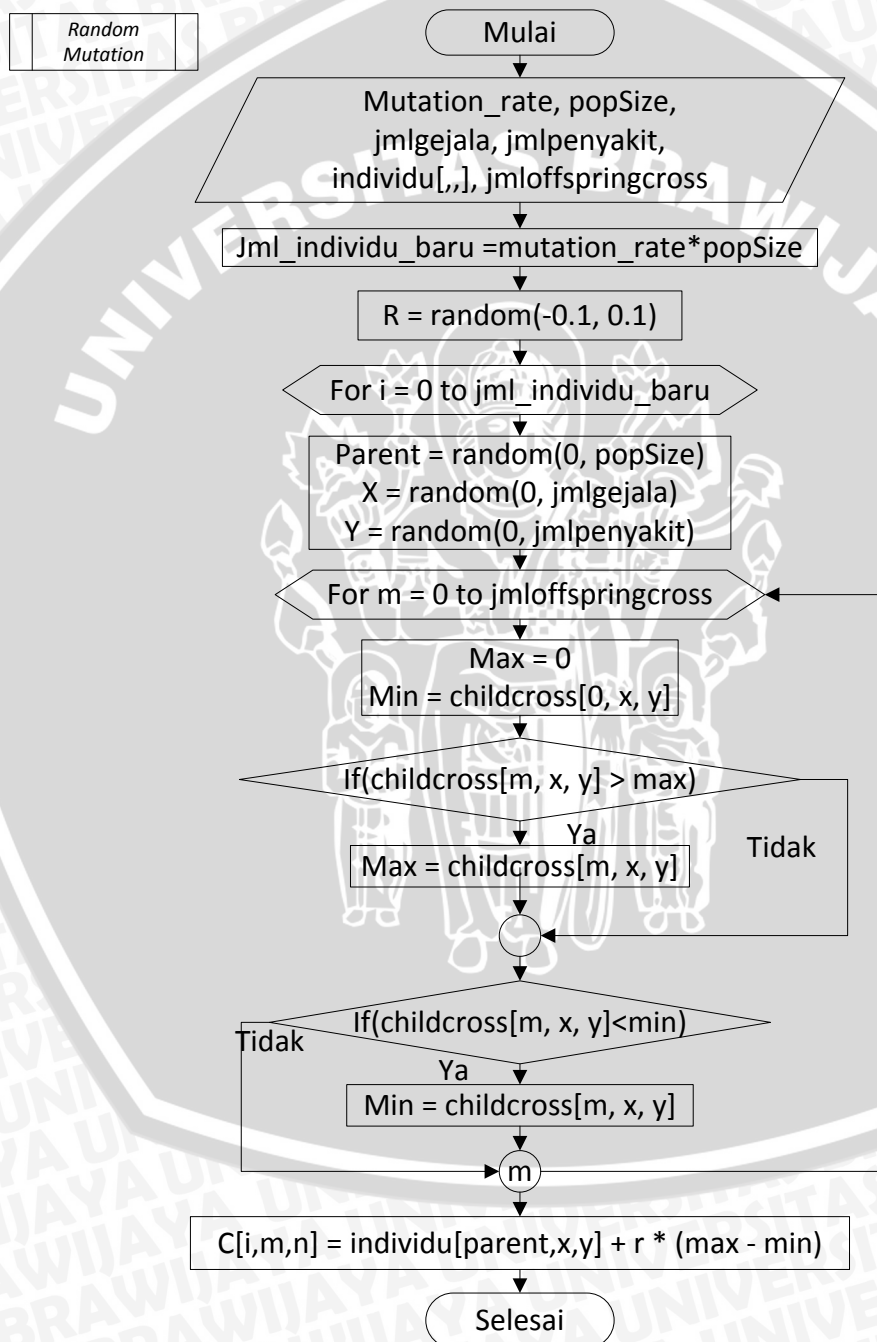
Gambar 4.5 Flowchart Proses Tukar Silang (Crossover)

Pada proses *crossover* ini parameter yang diperlukan yaitu probabilitas *crossover*. Untuk mendapatkan jumlah *offspring* dilakukan perhitungan probabilitas *crossover* (*crossover_rate*) dikalikan dengan ukuran populasi (*popSize*). Langkah selanjutnya membangkitkan α secara random dan

pemilihan dua individu (*parent*). *Offspring* dibangkitkan dengan dua persamaan, yaitu Persamaan 2.13 dan Persamaan 2.14.

4.2.4 Mutasi

Proses reproduksi selanjutnya adalah mutasi. Proses ini biasanya hanya memerlukan satu individu saja. Metode mutasi yang digunakan pada penelitian ini adalah *random mutation*. Proses pada metode *random mutation* ditunjukkan pada Gambar 4.6.

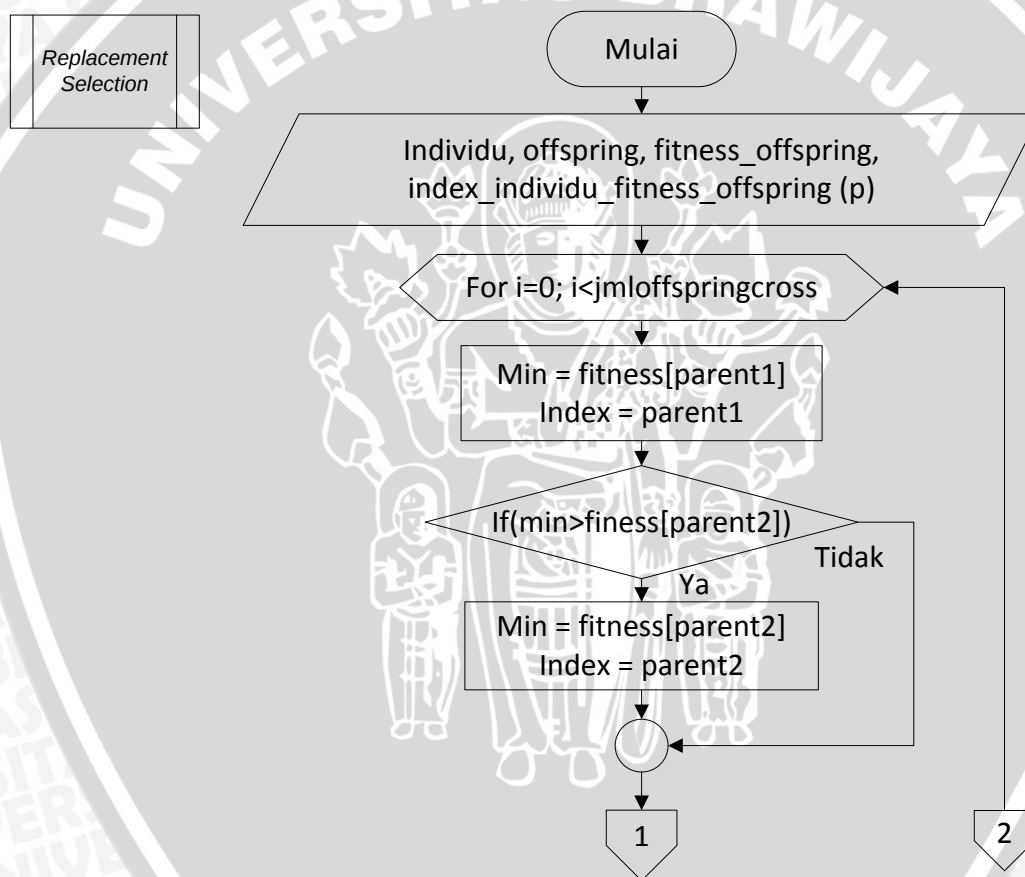


Gambar 4.6 Flowchart Proses Mutasi

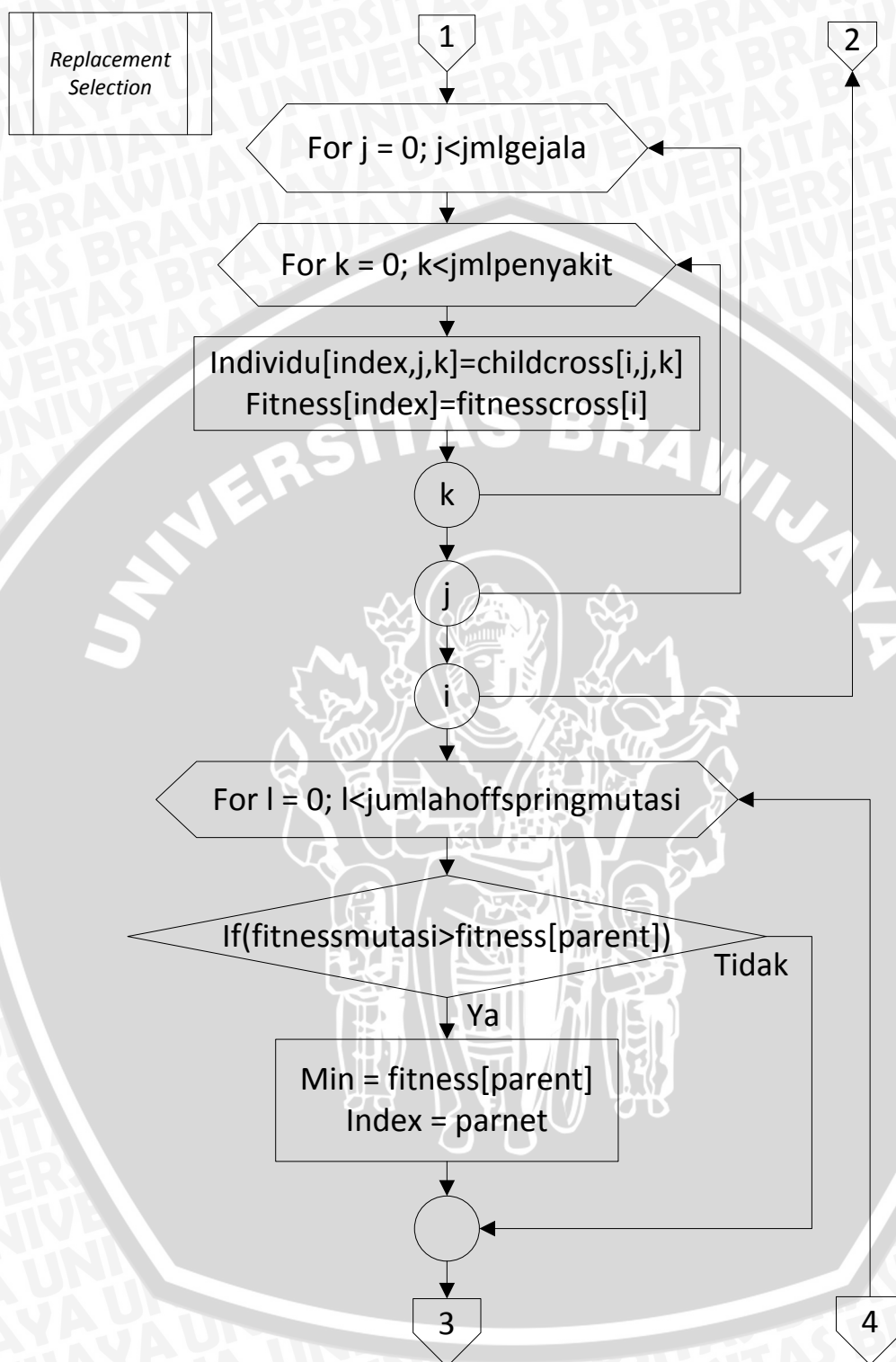
Pada proses mutasi ini parameter yang diperlukan yaitu probabilitas mutasi. Hampir sama dengan *crossover* untuk mendapatkan jumlah *offspring* dilakukan perhitungan probabilitas mutasi (*mutation_rate*) dikalikan dengan ukuran populasi (*popSize*). Selanjutnya membangkitkan *r* secara random dan memilih satu individu (*parent*). Selanjutnya *offspring* dibangkitkan dengan persamaan, yaitu Persamaan 2.15.

4.2.5 Seleksi

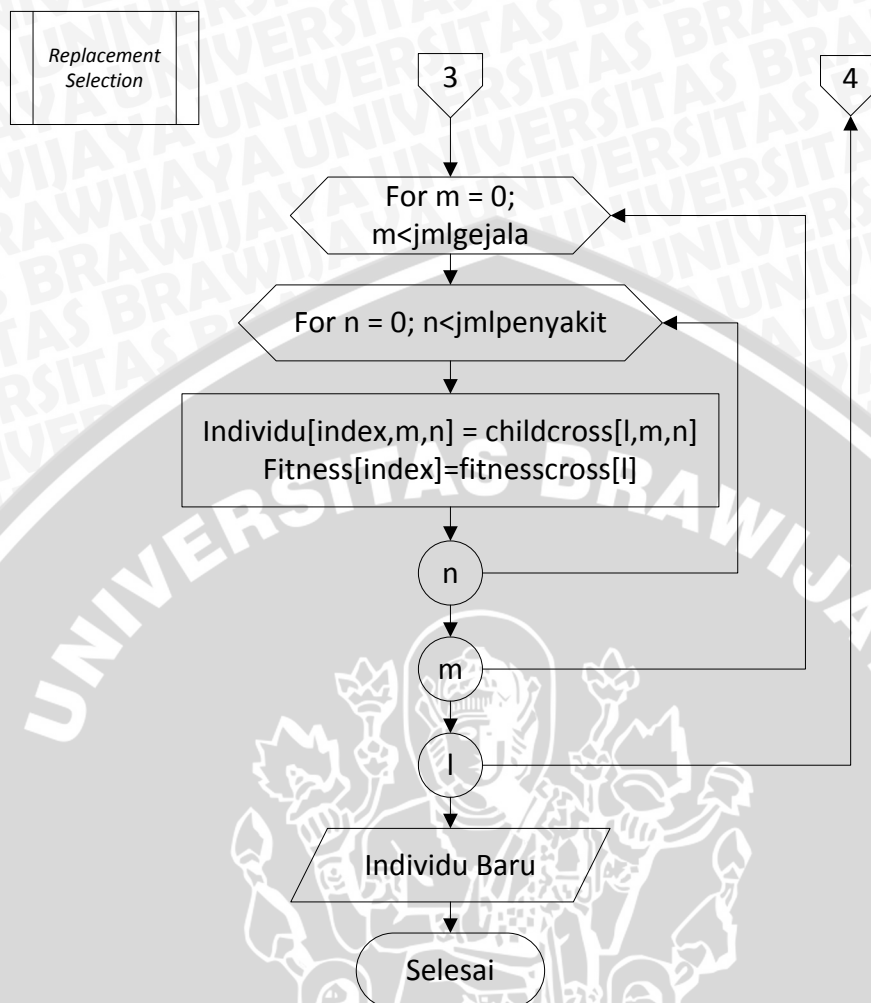
Setelah melewati proses reproduksi (*crossover* dan mutasi), proses selanjutnya adalah proses seleksi. Proses ini akan menghasilkan generasi baru. Seleksi yang digunakan pada penelitian ini adalah *Replacement Selection*. Proses pada metode *Replacement Selection* ditunjukkan pada Gambar 4.7.



Gambar 4.7 Flowchart Proses Seleksi



Gambar 4.7 Flowchart Proses Seleksi (lanjutan)



Gambar 4.7 Flowchart Proses Seleksi (lanjutan)

Proses seleksi ini individu dan offspring yang memiliki *fitness* terbaik. *Fitness offspring* akan dibandingkan dengan *fitness parent* (induk) yang membentuknya. Jika *fitness offspring* lebih besar dari *fitness parent*, maka *offspring* akan menggantikan *parent* yang memiliki *fitness* paling kecil.

4.3 Perhitungan Manual

Subbab perhitungan manual menjelaskan tentang perhitungan pada sistem secara manual dan memberikan gambaran umum tentang perancangan sistem. Contoh manualisasi ini menggunakan ukuran populasi = 5, probabilitas crossover = 0,8 dan probabilitas mutasi = 0,2.

4.3.1 Inisialisasi Populasi Awal

Hal pertama yang dilakukan adalah membangkitkan secara random populasi awal. Pada perhitungan manual ini menggunakan popSize=5. Tabel 4.4 merupakan contoh salah satu individu dari 5 individu dalam populasi.

Tabel 4.4 Individu 1

	A	BR	M	TB	BL	PE	JD	S	BT	RK	V	N	K	SA	DT
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0.702	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0.738	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	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
⋮															
179	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Tabel 4.4 Individu 1 (lanjutan)

	BV	MC	MK	HV	R	AS	RP	D	IS	KB	AR	SR	C	KM	B
1	0	0	0	0	0	0.817	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0.749	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	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	0	0	0
6	0	0	0	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	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0.778	0.859	0	0.801
10	0	0	0	0	0	0	0	0	0.523	0.491	0	0	0.443	0	0
⋮															
179	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.843

4.3.2 Perhitungan *Fitness*

Setelah didapatkan populasi awal, proses selanjutnya adalah menghitung *fitness*. Proses perhitungan *fitness* menggunakan perhitungan pada persamaan 2.12.

Klasifikasi yang benar didapatkan dari proses deteksi penyakit sapi menggunakan Dempster-Shafer dengan menggunakan individu pada populasi. Berikut ini adalah salah satu perhitungan kasus dari 100 kasus yang ada.

- Gejala 1 : Nafsu makan menurun

Langkah awal setelah mengetahui gejala yang dimasukkan yaitu membentuk himpunan penyakit dan menentukan nilai densitasnya. Nilai densitas diambil dari individu yang dihitung nilai *fitness*-nya, contoh untuk gejala 1 nafsu makan menurun (gejala nomor 105), pada tabel nilai untuk gejala 105 pada penyakit Antraks 0,708. Berikut ini daftar penyakit dan nilai densitasnya:

- Antrak (A) = 0.708
- Brucellosis (BR) = 0.444
- Mastitis (M) = 0.44
- Salmonelosis (S) = 0.72
- Ngorok (N) = 0.467
- Kelumpuhan (K) = 0.479
- Sindrom Sapi Ambruk (SA) = 0.597
- Demam 3 Hari (DT) = 0.549
- Diare Ganas Menahun (BV) = 0.427
- Malignan Chataral Fever (MC) = 0.592
- Penyakit Mulut dan Kuku (MK) = 0.456
- Bovine Herpes Virus (HV) = 0.468
- Ringworm (R) = 0.462
- Retensi Plasenta (RP) = 0.565
- Indigesti Sederhana (IS) = 0.808
- Kembung (KB) = 0.506
- Asidosis Rumen (AR) = 0.703
- Surra (SR) = 0.586
- Cacingan (C) = 0.804
- Belatungan (B) = 0.412

Untuk m_1 nilai densitas yang dipilih adalah yang tertinggi, maka

$$m_1\{A, BR, M, S, N, K, SA, DT, BV, MC, MK, HV, R, RP, IS, KB, AR, SR, C, B\} = 0.808$$

$$m_1\{\emptyset\} = 0.202$$

- Gejala 2 : Lesu

Gejala ini merupakan gejala dari penyakit berikut dengan densitas:

- $A = 0.544$
- $S = 0.862$
- $N = 0.445$
- $DT = 0.865$
- $MC = 0.401$
- $IS = 0.714$
- $KB = 0.418$
- $AR = 0.728$
- $SR = 0.514$
- $C = 0.865$

Untuk m_1 nilai densitas yang dipilih adalah yang tertinggi, maka

$$m_2\{A,S,N,DT,MC,IS,KB,AR,SR,C\} = 0.865$$

$$m_2\{\emptyset\} = 1 - 0.865 = 0.135$$

Selanjutnya menghitung nilai densitas baru untuk kombinasi dengan fungsi densitas m_3 yang dapat dilihat pada Tabel 4.5.

Tabel 4.5 Aturan kombinasi untuk m_3

m_1	m_2			
	$\{A,S,N,DT,M,C,IS,KB,AR,S,R,C\}$		$\{\emptyset\}$	
$\{A,BR,M,S,N,K,SA,DT,BV,MC,MK,HV,R,RP,IS,KB,AR,SR,C,B\}$	0.808	0.865	$\{A,BR,M,S,N,K,SA,DT,BV,MC,MK,HV,R,RP,IS,KB,AR,SR,C,B\}$	0.109
$\{\emptyset\}$	0.192	0.166	$\{\emptyset\}$	0.026

Sehingga dapat dihitung

$$m_3\{A,S,N,DT,MC,IS,KB,AR,SR,C\} = \frac{0.7}{1-0} = 0.7$$

$$m_3\{A,BR,M,S,N,K,SA,DT,BV,MC,MK,HV,R,RP,IS,KB,AR,SR,C,B\} = \frac{0.109}{1-0} = 0.109$$

$$m_3\{\emptyset\} = \frac{0.026}{1-0} = 0.026$$

- Gejala 3 : Malas bergerak

Gejala ini merupakan gejala dari penyakit berikut dengan densitas:

- $M=0.517$
- $MK = 0.472$
- $IS = 0.732$
- $AR = 0.784$

Untuk m_1 nilai densitas yang dipilih adalah yang tertinggi, maka

$$m_2\{A,S,N,DT,MC,IS,KB,AR,SR,C\} = 0.784$$

$$m_2\{\emptyset\} = 0.216$$

Tabel 4.6 Aturan kombinasi untuk m_5

m3		m4			
		{M,MK,IS,A R}		{ $\emptyset$ }	
{A,S,N,DT,MC,IS, KB,AR,SR,C}	0.865	{IS,AR}	0.678	{A,S,N,DT,MC,IS,KB, AR,SR,C}	0.187
{A,BR,M,S,N,K,SA ,DT,BV,MC,MK,H V,R,RP,IS,KB,AR,S R,C,B}	0.109	{M,MK,IS,A R}	0.085	{A,BR,M,S,N,K,SA,D T,BV,MC,MK,HV,R, RP,IS,KB,AR,SR,C,B}	0.024
{ $\emptyset$ }	0.026	{M,MK,IS,A R}	0.02	{ $\emptyset$ }	0.006

$$m_5\{IS, AR\} = \frac{0.678}{1 - 0} = 0.678$$

$$m_5\{M, MK, IS, AR\} = \frac{0.085 + 0.02}{1 - 0} = 0.106$$

$$m_5\{A, S, N, DT, MC, IS, KB, AR, SR, C\} = \frac{0.187}{1 - 0} = 0.187$$

$$m_5\{A, BR, M, S, N, K, SA, DT, BV, MC, MK, HV, R, RP, IS, KB, AR, SR, C, B\} = \frac{0.024}{1 - 0} = 0.024$$

$$m_5\{\emptyset\} = \frac{0.006}{1 - 0} = 0.006$$

Dari hasil perhitungan dengan metode Dempster-Shafer, nilai densitas paling tinggi adalah penyakit Indigesti Sederhana dan Asidosis Rumen.

Setelah semua kasus sudah dihitung selanjutnya adalah menghitung nilai akurasi yang digunakan sebagai nilai fitness. Berikut ini pada Tabel 4.7 adalah nilai fitness dari salah satu individu dari 5 individu.

Tabel 4.7 Hasil Akurasi Individu 1

NO	PAKAR	SISTEM	Akurasi Hasil
1	C	{IS,C}	1
2	DT	{DT}	1
3	C	{C}	1
4	IS	{IS,AR}	1
5	B	{B}	1
6	C	{C}	1
7	DT,IS,C	{IS}	1
8	IS	{IS,AR}	1
9	M	{M}	1
10	DT	{C}	0
11	IS	{IS,AR}	1
12	RHINITIS	{MC}	0
13	KB	{KB}	1
14	C	{C}	1
15	M	{M}	1
16	IS	{IS}	1
17	SA	{SA}	1
18	IS	{IS}	1

$$fitness = \frac{16}{18} = 0.889$$

Perhitungan *fitness* ini dilakukan terhadap 4 individu lainnya. Hasil perhitungan *fitness* ditunjukkan pada Tabel 4.8.

Tabel 4.8 Hasil *Fitness* 5 Individu

No.	Individu ke-	Fitness
1	P ₁	0.889
2	P ₂	0.944
3	P ₃	0.889
4	P ₄	0.944
5	P ₅	0.944

4.3.3 Crossover

Hal pertama yang dilakukan pada proses *crossover* adalah menghitung jumlah offspring yang akan dihasilkan. Pada perhitungan ini *crossover_rate* yang digunakan adalah 0,8 dengan *popSize* sejumlah 5.

$$\text{Jumlah offspring} = 0,8 \times 5 = 4$$

Sehingga pada perhitungan ini jumlah *offspring* yang dihasilkan adalah 4. Selanjutnya adalah membangkitkan α dengan range (-0,25; 1,25). Tabel 4.9 menunjukkan α yang telah dibangkitkan pada proses *crossover* Individu 1 dan Individu 2.

Tabel 4.9 Tabel Alpha

	A	BR	M	TB	BL	PE	JD	S	BT	RK	V	N	K	SA	DT
1	-0.048	0.674	1.244	-0.168	0.073	0.655	1.107	0.961	0.656	0.554	0.146	0.127	0.17	0.801	0.286
2	0.774	0.715	1.173	1.142	-0.203	0.131	0.105	0.25	-0.196	0.359	1.224	0.094	0.501	1.22	0.629
3	0.267	-0.208	0.156	0.134	-0.033	0.317	-0.141	-0.014	1.188	1.238	1.216	0.116	0.42	0.496	0.591
4	0.76	0.118	-0.029	-0.143	1.2	-0.036	-0.191	0.275	0.764	0.037	0.473	0.954	-0.177	1.207	1.022
5	1.157	1.227	0.277	1.035	1.16	1.238	0.939	1.167	-0.119	0.9	1.23	-0.014	0.924	-0.045	1.084
6	0.296	0.013	0.042	0.719	0.137	-0.216	0.198	-0.014	0.489	0.741	-0.177	0.026	0.078	0.274	0.818
7	0.315	1.112	0.5	0.318	0.248	0.138	0.537	0.34	0.102	-0.192	0.236	0.722	0.823	1.086	0.835
8	0.921	0.569	-0.125	1.158	0.596	0.207	0.039	0.818	0.64	0.16	0.236	0.911	0.106	0.001	0.128
9	0.41	1.117	0.11	1.21	1.198	-0.226	0.669	0.736	-0.22	0.312	1.01	-0.153	0.064	0.628	0.89
10	0.94	0.508	0.412	-0.236	-0.095	0.002	0.211	-0.044	-0.23	-0.014	0.867	1.232	0.61	1.085	0.565
⋮															
179	1.238	1.056	-0.018	0.762	0.309	0.144	0.931	0.234	0.83	-0.196	0.584	0.335	0.283	0.414	0.459

Tabel 4.9 Tabel Alpha (lanjutan)

	BV	MC	MK	HV	R	AS	RP	D	IS	KB	AR	SR	C	KM	B
1	1.16	-0.097	1.215	0.436	0.311	0.035	-0.213	0.475	-0.071	0.041	-0.182	0.04	0.174	1.145	-0.109
2	0.314	1.163	0.308	1.082	0.691	0.265	0.986	0.201	0.532	1.175	-0.208	0.729	0.369	0.241	0.973
3	0.819	1.194	0.212	0.014	0.757	-0.034	0.196	1.172	-0.058	-0.229	0.416	0.716	0.04	1.085	0.899
4	0.826	-0.139	0.456	-0.225	0.184	1.046	0.438	0.502	0.757	-0.117	1.017	0.306	0.026	0.067	0.803
5	0.186	0.901	1.03	0.159	1.076	0.782	0.556	0.613	-0.032	-0.233	0.4	0.397	0.846	0.522	0.627
6	0.533	0.374	1.146	-0.083	-0.205	0.418	-0.028	0.507	0.863	-0.172	0.944	0.774	-0.181	0.934	-0.149
7	1.145	-0.214	0.952	0.599	0.118	-0.093	0.004	0.785	0.667	0.422	0.075	0.784	1.175	0.801	-0.151
8	0.008	0.94	-0.164	0.885	0.864	0.423	0.2	-0.046	0.611	0.047	-0.126	0.263	0.397	1.231	0.515
9	0.404	1.245	1.123	-0.033	1.022	0.66	-0.205	1.198	0.914	1.035	1.225	0.233	0.492	0.57	0.105
10	1.105	0.931	1.097	0.409	0.742	1.065	1.172	-0.016	0.492	0.27	0.718	0.61	0.942	-0.218	1.178
⋮															
179	0.342	0.968	0.023	0.584	0.003	0.544	0.908	0.887	0.725	0.169	0.271	1.054	1.156	0.888	0.457

Setelah mendapatkan alpha, selanjutnya adalah membangkitkan *offspring* dengan persamaan Persamaan 2.13 dan Persamaan 2.14. Tabel 4.10 menunjukkan salah satu *offspring* hasil *crossover* antara individu 1 dengan individu 2.

Tabel 4.10 Offspring Hasil Crossover Individu 1 dan Individu 2

	A	BR	M	TB	BL	PE	JD	S	BT	RK	V	N	K	SA	DT
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0.942	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0

	A	BR	M	TB	BL	PE	JD	S	BT	RK	V	N	K	SA	DT
7	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0.719	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	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
⋮															
179	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Tabel 4.10 Offspring Hasil Crossover Individu 1 dan Individu 2 (lanjutan)

	BV	MC	MK	HV	R	AS	RP	D	IS	KB	AR	SR	C	KM	B
1	0	0	0	0	0	0.815	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0.75	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	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	0	0	0
6	0	0	0	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	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0.781	0.847	0	0.807
10	0	0	0	0	0	0	0	0	0.538	0.52	0	0	0.495	0	0
⋮															
179	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.804

4.3.4 Mutasi

Hal pertama yang dilakukan pada proses mutasi adalah menghitung jumlah *offspring* yang akan dihasilkan. Pada perhitungan ini *mutation_rate* yang digunakan adalah 0,2 dengan *popSize* sejumlah 5.

$$\text{Jumlah offspring} = 0,2 \times 5 = 1$$

Sehingga pada perhitungan ini jumlah *offspring* yang dihasilkan adalah 1. Selanjutnya membangkitkan *r* dengan range (-0,1; 0,1). Pada perhitungan ini *r* yang digunakan adalah 0.068. Selanjutnya adalah membangkitkan *offspring* dengan Persamaan 2.15. Tabel 4.11 menunjukkan *offspring* yang dihasilkan dari proses mutasi individu 3.

Tabel 4.11 Offspring Hasil Mutasi Individu 3

	A	BR	M	TB	BL	PE	JD	S	BT	RK	V	N	K	SA	DT
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0.85	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0.726	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	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
⋮															
179	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Tabel 4.11 Offspring Hasil Mutasi Individu 3 (lanjutan)

	BV	MC	MK	HV	R	AS	RP	D	IS	KB	AR	SR	C	KM	B
1	0	0	0	0	0	0.732	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0.758	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	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	0	0	0
6	0	0	0	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	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0.843	0.737	0	0.755
10	0	0	0	0	0	0	0	0	0.461	0.558	0	0	0.587	0	0
⋮															
179	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.889

4.3.5 Seleksi

Setelah melakukan proses reproduksi (*crossover* dan mutasi), proses selanjutnya adalah proses seleksi. Pada perhitungan ini menggunakan $\text{popSize} = 5$, sehingga terdapat 5 individu. Tabel 4.12 menunjukkan individu dengan *fitness*-nya.

Tabel 4.12 Himpunan Populasi yang Menjadi Parent

No.	Individu ke-	<i>Fitness</i>
1	P ₁	0.889
2	P ₂	0.944
3	P ₃	0.889
4	P ₄	0.944
5	P ₅	0.944

Dari proses *crossover* dihasilkan 4 *offspring* (C₁, C₂, C₃ dan C₄) dan proses mutasi menghasilkan 1 *offspring* (C₅). Sehingga didapatkan 5 *offspring* seperti yang ditunjukkan Tabel 4.13.

Tabel 4.13 Offspring Beserta Parent dan Fitness-nya

No.	Individu	Parent	<i>Fitness</i>
1	C ₁	P ₁ dan P ₂	0.944
2	C ₂	P ₁ dan P ₂	0.944
3	C ₃	P ₄ dan P ₅	0.944
4	C ₄	P ₄ dan P ₅	0.944
5	C ₅	P ₃	0.889

Selanjutnya *fitness offspring* dibandingkan dengan *fitness parent*, jika ada *offspring* yang memiliki *fitness* yang lebih besar dari *fitness parent*, maka *offspring* akan menggantikan *parent* yang memiliki *fitness* terkecil. Hasil dari proses seleksi pada perhitungan ini ditunjukkan pada Tabel 4.14.

Tabel 4.14 Individu yang Masuk ke Generasi Berikutnya

No.	P _(t+1)	Asal P _(t)	<i>Fitness</i>
1	P ₁	C ₁	0.944
2	P ₂	P ₂	0.944
3	P ₃	P ₃	0.889
4	P ₄	C ₃	0.944
5	P ₅	P ₅	0.944

Pada Tabel 4.13 C₁ memiliki *fitness* lebih baik dari P₁, begitu juga C₃ memiliki *fitness* yang lebih baik dari P₄, sehingga C₁ menggantikan P₁ dan C₃ menggantikan P₄. Sedangkan *fitness offspring* lain tidak lebih baik dari *fitness parent*, sehingga P₂, P₃ dan P₅ tetap. Sehingga generasi baru yang dihasilkan ditunjukkan pada Tabel 4.15.

Tabel 4.15 Individu pada Generasi Baru

No.	Individu	Fitness
1	P ₁	0.944
2	P ₂	0.944
3	P ₃	0.889
4	P ₄	0.944
5	P ₅	0.944

4.4 Perancangan Antarmuka

Antarmuka (*interface*) digunakan pengguna untuk melihat informasi yang ada pada sistem. Informasi yang bisa didapat dari sistem yaitu proses yang terdapat pada algoritma genetika, nilai densitas optimal yang dihasilkan dan hasil akurasi.

Sebelum melakukan proses optimasi, pengguna harus mengisi kolom parameter algoritma genetika yang berada di bawah nama sistem. Parameter yang dimasukkan adalah probabilitas *crossover*, probabilitas mutasi, ukuran populasi, dan jumlah generasi.

Implementasi Algoritma Genetika Untuk Optimasi Nilai Densitas Metode *Dempster-Shafer* pada Deteksi Penyakit Sapi

Probabilitas Crossover

Ukuran Populasi

Probabilitas Mutasi

Jumlah Generasi

PROSES

Proses Algoritma Genetika	Nilai Densitas Optimal	Hasil Akurasi

Gambar 4.8 Antarmuka Proses Algoritma Genetika

Pada Gambar 4.8 menunjukkan tab Proses Algoritma Genetika yang sedang dibuka. Pada tampilan ini menunjukkan informasi yang dihasilkan proses genetika pada generasi terakhir seperti hasil *crossover*, mutase, dan seleksi.

Implementasi Algoritma Genetika Untuk Optimasi Nilai Densitas Metode *Dempster-Shafer* pada Deteksi Penyakit Sapi

Probabilitas Crossover Ukuran Populasi

Probabilitas Mutasi Jumlah Generasi

Proses Algoritma Genetika	Nilai Densitas Optimal	Hasil Akurasi

Gambar 4.9 Antarmuka Nilai Densitas Optimal

Pada Gambar 4.9 menunjukkan tab Nilai Densitas Optimal yang sedang dibuka. Pada tampilan ini berisi nilai densitas optimal yang dihasilkan oleh sistem.

Implementasi Algoritma Genetika Untuk Optimasi Nilai Densitas Metode *Dempster-Shafer* pada Deteksi Penyakit Sapi

Probabilitas Crossover Ukuran Populasi

Probabilitas Mutasi Jumlah Generasi

Proses Algoritma Genetika	Nilai Densitas Optimal	Hasil Akurasi

Gambar 4.10 Antarmuka Hasil Akurasi

Pada Gambar 4.10 menunjukkan tab Hasil Akurasi yang sedang dibuka. Pada tampilan ini berisi informasi mengenai kecocokan diagnosa yang dihasilkan sistem dengan diagnosa pakar dan nilai akurasi yang dihasilkan dengan nilai densitas optimal.

BAB 5 IMPLEMENTASI

Bab ini menjabarkan mengenai implementasi perangkat lunak berdasarkan hasil yang diperoleh dari proses perancangan dan analisis kebutuhan yang telah dibuat. Implementasi perangkat lunak yang akan dijabarkan terdiri dari implementasi spesifikasi sistem, batasan implementasi dan implementasi program.

5.1 Spesifikasi Sistem

Analisis kebutuhan dan perancangan yang dijelaskan pada Bab 4 dijadikan acuan untuk implementasi sistem, sehingga sistem berfungsi sesuai dengan kebutuhan. Spesifikasi perangkat keras dan spesifikasi perangkat lunak yang digunakan pada implementasi sistem adalah sebagai berikut.

5.1.1 Spesifikasi Perangkat Keras

Implementasi sistem menggunakan sebuah komputer dengan spesifikasi perangkat keras yang dijabarkan pada Tabel 5.1.

Tabel 5.1 Spesifikasi Perangkat Keras

Komponen	Spesifikasi
Processor	Intel(R) Core i5
Memori (RAM)	4 GB RAM
Kartu Grafis	Intel Graphics Media Accelerator HD
Harddisk	500 GB HDD
Layar LCD Monitor	13-inch

5.1.2 Spesifikasi Perangkat Lunak

Implementasi sistem menggunakan sebuah komputer dengan spesifikasi perangkat lunak yang dijabarkan pada Tabel 5.2.

Tabel 5.2 Spesifikasi Perangkat Lunak

Komponen	Spesifikasi
Sistem Operasi	Microsoft Wondows 7 Home Premium (64-bit)
Bahasa Pemrograman	C#
Tools Pemrograman	Microsoft Visual Studio 2010

5.2 Implementasi Sistem

Berdasarkan perancangan yang diuraikan pada Bab 4, selanjutnya pada subbab ini akan dibahas mengenai implementasi sistem sesuai dengan perancangan yang telah dibuat. Sistem diimplementasikan dengan menggunakan bahasa pemrograman C#.

5.2.1 Implementasi Inisialisasi Populasi Awal

Pada proses inisialisasi populasi awal, kromosom tiap individu dibangkitkan dengan cara acak namun menggunakan syarat yaitu:

- jika densitas yang diberikan oleh pakar berada pada range 0,1-0,3, maka gen diacak di antara range 0,1-0,3,
- jika densitas yang diberikan oleh pakar berada pada range 0,4-0,6, maka gen diacak di antara range 0,4-0,6,
- jika densitas yang diberikan oleh pakar berada pada range 0,7-0,9, maka gen diacak di antara range 0,7-0,9,
- jika densitas yang diberikan adalah 1 maka gen ditetapkan 1, dan
- jika densitas yang diberikan adalah 0, maka gen akan ditetapkan bernilai 0.

Pada sistem kode untuk inisialisasi populasi awal dapat dilihat pada *Source code* 5.1. Syarat tersebut dilakukan sistem seperti pada baris kode ke-9 sampai dengan baris kode ke-28

```
static void inisialisasiawal()
1.  random random = new Random();
2.  individu = new double[popsize, jmlgejala, jmlpenyakit];
3.  for (int i = 0; i < popsize; i++)
4.  {
5.      for (int j = 0; j < jmlgejala; j++)
6.      {
7.          for (int k = 0; k < jmlpenyakit; k++)
8.          {
9.              if (aturan[j, k] >= 0.1 && aturan[j, k] <= 0.3)
10.             {
11.                 individu[i, j, k] = random.NextDouble() * (0.3 - 0.1) +
12.                 0.1;
13.             }
14.             else if (aturan[j, k] >= 0.4 && aturan[j, k] <= 0.6)
15.             {
16.                 individu[i, j, k] = random.NextDouble() * (0.6 - 0.4) +
17.                 0.4;
18.             }
19.             else if (aturan[j, k] >= 0.7 && aturan[j, k] <= 0.9)
20.             {
21.                 individu[i, j, k] = random.NextDouble() * (0.9 - 0.7) +
22.                 0.7;
23.             }
24.             else if (aturan[j, k] == 1)
25.             {
26.                 individu[i, j, k] = 1;
27.             }
28.             else
29.             {
30.                 individu[i, j, k] = 0;
31.             }
32.         }
33.     }
34. }
```

```

26.     {
27.         individu[i, j, k] = 0;
28.     }
29.     }
30. }
31. }

```

Source Code 5.1 Inisialisasi Populasi Awal

5.2.2 Implementasi *Fitness*

Proses perhitungan *fitness* pada penelitian ini dilakukan dengan menggunakan perhitungan akurasi dalam mendiagnosa penyakit dengan menggunakan individu yang telah dibangkitkan sebagai nilai densitas. Metode yang digunakan untuk mendiagnosa penyakit sapi adalah Metode Dempster-Shafer Nilai *fitness* yang akan digunakan dalam menentukan baik tidaknya individu.

Perhitungan *fitness* dibagi ke dalam dua fungsi, fungsi pertama bertugas untuk menghitung kecocokan diagnosa sistem dengan diagnosa pakar. Perhitungan *fitness* dapat dilihat pada *source code* 5.2 Setiap kasus akan didiagnosa oleh sistem dengan memanggil fungsi ds() (fungsi Dempster-Shafer) yang ditunjukkan pada baris kode ke-11. Setelah kasus sudah didiagnosa, diagnosa sistem dengan diagnosa pakar akan dicocokkan seperti yang ditunjukkan baris kode ke-13 sampai dengan baris kode ke-27. Nilai *fitness* didapat dari jumlah data yang cocok dibagi dengan jumlah data kasus seperti yang ditunjukkan baris kode ke-29.

```

static void evaluasi()
1.  double semua = penyakitdatalatih.GetLength(0);
2.  double cocok;
3.  int kecocokan;
4.  for (int i = 0; i < popsize; i++)
5.  {
6.      //0 : perhitungan dampster shafer untuk evaluasi populasi
        awal
7.      cocok = 0;
8.      for (int j = 0; j < penyakitdatalatih.GetLength(0); j++)
9.      {
10.         penyakit_awal = new int[jmlpenyakit];
11.         ds(kategorievaluasi, i, j);
12.         //menghitung kecocokan sistem dengan pakar
13.         kecocokan = 0;
14.         for (int k = 0; k < jmlpenyakit; k++)
15.         {
16.             if (penyakit_awal[k] != 0 && penyakitdatalatih[j] != 0)
17.             {
18.                 if (penyakit_awal[k] == penyakitdatalatih[j])
19.                 {
20.                     kecocokan++;
21.                 }
22.             }
23.         }
24.         if(kecocokan != 0)

```

```

25.     {
26.         cocok++;
27.     }
28. }
29. fitness[i] = cocok / semua;
30. }

```

Source Code 5.2 Perhitungan *Fitness*

Perhitungan Dempster-Shafer digunakan untuk menentukan diagnosa penyakit sapi yang nantinya akan digunakan untuk perhitungan *fitness*. Perhitungan metode Dempster-Shefer ditunjukkan pada *Source code* 5.3. Hal pertama yang dilakukan pada proses ini adalah mengambil penyakit-penyakit yang memiliki gejala yang sudah ditentukan ke dalam satu himpunan. Kemudian densitas dari penyakit-penyakit tersebut ditentukan dengan mengambil nilai densitas maksimal diantara penyakit-penyakit dengan gejala yang sama. Proses ini ditunjukkan pada baris kode ke-16 sampai baris kode ke-30. Proses berikutnya adalah menghitung kombinasi antara 2 himpunan penyakit yang sudah ditentukan sebelumnya. Proses ini ditunjukkan baris kode ke-32 sampai baris kode ke-106. Selanjutnya sistem akan memeriksa apakah ada kombinasi yang sama dan apakah ada yang himpunan kosong, untuk kombinasi yang sama akan dijumlahkan dan untuk himpunan kosong akan dijumlahkan. Proses ini ditunjukkan pada baris kode ke-109 sampai baris kode ke-136 untuk menghitung himpunan yang sama dan baris kode ke-137 sampai baris kode ke-210. Setelah mendapatkan kombinasi yang sama dan jumlah himpunan kosong, proses berikutnya adalah perhitungan aturan kombinasi Dempster-Shafer, yaitu Persamaan 2.7 dan Persamaan 2.8 seperti yang dijelaskan pada Bab 4 pada perhitungan manual. Proses ini ditunjukkan pada baris kode ke-211 sampai baris kode ke-218. Setelah semua gejala dihitung, proses terakhir yang dilakukan adalah mengambil himpunan penyakit yang memiliki nilai terbesar. Himpunan penyakit ini yang digunakan sebagai diagnosa penyakit. Proses ini ditunjukkan pada baris kode ke-222 sampai baris kode ke-237.

```

static void ds(int kromosom, int i individu)
1.  himpunanpenyakit = new int[gejaladatalatih.GetLength(1),
    jmlpenyakit];
2.  maxdensitas = new double[gejaladatalatih.GetLength(1)];
3.  kombinasi = new int[gejaladatalatih.GetLength(1) - 1,
    Convert.ToInt32(Math.Pow(2, (gejaladatalatih.GetLength(1) +
    1))), jmlpenyakit]; //2^(n+1)
4.  densitaskombinasi = new double[gejaladatalatih.GetLength(1)
    - 1, Convert.ToInt32(Math.Pow(2,
    (gejaladatalatih.GetLength(1) + 1)))]; //2^(n+1)
5.  teta = new double[gejaladatalatih.GetLength(1)];
6.  int indexkombinasi = 4;
7.  totalgejala = 0;
8.  int index;
9.  for (int k = 0; k < gejalaadatalatih.GetLength(1);
    k++) //jumlah gejala
10. {

```

```

11.  if (gejaladatalatih[no_kasus, k] != 0)
12.  {
13.      index = 0;
14.      kombinasisementara = new int[Convert.ToInt32(Math.Pow(2,
15.          (gejaladatalatih.GetLength(1) + 1))),
16.          jmlpenyakit]; //2^(n+1)
17.      densitaskombinasisementara = new
18.          double[Convert.ToInt32(Math.Pow(2,
19.              (gejaladatalatih.GetLength(1) + 1)))]];
20.      maxdensitas[k] = 0;
21.      for (int l = 0; l < jmlpenyakit; l++)
22.      {
23.          if (individu[i_individu, (gejaladatalatih[no_kasus, k] -
24.              1), l] != 0)
25.          {
26.              himpunanpenyakit[k, l] = 1 + 1; //Himpunan penyakit
27.              if (maxdensitas[k] < individu[i_individu,
28.                  (gejaladatalatih[no_kasus, k] - 1), l])
29.              {
30.                  maxdensitas[k] = individu[i_individu,
31.                      (gejaladatalatih[no_kasus, k] - 1), l]; //Max densitas
32.                      di satu gejala
33.              }
34.          }
35.          //hitung teta
36.          teta[k] = 1 - maxdensitas[k];
37.          //Kombinasi m1.m2
38.          if (k == 1)
39.          {
40.              //m{A}*m{B}
41.              for (int a = 0; a < jmlpenyakit; a++)
42.              {
43.                  if (himpunanpenyakit[0, a] == himpunanpenyakit[1, a])
44.                  {
45.                      kombinasisementara[0, a] = himpunanpenyakit[0, a];
46.                  }
47.              }
48.              densitaskombinasisementara[0] = maxdensitas[0] *
49.                  maxdensitas[1];
50.              index++;
51.              //m{A}*m{teta_B}
52.              for (int a = 1; a < jmlpenyakit; a++) //k dimulai dari 1
53.              {
54.                  kombinasisementara[1, a] = himpunanpenyakit[0, a];
55.              }
56.              densitaskombinasisementara[1] = maxdensitas[0] *
57.                  teta[1];
58.              index++;
59.              //m{teta_A}*m{B}
60.              for (int a = 1; a < jmlpenyakit; a++) //k dimulai dari 1
61.              {
62.                  kombinasisementara[2, a] = himpunanpenyakit[1, a];
63.              }
64.              densitaskombinasisementara[2] = teta[0] *
65.                  maxdensitas[1];

```

```

56.     index++;
57.     //m{teta_A}*m{teta_B}
58.     for (int a = 1; a < jmlpenyakit; a++)//k dimulai dari 1
59.     {
60.         kombinasisementara[3, a] = 32;
61.     }
62.     densitaskombinasisementara[3] = teta[0] * teta[1];
63.     index++;
64. }//end if kombinasi pertama
65. else if (k > 1)
66. {
67.     //k{A(k-2)}*m{B(k)}
68.     for (int b = 0; b < (indexkombinasi - 1); b++)
69.     {
70.         for (int a = 0; a < jmlpenyakit; a++)
71.         {
72.             if (kombinasi[(k - 2), b, a] == himpunanpenyakit[k,
73.                 a])
74.             {
75.                 kombinasisementara[b, a] = himpunanpenyakit[k, a];
76.             }
77.             densitaskombinasisementara[b] = densitaskombinasi[(k -
78.                 2), b] * maxdensitas[k];
79.         }
80.         index = index + (indexkombinasi - 1);
81.         //k{A(k-2)}*m{teta_B(k)}
82.         for (int b = 0; b < (indexkombinasi - 1); b++)
83.         {
84.             for (int a = 1; a < jmlpenyakit; a++)//k dimulai dari 1
85.             {
86.                 kombinasisementara[(index + b), a] = kombinasi[(k -
87.                     2), b, a];
88.             }
89.             densitaskombinasisementara[index + b] =
90.                 densitaskombinasi[(k - 2), b] * teta[k];
91.         }
92.         index = index + (indexkombinasi - 1);
93.         //k{teta_A(k-2)}*m{B(k)}
94.         for (int a = 1; a < jmlpenyakit; a++)//k dimulai dari 1
95.         {
96.             kombinasisementara[index, a] = himpunanpenyakit[k, a];
97.         }
98.         densitaskombinasisementara[index] = densitaskombinasi[(k
99.             - 2), indexkombinasi - 1] * maxdensitas[k];
100.        index++;
101.        //k{teta_A(k-2)}*m{teta_B(k)}
102.        for (int a = 1; a < jmlpenyakit; a++)//k dimulai dari 1
103.        {
104.            kombinasisementara[index, a] = 32;
105.        }
106.        densitaskombinasisementara[index] = densitaskombinasi[(k
107.            - 2), indexkombinasi - 1] * teta[k];
108.        index++;
109.    }//if setelah m1*m2
110.    int[] hapus = new int[index * index];

```

```

106. int indexhapus = 0;
107. //CEK HIMPUNAN KOSONG
108. double nilaikosong = 0;
109. int empty;
110. int himpunankosongdiawal = 0;
111. for (int b = 0; b < ((index / 2) - 1); b++)
112. {
113.     //Cek apakah kosong semua
114.     empty = 0;
115.     for (int c = 0; c < jmlpenyakit; c++)
116.     {
117.         if (kombinasisementara[b, c] == 0)
118.         {
119.             empty++;
120.         }
121.     }
122.     //Jika Kosong
123.     if (empty == jmlpenyakit)
124.     {
125.         nilaikosong = nilaikosong + densitaskombinasi[k - 1,
126.         b];
127.         hapus[indexhapus] = b;
128.         indexhapus++;
129.         if (b == 0)
130.         {
131.             himpunankosongdiawal = 1;
132.         }
133.     }
134.     indexkombinasi = 0;
135.     //CEK HIMPUNAN YANG SAMA
136.     int kesamaan = 0;
137.     for (int b = 0; b < (index); b++)
138.     {
139.         if (b == 0)
140.         {
141.             if (himpunankosongdiawal == 0)
142.             {
143.                 densitaskombinasi[k - 1, indexkombinasi] =
144.                 densitaskombinasisementara[b];
145.                 for (int d = 0; d < jmlpenyakit; d++)
146.                 {
147.                     kombinasi[k - 1, indexkombinasi, d] =
148.                     kombinasisementara[b, d];
149.                 }
150.                 hapus[indexhapus] = b;
151.                 indexhapus++;
152.                 for (int c = 0; c < (index); c++)
153.                 {
154.                     if (!(hapus.Any(c.Equals)))
155.                     {
156.                         kesamaan = 0;
157.                         for (int d = 0; d < jmlpenyakit; d++)
158.                         {
159.                             if (kombinasi[k - 1, indexkombinasi, d] ==
160.                             kombinasisementara[c, d])

```

```

158.     {
159.         kesamaan++;
160.     }
161. }
162. if (kesamaan == jmlpenyakit)
163. {
164.     densitaskombinasi[k - 1, indexkombinasi] =
        densitaskombinasi[k - 1, indexkombinasi] +
        densitaskombinasisementara[c];
165. }
166. hapus[indexhapus] = c;
167. indexhapus++;
168. }
169. }
170. }
171. indexkombinasi++;
172. }
173. }
174. else//B masuk mulai dari 1
175. {
176.     if (!(hapus.Any(b.Equals)))
177.     {
178.         densitaskombinasi[k - 1, indexkombinasi] =
            densitaskombinasisementara[b];
179.         for (int d = 0; d < jmlpenyakit; d++)
180.         {
181.             kombinasi[k - 1, indexkombinasi, d] =
                kombinasisementara[b, d];
182.         }
183.         hapus[indexhapus] = b;
184.         indexhapus++;
185.         for (int c = 0; c < (index); c++)
186.         {
187.             if (!(hapus.Any(c.Equals)))
188.             {
189.                 kesamaan = 0;
190.                 for (int d = 0; d < jmlpenyakit; d++)
191.                 {
192.                     if (kombinasi[k - 1, indexkombinasi, d] ==
                        kombinasisementara[c, d])
193.                     {
194.                         kesamaan++;
195.                     }
196.                 }
197.                 if (kesamaan == jmlpenyakit)
198.                 {
199.                     densitaskombinasi[k - 1, indexkombinasi] =
                        densitaskombinasi[k - 1, indexkombinasi] +
                        densitaskombinasisementara[c];
200.                     hapus[indexhapus] = c;
201.                     indexhapus++;
202.                 }
203.             }
204.         }
205.         indexkombinasi++;
206.     }

```

```

207.     }
208.     }
209.     //hitung rumus kombinasi DS
210.     if (nilaikosong != 0)
211.     {
212.         for (int c = 0; c < (index); c++)
213.         {
214.             densitaskombinasi[k - 1, c] = densitaskombinasi[k - 1,
                c] / (1 - nilaikosong);
215.         }
216.     }
217.     totalgejala++;
218. }
219. } //end for jumlah gejala pada data kasus
220. //MAX densitas densitas kombinasi terakhir
221. double max = 0;
222. int indexmax = 0;
223. int indexterakhir = totalgejala - 2;
224. for (int a = 0; a < (Convert.ToInt32(Math.Pow(2,
    (gejaladatalatih.GetLength(1) + 1)))); a++) //untuk
    mendapatkan deret terakhir ((2^k)+1)
225. {
226.     if (max < densitaskombinasi[indexterakhir, a])
227.     {
228.         max = densitaskombinasi[indexterakhir, a];
229.         indexmax = a;
230.     }
231. }
232. for (int b = 0; b < jmlpenyakit; b++)
233. {
234.     penyakit_awal[no_kasus, b] = kombinasi[indexterakhir,
        indexmax, b];
235. }

```

Source Code 5.3 Metode Dempster-Shafer

5.2.3 Implementasi Crossover

Metode *crossover* yang digunakan pada penelitian ini adalah *Extended Intermediate Crossover*. Dalam membangkitkan *offspring* dari proses *crossover* dilakukan pemilihan 2 *parent* dipilih secara acak. Selain itu yang harus dilakukan pada proses ini adalah membangkitkan bilangan random α dengan range -0.25 - 1.25. Setelah membangkitkan α dan mendapatkan 2 *parent*, selanjutnya adalah menghitung *offspring* dengan menggunakan Persamaan 2.13 dan Persamaan 2.14. Proses membangkitkan bilangan random α ditunjukkan pada baris kode ke-7 sampai dengan baris kode ke-14. Pemilihan 2 *parent* hingga perhitungan *offspring* ditunjukkan pada baris kode ke-15 sampai dengan baris kode ke-31.

```

static void crossover()
1.     alpha = new double[jmlgejala, jmlpenyakit];
2.     jmloffspringcross = Convert.ToInt32(probcross * popsize);
3.     childcross = new double[jmloffspringcross, jmlgejala,
        jmlpenyakit];

```

```

4.   index_parent1 = new int[jmloffspringcross];
5.   index_parent2 = new int[jmloffspringcross];
6.   //random alpha -0.25 - 1.25
7.   Random random = new Random();
8.   for (int i = 0; i < jmlgejala; i++)
9.   {
10.    for (int j = 0; j < jmlpenyakit; j++)
11.    {
12.      alpha[i, j] = random.NextDouble() * (1.25 - (-0.25)) + (-
13.        0.25);
14.    }
15.  }
16.  for (int i = 0; i < jmloffspringcross; i++)
17.  {
18.    index_parent1[i] = random.Next(0, popsize);
19.    index_parent2[i] = random.Next(0, popsize);
20.    for (int j = 0; j < jmlgejala; j++)
21.    {
22.      for (int k = 0; k < jmlpenyakit; k++)
23.      {
24.        childcross[i, j, k] = individu[index_parent1[i], j, k] +
25.          (alpha[j, k] * (individu[index_parent2[i], j, k] -
26.            individu[index_parent1[i], j, k]));
27.        if (i + 1 < jmloffspringcross)
28.        {
29.          childcross[i + 1, j, k] = individu[index_parent2[i], j,
30.            k] + (alpha[j, k] * (individu[index_parent1[i], j, k] -
31.              individu[index_parent2[i], j, k]));
32.        }
33.      }
34.    }
35.    i++;
36.  }
37. }

```

Source Code 5.4 Proses Extended Intermediate Crossover

5.2.4 Implementasi Mutasi

Metode mutasi yang digunakan pada penelitian ini adalah *Random Mutation*. Untuk mendapatkan *offspring* dari proses mutasi hal-hal yang dilakukan adalah memilih individu yang menjadi induk pada proses mutasi secara acak, membangkitkan bilangan random r , dan membangkitkan *offspring* dengan menggunakan Persamaan 2.15. Membangkitkan bilangan random r ditunjukkan pada baris kode ke-9 dan ke-10. Untuk pemilihan induk dan perhitungan *offspring* ditunjukkan pada baris kode ke-11 sampai dengan baris kode ke-37.

```

static void mutation()
1.   double r;
2.   double jumlahoffspring = probmutasi * popsize;
3.   jmloffspringmutasi = Convert.ToInt32(jumlahoffspring);
4.   childmutasi = new double[jmloffspringmutasi, jmlgejala,
5.     jmlpenyakit];
6.   index_parent = new int[jmloffspringmutasi];
7.   int x;

```

```

7.  int y;
8.  //Random r
9.  Random random = new Random();
10. r = random.Next() * (0.1 - (-0.1)) + (-0.1);
11. for (int i = 0; i < jml offspringmutasi; i++)
12. {
13.     index_parent[i] = random.Next(0, popsize - 1);
14.     x = random.Next(0, jmlgejala);
15.     y = random.Next(0, jmlpenyakit);
16.     for (int j = 0; j < jmlgejala; j++)
17.     {
18.         for (int k = 0; k < jmlpenyakit; k++)
19.         {
20.             childmutasi[i, j, k] = individu[index_parent[i], j, k];
21.         }
22.     }
23.     double max = 0;
24.     double min = childcross[0, x, y];
25.     //Mencari nilai min dan max pada offspring
26.     for(int a = 0; a < jml offspringcross; a++)
27.     {
28.         if (childcross[a, x, y] > max)
29.         {
30.             max = childcross[a, x, y];
31.         }
32.         if (childcross[a, x, y] < min)
33.         {
34.             min = childcross[a, x, y];
35.         }
36.     }
37.     childmutasi[i, x, y] = childmutasi[i, x, y] + (r * (max -
38. min));
}

```

Source Code 5.5 Proses Random Mutation

5.2.5 Implementasi Seleksi

Metode yang digunakan pada proses seleksi adalah *Replacement Selection*, dimana jika nilai *fitness offspring* lebih baik dari nilai *fitness parent*, maka *offspring* akan menggantikan *parent*. Proses seleksi pada penelitian ini dibagi menjadi 2 bagian, yaitu seleksi untuk *offspring* yang dihasilkan dari proses *crossover* dan seleksi untuk *offspring* yang dihasilkan dari proses mutasi.

Pada proses seleksi untuk *offspring* proses *crossover* hal pertama yang dilakukan adalah memilih *parent*/induk yang memiliki nilai *fitness* terkecil, proses ini ditunjukkan pada baris kode ke-4 sampai dengan baris kode ke-10. Selanjutnya induk dengan nilai *fitness* terkecil dibandingkan dengan *offspring*, jika nilai *fitness offspring* lebih besar dari induknya maka *offspring* akan menggantikan induknya, jika tidak lebih besar maka induk tetap. Proses ini ditunjukkan pada baris kode ke-11 sampai dengan baris kode ke-21.

Pada proses seleksi untuk *offspring* yang dihasilkan dari proses mutasi induk langsung dibandingkan dengan *offspring*, karena induk pada proses mutasi hanya

1. Sama seperti proses seleksi *offspring crossover* jika nilai *fitness offspring* lebih

besar dari induknya maka *offspring* akan menggantikan induknya, jika tidak maka induk tidak diganti. Proses seleksi *offspring* mutasi ditunjukkan pada baris kode ke-25 sampai dengan baris kode ke-35.

```
static void seleksi()
1. //seleksi offspring crossover
2. for (int i = 0; i < jmlffspringcross; i++)
3. {
4.     double min = fitness[index_parent1[i]];
5.     int index = index_parent1[i];
6.     if (fitness[index_parent2[i]] < min)
7.     {
8.         min = fitness[index_parent2[i]];
9.         index = index_parent2[i];
10.    }
11.    if (fitnesscross[i] > min)
12.    {
13.        for (int j = 0; j < jmlgejala; j++)
14.        {
15.            for (int k = 0; k < jmlpenyakit; k++)
16.            {
17.                individu[index, j, k] = childcross[i, j, k];
18.                fitness[index] = fitnesscross[i];
19.            }
20.        }
21.    }
22. }
23. //seleksi offspring mutasi
24. for (int i = 0; i < jmlffspringmutasi; i++)
25. {
26.     if (fitnessmutasi[i] > fitness[index_parent[i]])
27.     {
28.         for (int j = 0; j < jmlgejala; j++)
29.         {
30.             for (int k = 0; k < jmlpenyakit; k++)
31.             {
32.                 individu[index_parent[i], j, k] = childmutasi[i, j, k];
33.                 fitness[index_parent[i]] = fitnessmutasi[i];
34.             }
35.         }
36.     }
37. }
```

Source Code 5.6 Proses Replacement Selection

5.3 Implementasi Antarmuka

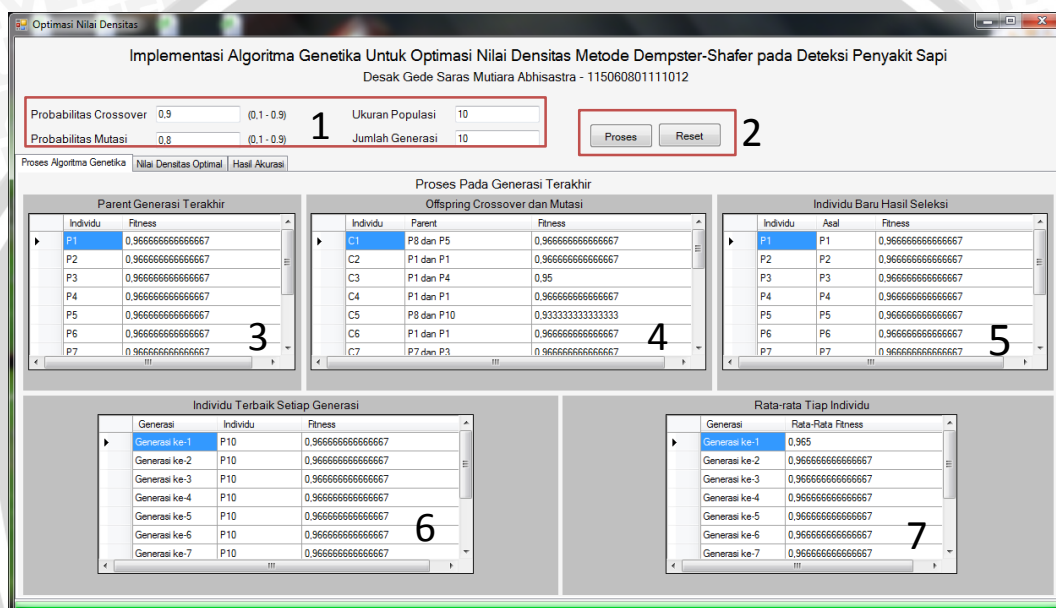
Antarmuka sistem digunakan oleh pengguna untuk berinteraksi dengan sistem. Antarmuka dibagi menjadi 3, yaitu halaman Proses Algoritma Genetika, halaman nilai densitas optimal, dan halaman hasil akurasi.

Pada antarmuka sistem terdapat 4 kolom parameter-parameter algoritma genetika yang harus diisi oleh pengguna. Parameter-parameter tersebut antara lain probabilitas *crossover*, probabilitas mutasi, jumlah generasi, dan ukuran populasi. Setelah melakukan optimasi, sistem akan menampilkan informasi-

informasi. Informasi yang diberikan dibagi menjadi 3 bagian, yaitu Proses Algoritma Genetika, Nilai Densitas Optimal dan Hasil Akurasi.

5.3.1 Tampilan Proses Algoritma Genetika

Setelah sistem melakukan optimasi pada tab Proses Algoritma Genetika akan ditampilkan tabel-tabel perhitungan proses algoritma genetika pada generasi terakhir. Tabel-tabel tersebut meliputi tabel *parent* dan *fitness*-nya, tabel *offspring*, *parent* dan *fitness*-nya, individu hasil seleksi, individu dengan *fitness* terbaik, dan rata-rata *fitness* di setiap generasi. Tampilan proses-proses pada algoritma genetika ditunjukkan pada Gambar 5.1.



Gambar 5.1 Tampilan Proses Algoritma Genetika

Pada Gambar 5.1, kolom-kolom yang ditunjukkan oleh nomor 1 merupakan kolom-kolom parameter yang harus diisi oleh pengguna. Kolom-kolom tersebut antara lain kolom probabilitas *crossover*, probabilitas probabilitas mutasi, ukuran populasi dan jumlah generasi. Untuk probabilitas *crossover* dan probabilitas mutasi pengguna harus mengisinya dengan nilai 0,1-0,9. Gambar nomor 2 merupakan tombol Proses dan tombol Reset. Tombol Proses digunakan untuk menjalankan proses optimasi dengan menggunakan parameter-parameter yang telah dimasukkan oleh pengguna. Sedangkan untuk tombol Reset digunakan untuk menghapus tabel-tabel informasi dan kolom-kolom.

Setelah menekan tombol Proses sistem akan menampilkan hasil pada 5 tabel. Tabel nomor 3 akan menampilkan *parent* pada generasi terakhir beserta *fitness*-nya. Tabel nomor 4 menampilkan *offspring* yang dihasilkan dari proses *crossover* dan mutasi dengan *parent* dan *fitness*-nya. Tabel nomor 5 menampilkan individu baru setelah melalui proses seleksi dengan asal individu dan *fitness*-nya. Tabel nomor 6 menampilkan generasi terbaik di setiap generasi

dan *fitness*-nya. Tabel nomor 7 menunjukkan *fitness* rata-rata yang dihasilkan di setiap generasi.

5.3.2 Tampilan Nilai Densitas Optimal

Pada tab Nilai Densitas Optimal akan menampilkan nilai densitas yang merupakan hasil dari proses optimasi menggunakan algoritma genetika beserta nilai *fitness* yang dihasilkan. Nilai densitas ini menunjukkan besarnya peran gejala-gejala terhadap penyakit yang ada. Gejala yang digunakan terdiri dari 179 gejala dan penyakit yang digunakan sebanyak 30 penyakit. Tampilan nilai densitas optimal dan *fitness*-nya ditunjukkan pada Gambar 5.2.

Optimasi Nilai Densitas

Implementasi Algoritma Genetika Untuk Optimasi Nilai Densitas Metode Dempster-Shafer pada Deteksi Penyakit Sapi

Desak Gede Saras Mutiara Abhisastira - 115060801111012

Probabilitas Crossover0.9(0.1 - 0.9)

Ukuran Populasi10

Probabilitas Mutasi0.8(0.1 - 0.9)

Jumlah Generasi10

Proses

Reset

Proses Algoritma Genetika

Nilai Densitas Optimal

Hasil Akurasi

Nilai Densitas Optimal :

Gejala/Penyakit	A	BR	M	TB	BL	PE	JD	S	BT	RK	V	N	K	SA	DT	BV	MC	MK	HV	R	AS
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.9322
2	0	0	0	0	0	0	0	0	0	0	0.8312	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.7915	0	0
4	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0.8702	0	0	0	0	0	0	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	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	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	0	0	0	0	0	0	0	0	0
12	0	0	0	0.7759	0	0	0	0	0	0	0	0.5893	0	0	0	0	0	0	0.7825	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.9122	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0.5124	0.3041	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0.6616	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Fitness = 0.966666666666667

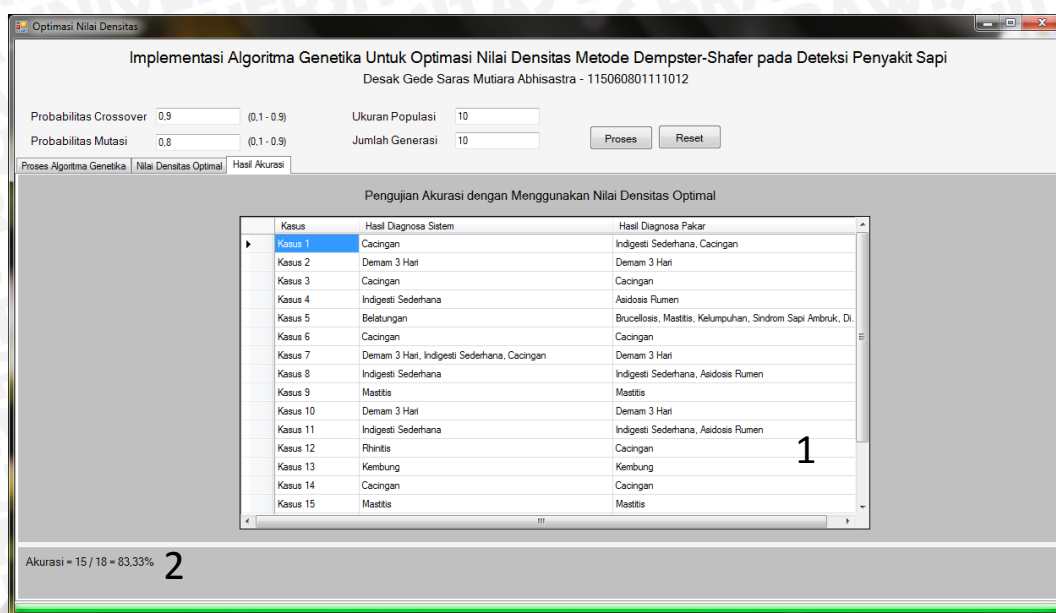
2

Gambar 5.2 Tampilan Nilai Densitas Optimal

Pada Gambar 5.2 tabel nomor 1 menampilkan individu terbaik yang dihasilkan dari proses optimasi dengan menggunakan algoritma genetika. Individu atau nilai densitas ini yang digunakan pada perhitungan akurasi. Gambar nomor 2 menampilkan nilai *fitness* yang dihasilkan oleh individu terbaik.

5.3.3 Tampilan Hasil Akurasi

Pada tab Hasil Akurasi akan menampilkan tabel perbandingan antara hasil diagnosa sistem dengan hasil diagnosa pakar. Perhitungan ini yang nantinya akan digunakan pada perhitungan akurasi. Perhitungan akurasi juga akan ditampilkan pada tab ini. Tampilan hasil akurasi sistem ditunjukkan pada Gambar 5.3.



Gambar 5.3 Tampilan Hasil Akurasi

Pada Gambar 5.3 tabel nomor 1 menampilkan hasil diagnosa yang diberikan pakar dan hasil diagnosa yang dihasilkan oleh sistem. Gambar nomor 2 menampilkan hasil akurasi yang dihasilkan sistem dengan menjumlahkan diagnosa sistem yang cocok dengan diagnosa pakar, kemudian dibagi dengan jumlah data yang digunakan dan dikalikan dengan 100%.

BAB 6 PENGUJIAN

Bab ini membahas tentang proses pengujian dan analisis terhadap sistem yang telah dibuat. Pada penelitian ini terdapat dua pengujian yaitu pengujian parameter algoritma genetika dan pengujian akurasi. Parameter algoritma genetika yang diuji antara lain nilai probabilitas *crossover* dan mutasi, ukuran populasi dan jumlah generasi. Pengujian akurasi digunakan untuk menguji tingkat akurasi antara perhitungan pada penelitian sebelumnya dengan perhitungan sistem.

6.1 Pengujian Parameter Algoritma Genetika

Pengujian parameter algoritma genetika terdiri dari tiga pengujian yaitu pengujian nilai probabilitas *crossover* dan mutasi, pengujian ukuran populasi dan pengujian jumlah generasi. Pengujian dilakukan dengan membandingkan perubahan nilai parameter algoritma genetika terhadap nilai *fitness* yang dihasilkan.

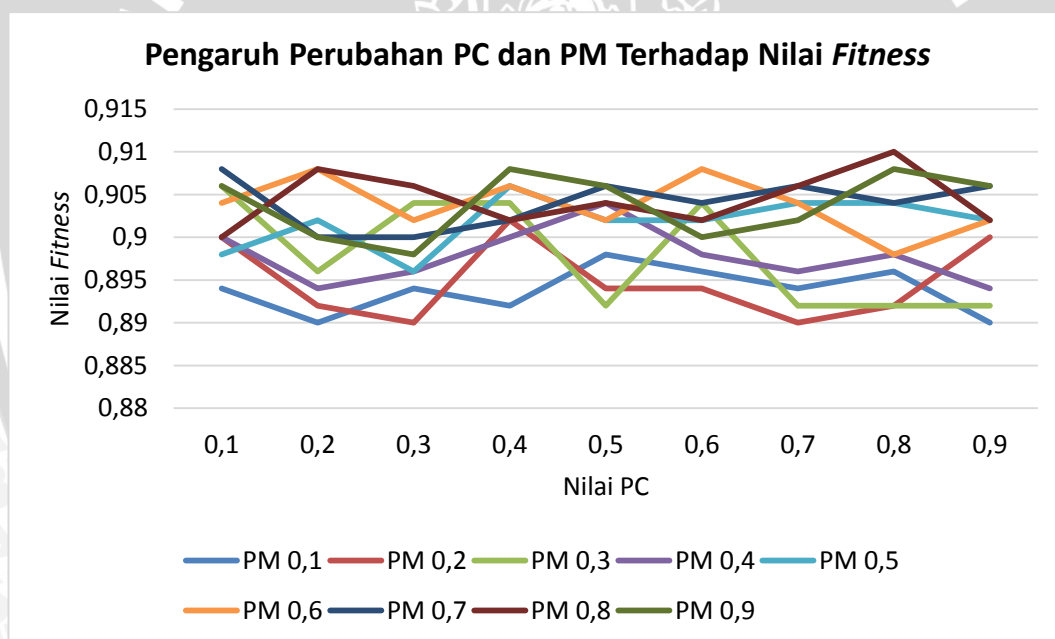
6.1.1 Pengujian Probabilitas *Crossover* dan Probabilitas Mutasi

Pengujian probabilitas *crossover* dan probabilitas mutasi ini bertujuan untuk mengetahui pengaruh nilai probabilitas *crossover* (*pc*) dan probabilitas mutasi (*pm*) terhadap besar nilai *fitness* yang dihasilkan. Selain itu pengujian ini bertujuan untuk mengetahui besar nilai *crossover* dan mutasi yang menghasilkan nilai *fitness* terbaik. Pada pengujian parameter yang digunakan yaitu ukuran populasi 10 dan jumlah generasi 10. Pengujian ini dilakukan terhadap kombinasi nilai dari probabilitas *crossover* (0,1 - 0,9) dan probabilitas mutasi (0,1 - 0,9) dan dilakukan 5 kali percobaan di setiap kombinasi lalu diambil nilai rata-rata *fitness*-nya. Nilai rata-rata *fitness* di setiap kombinasi akan dibandingkan dan dianggap kombinasi terbaik jika memiliki nilai *fitness* lebih tinggi dari pada yang lainnya. *Fitness* yang didapat setelah melakukan 5 kali percobaan akan diambil rata-ratanya. Hasil rata-rata kelima percobaan ditunjukkan pada Tabel 6.1.

Tabel 6.1 Rata-rata *Fitness* di lima Pengujian

PC	PM								
	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9
0,1	0,894	0,89	0,894	0,892	0,898	0,896	0,894	0,896	0,89
0,2	0,9	0,892	0,89	0,902	0,894	0,894	0,89	0,892	0,9
0,3	0,906	0,896	0,904	0,904	0,892	0,904	0,892	0,892	0,892
0,4	0,9	0,894	0,896	0,9	0,904	0,898	0,896	0,898	0,894
0,5	0,898	0,902	0,896	0,906	0,902	0,902	0,904	0,904	0,902
0,6	0,904	0,908	0,902	0,906	0,902	0,908	0,904	0,898	0,902
0,7	0,908	0,9	0,9	0,902	0,906	0,904	0,906	0,904	0,906
0,8	0,9	0,908	0,906	0,902	0,904	0,902	0,906	0,91	0,902
0,9	0,906	0,9	0,898	0,908	0,906	0,9	0,902	0,908	0,906

Berdasarkan Tabel 6.1 nilai *fitness* rata-rata terbesar adalah 0,91, nilai ini terdapat pada saat nilai pc 0,8 dan nilai pm 0,8. Grafik pengaruh perubahan nilai probabilitas *crossover* dan probabilitas mutasi terhadap nilai *fitness*.



Gambar 6.1 Grafik Pengujian Nilai Probabilitas *Crossover* dan Probabilitas Mutasi Terhadap Nilai *Fitness*

Berdasarkan grafik pada Gambar 6.1 nilai *fitness* yang didapat dari kombinasi nilai probabilitas *crossover* dan probabilitas mutasi tidak tetap. Hal ini terjadi karena pada proses inisialisasi populasi awal, dan pemilihan *parent* pada proses *crossover* dan *mutasi* dilakukan secara *random*.

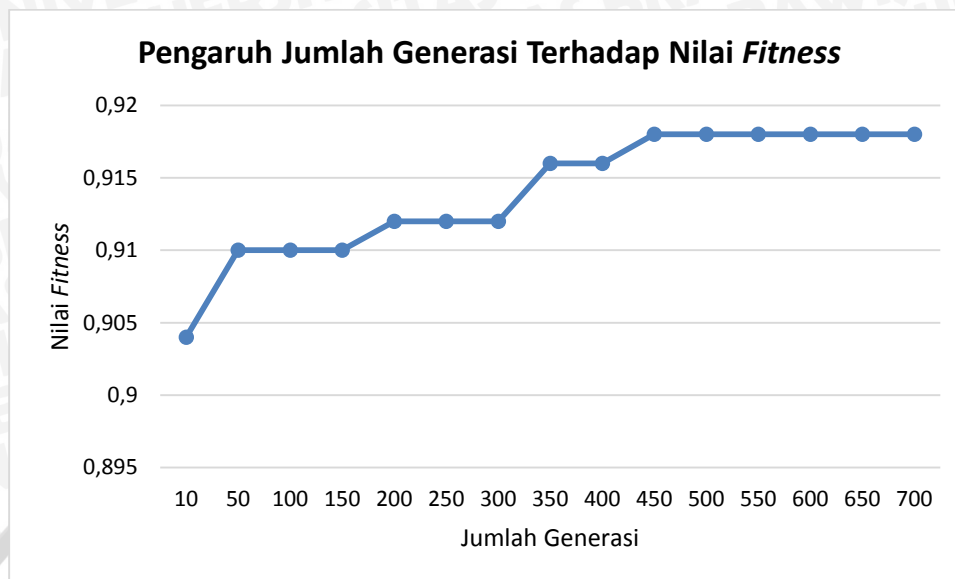
6.1.2 Pengujian Jumlah Generasi

Pengujian jumlah generasi bertujuan untuk mengetahui pengaruh jumlah generasi terhadap besar nilai *fitness* yang dihasilkan. Dalam pengujian ini, digunakan parameter nilai pc 0,8, pm 0,8 dan ukuran populasi 10. Jumlah generasi yang diujikan yaitu 10, 30, 50, 70 dan 90 generasi. Percobaan dilakukan sebanyak 5 kali di setiap jumlah generasi, kemudian nilai *fitness* dirata-ratakan. Nilai rata-rata *fitness* di setiap jumlah generasi akan dibandingkan, jumlah generasi dianggap terbaik jika nilai rata-rata *fitness* yang dimiliki lebih besar diantara nilai rata-rata *fitness* yang lainnya. Tabel 6.2 menunjukkan hasil pengujian jumlah generasi.

Tabel 6.2 Pengujian Jumlah Generasi

Jumlah Generasi	Percobaan ke- <i>i</i>					Fitess Rata-rata
	1	2	3	4	5	
10	0,9	0,9	0,9	0,91	0,91	0,904
50	0,91	0,91	0,91	0,91	0,91	0,91
100	0,91	0,91	0,91	0,91	0,91	0,91
150	0,91	0,91	0,91	0,91	0,91	0,91
200	0,92	0,91	0,91	0,91	0,91	0,912
250	0,92	0,91	0,91	0,91	0,91	0,912
300	0,91	0,91	0,92	0,91	0,91	0,912
350	0,92	0,92	0,91	0,92	0,91	0,916
400	0,92	0,91	0,92	0,92	0,91	0,916
450	0,92	0,92	0,92	0,91	0,92	0,918
500	0,92	0,92	0,92	0,92	0,91	0,918
550	0,91	0,92	0,92	0,92	0,92	0,918
600	0,92	0,92	0,92	0,91	0,92	0,918
650	0,92	0,92	0,91	0,92	0,92	0,918
700	0,92	0,91	0,92	0,92	0,92	0,918

Berdasarkan Tabel 6.2 dapat dikatakan bahwa *fitnes* rata-rata terbesar yang dihasilkan saat jumlah generasi sebesar 450 yaitu 0,918. Dari jumlah generasi 450 dan selanjutnya nilai *fitness* mengalami konvergensi. Grafik pengaruh perubahan jumlah generasi terhadap nilai *fitness* ditunjukkan pada Gambar 6.2.



Gambar 6.2 Grafik Pengujian Jumlah Generasi Terhadap Nilai *Fitness*

Pada grafik Gambar 6.2 menunjukkan bahwa jumlah generasi berbanding lurus dengan nilai *fitness* yang dihasilkan, semakin banyak jumlah generasi nilai *fitness* semakin meningkat. Namun peningkatan nilai *fitness* berhenti pada titik tertentu yaitu pada jumlah generasi 450 dan selanjutnya nilai *fitness* mengalami konvergensi.

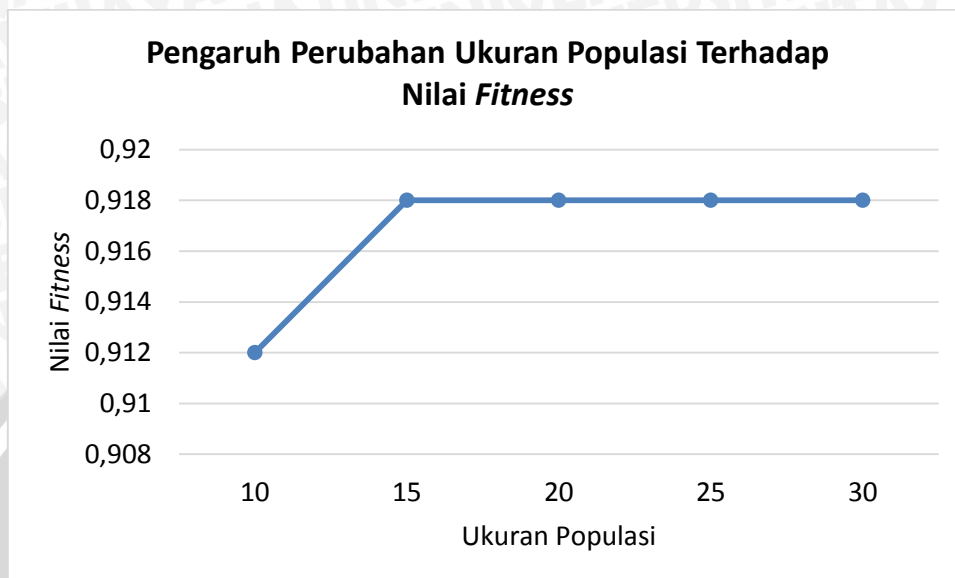
6.1.3 Pengujian Ukuran Populasi

Pengujian ukuran populasi bertujuan untuk mengetahui pengaruh ukuran populasi terhadap besar nilai *fitness* yang dihasilkan. Dalam pengujian ini, digunakan parameter nilai pc 0,8, nilai pm 0,8 dan jumlah generasi 450. Ukuran populasi yang diujikan yaitu 10, 15, 20, 25 dan 30 populasi. Pada pengujian ini akan dilakukan 5 kali percobaan di setiap ukuran populasi, kemudian nilai *fitness* akan dirata-rata. Setiap rata-rata nilai *fitness* di setiap ukuran populasi akan dibandingkan, ukuran populasi dianggap terbaik jika *fitness* yang dihasilkan lebih besar dari pada yang lainnya. Tabel 6.3 menunjukkan hasil pengujian ukuran.

Tabel 6.3 Pengujian Ukuran Populasi

Ukuran Generasi	Percobaan ke- <i>i</i>					<i>Fitness</i> Rata-rata
	1	2	3	4	5	
10	0,92	0,91	0,91	0,91	0,91	0,912
15	0,92	0,92	0,92	0,91	0,92	0,918
20	0,92	0,91	0,92	0,92	0,92	0,918
25	0,92	0,92	0,92	0,92	0,91	0,918
30	0,92	0,92	0,92	0,91	0,92	0,918

Berdasarkan hasil yang ditunjukkan Tabel 6.3 dapat dikatakan bahwa nilai *fitness* rata-rata terbesar yaitu 0,918 pada ukuran populasi sebesar 15 dan pada ukuran populasi selanjutnya telah terjadi konvergensi.



Gambar 6. 3 Grafik Perubahan Ukuran Populasi Terhadap Nilai *Fitness*

Berdasarkan Gambar 6.3 nilai *fitnes* meningkat dari ukuran populasi 10 ke ukuran populasi 15. Pada ukuran populasi 15 dan ukuran populasi selanjutnya nilai *fitness* menjadi konvergen.

6.2 Nilai Densitas Optimal

Setelah dilakukan pengujian dengan menggunakan parameter algoritma yang dianggap paling baik, sistem menghasilkan nilai densitas optimal. Tabel 6.5 menunjukkan nilai densitas yang optimal dengan nilai *fitness* 0,92. Nilai densitas optimal yang lebih lengkap ditunjukkan pada Lampiran E.

Tabel 6.4 Nilai Densitas Optimal

	A	BR	M	TB	BL	PE	JD	S	BT	RK	V	N	K	SA	DT
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0,815	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0,814	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	0	0	0
⋮															
179	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Tabel 6.5 Nilai Densitas Optimal (Lanjutan)

	BV	MC	MK	HV	R	AS	RP	D	IS	KB	AR	SR	C	KM	B
1	0	0	0	0	0	0,805	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0,779	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	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	0	0	0
6	0	0	0	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	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0,792	0,862	0	0,706
10	0	0	0	0	0	0	0	0	0,54	0,583	0	0	0,586	0	0
⋮															
179	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0,872

6.3 Pengujian Akurasi

Pengujian akurasi ini menggunakan nilai densitas yang sudah dioptimasi menggunakan algoritma genetika. Data yang digunakan pada pengujian akurasi ini adalah data kasus penyakit sapi yang digunakan pada penelitian sebelumnya yaitu penelitian yang dilakukan oleh Mustikadewi Prihastuti (2014).

Data pada pengujian akurasi ini sebanyak 18 data kasus. Parameter yang digunakan adalah parameter-parameter yang telah diuji sebelumnya, yaitu probabilitas *crossover* dan probabilitas mutasi 0,8, ukuran populasi 15 dan jumlah generasi 450. Pengujian akurasi dilakukan dengan menghitung klasifikasi yang cocok dengan diagnosa pakar. Klasifikasi dianggap cocok jika salah satu atau semua diagnosa sama dengan pakar. Jika hasil diagnosa sistem cocok dengan hasil diagnosa pakar maka akurasi hasil bernilai 1, jika tidak cocok maka akurasi hasil bernilai 0. Hasil pengujian akurasi ditunjukkan pada Tabel 6.5.

Tabel 6.5 Hasil Pengujian Akurasi

Kasus	Hasil Diagnosa Sistem	Nilai Densitas Sistem	Hasil Diagnosa Pakar	Akurasi Hasil
1	Indigesti Sederhana, Cacingan	0,926	Ascariasis (Cacingan)	1
2	Demam 3 Hari	0,99	Demam 3 Hari	1
3	Cacingan	0,993	Helminthiasis (Cacingan)	1
4	Indigesti Sederhana, Asidosis Rumen	0,662	Indigesti Sederhana	1
5	Belatungan	0,633	Myasis (Belatungan)	1
6	Cacingan	1	Enteritis (Cacingan)	1
7	Demam 3 Hari	0,993	Indigesti Sederhana, Ascariasis (Cacingan), Demam 3 Hari	1
8	Indigesti Sederhana,	0,866	Indigesti Sederhana	1

Kasus	Hasil Diagnosa Sistem	Nilai Densitas Sistem	Hasil Diagnosa Pakar	Akurasi Hasil
	Asidosis Rumen			
9	Mastitis	0,997	Mastitis	1
10	Demam 3 Hari	0,572	Demam 3 Hari	1
11	Indigesti Sederhana, Asidosis Rumen	0,866	Indigesti Sederhana	1
12	Demam 3 Hari	0,572	Rhinitis	0
13	Kembung	0,997	Bloat (Kembung)	1
14	Cacingan	0,787	Fasiolasis (Cacingan)	1
15	Mastitis	1	Mastitis	1
16	Indigesti Sederhana	0,993	Indigesti Sederhana	1
17	Sindrom Sapi Ambruk	0,867	Sindrom Sapi Ambruk	1
18	Indigesti Sederhana	0,566	Indigesti Sederhana	1

Berdasarkan Tabel 6.5 klasifikasi yang cocok dengan pakar sebanyak 17. Akurasi yang didapat adalah sebagai berikut

$$Akurasi = \frac{17}{18} \times 100\% = 94,44\%$$

Perbandingan antara hasil diagnosa sistem, hasil diagnosa penelitian sebelumnya dan hasil diagnosa pakar ditunjukkan pada Tabel 6.6.

Tabel 6.6 Hasil Diagnosa Sistem, Penelitian Sebelumnya dan Pakar

Kasus	Hasil Diagnosa Sistem	Nilai Densitas Sistem	Hasil Diagnosa Penelitian Sebelumnya	Nilai Densitas Penelitian Sebelumnya	Hasil Diagnosa Pakar
1	Indigesti Sederhana, Cacingan	0,926	Indigesti Sederhana, Cacingan	0,752	Ascariasis (Cacingan)

Kasus	Hasil Diagnosa Sistem	Nilai Densitas Sistem	Hasil Diagnosa Penelitian Sebelumnya	Nilai Densitas Penelitian Sebelumnya	Hasil Diagnosa Pakar
2	Demam 3 Hari	0,99	Demam 3 Hari	0,921	Demam 3 Hari
3	Cacingan	0,993	Cacingan	0,976	Helminthiasis (Cacingan)
4	Indigesti Sederhana, Asidosis Rumen	0,662	Asidosis Rumen	0,7	Indigesti Sederhana
5	Belatungan	0,633	Belatungan	0,73	Myasis (Belatungan)
6	Cacingan	1	Cacingan	0,982	Enteritis (Cacingan)
7	Demam 3 Hari	0,993	Demam 3 Hari	0,966	Indigesti Sederhana, Ascariasis (Cacingan), Demam 3 Hari
8	Indigesti Sederhana, Asidosis Rumen	0,866	Indigesti Sederhana, Asidosis Rumen	0,8	Indigesti Sederhana
9	Mastitis	0,997	Mastitis	0,992	Mastitis
10	Demam 3 Hari	0,572	Demam 3 Hari	0,287	Demam 3 Hari
11	Indigesti Sederhana, Asidosis Rumen	0,866	Indigesti Sederhana, Asidosis Rumen	0,8	Indigesti Sederhana
12	Demam 3 Hari	0,572	Demam 3 Hari	0,269	Rhinitis
13	Kembung	0,997	Kembung	0,997	Bloat (Kembung)
14	Cacingan	0,787	Cacingan	0,756	Fasiolasis (Cacingan)
15	Mastitis	1	Mastitis	1	Mastitis
16	Indigesti Sederhana	0,993	Indigesti Sederhana	0,926	Indigesti Sederhana
17	Sindrom	0,867	Sindrom	0,941	Sindrom Sapi

Kasus	Hasil Diagnosa Sistem	Nilai Densitas Sistem	Hasil Diagnosa Penelitian Sebelumnya	Nilai Densitas Penelitian Sebelumnya	Hasil Diagnosa Pakar
	Sapi Ambruk		Sapi Ambruk		Ambruk
18	Indigesti Sederhana	0,566	Indigesti Sederhana	0,366	Indigesti Sederhana

Berdasarkan Tabel 6.6 terdapat beberapa perbedaan hasil diagnosa antara diagnosa sistem dengan diagnosa sebelumnya antara lain.

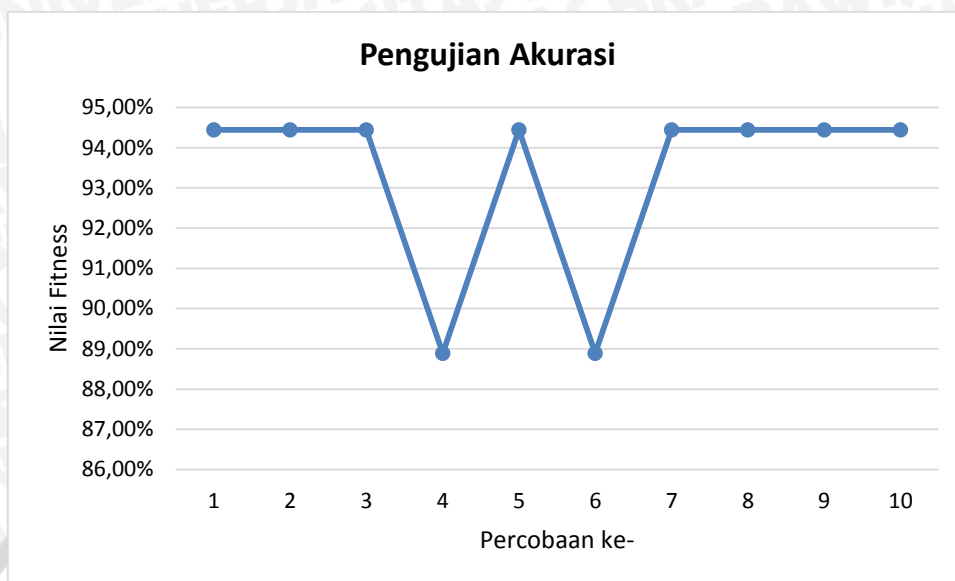
- Kasus 12 pakar mendiagnosa penyakit Rhinitis sedangkan sistem dan deagnosa penelitian sebelumnya Demam 3 Hari. Hal ini disebabkan karena pada penelitian ini maupun penelitian sebelumnya masih belum memiliki pengetahuan mengenai penyakit tersebut. Sehingga sistem tidak akan dapat mendiagnosa penyakit pada kasus ke-12.
- Kasus 1, 4, 8, dan 11 memiliki hasil diagnosa yang lebih dari 1. Hal ini dapat disebabkan karena beberapa penyakit memiliki gejala yang hampir sama, dan jika gejala yang dimasukkan merupakan gejala umum yang terjadi.
- Kasus 6 dan 15 memiliki nilai densitas yang tinggi yaitu 1. Hal ini dapat disebabkan karena gejala yang dimasukkan merupakan ciri khusus penyakit tersebut.

Pengujian ini dilakukan sebanyak 10 kali, kemudian hasil persentase dirata-rata. Hasil 10 kali pengujian akurasi ditunjukkan Tabel 6.7.

Tabel 6.7 Pengujian Akurasi

Percobaan ke- <i>i</i>	Hasil Persentase Akurasi
1	94,44%
2	94,44%
3	94,44%
4	88,89%
5	94,44%
6	88,89%
7	94,44%
8	94,44%
9	94,44%
10	94,44%

Berdasarkan hasil yang ditunjukkan pada Tabel 6.4, akurasi yang dihasilkan rata-rata akurasi yang dihasilkan dari 10 kali percobaan adalah 93,33%. Dari hasil akurasi yang dihasilkan sebagian besar percobaan menghasilkan akurasi sebesar 94,44%, dimana nilai ini lebih besar dari akurasi sebelumnya yaitu 88,89%.



Gambar 6.4 Pengujian Akurasi

Hasil pengujian akurasi terhadap 18 data kasus penyakit sapi setelah dilakukan optimasi pada nilai densitas metode Dempster-Shafer menggunakan algoritma genetika adalah 93,33%. Sedangkan penelitian sebelum dilakukan optimasi terhadap nilai densitas menghasilkan akurasi sebesar 88,89%. Hal ini menunjukkan bahwa sistem ini mampu meningkatkan nilai akurasi sistem sebesar 4,44%. Naiknya tingkat akurasi sistem ini disebabkan karena nilai densitas sudah optimal, sedangkan pada penelitian sebelumnya nilai densitas ditentukan secara manual oleh pakar sehingga hasil yang didapat kurang maksimal karena belum menggunakan optimasi.

BAB 7 PENUTUP

7.1 Kesimpulan

Berdasarkan hasil pengujian dan analisis yang dilakukan pada Implementasi Algoritma Genetika Untuk Optimasi Nilai Densitas Metode Dempster-Shafer pada Deteksi Penyakit Sapi, maka dapat diambil kesimpulan sebagai berikut:

1. Implementasi algoritma genetika dalam optimasi nilai densitas metode Dempster-Shafer dilakukan dengan mengubah nilai densitas menjadi kromosom. Kemudian dilakukan optimasi dengan menerapkan metode *Extended Intermediate Crossover*, metode *Random Mutation* dan metode *Replacement Selection*. Selanjutnya nilai densitas ini digunakan pada perhitungan Dempster-Shafer untuk diagnosa penyakit sapi.
2. Berdasarkan pengujian parameter probabilitas *crossover* dan probabilitas mutasi, kombinasi probabilitas *crossover* dan probabilitas mutasi yang menghasilkan nilai *fitness* terbaik adalah 0,8 dan 0,8. Pada pengujian ukuran populasi yang menghasilkan nilai *fitness* tertinggi adalah 15, nilai *fitness* yang dihasilkan ukuran populasi setelah 15 menjadi konvergen. Sedangkan pada pengujian jumlah generasi nilai *fitness* terbaik dihasilkan pada jumlah generasi 450, semakin besar jumlah generasi nilai *fitness* yang dihasilkan semakin tinggi, namun setelah 450 nilai *fitness* mengalami konvergensi.
3. Akurasi yang didapat oleh sistem ini dalam melakukan diagnosa penyakit sapi terhadap 18 data uji adalah sebesar 93,33%. Hasil ini mengalami peningkatan sebesar 4,44% dibandingkan dengan penelitian sebelumnya yang tidak menggunakan optimasi nilai densitas yaitu sebesar 88,89%.

7.2 Saran

Berdasarkan penelitian yang sudah dilakukan, optimasi nilai densitas pada metode Dempster-Shafer masih memiliki kekurangan. Saran yang dapat diberikan untuk pengembangan sistem ini lebih lanjut antara lain:

1. Berdasarkan pengujian akurasi, sistem ini tidak dapat mendeteksi penyakit lain selain 30 penyakit yang ada pada Lampiran A. Untuk penelitian selanjutnya sistem ini dapat dikembangkan dengan menambahkan pengetahuan baru.
2. Penelitian ini menggunakan *range* saat *me-random* populasi awal, sehingga masih memerlukan peranan pakar untuk memberikan *range*. Untuk penelitian selanjutnya sistem ini dapat dikembangkan dengan menggunakan metode klasifikasi untuk pembobotan, sehingga tidak memerlukan peranan pakar dalam menentukan bobot. Metode yang dapat digunakan misalnya Naïve Bayes.

DAFTAR PUSTAKA

- Anggariawan, H., 2014. *Penerapan Algoritma Genetika Untuk Optimasi Fungsi Keanggotaan Fuzzy Interface System Model Sugeno Pada Perhitungan Angka Metabolisme Basal (AMB)*. PTIIK Univ. Brawijaya.
- Correa, A., González, A., Ladino, C., 2011. *Genetic Algorithm Optimization for Selecting the Best Architecture of a Multi-Layer Perceptron Neural Network: A Credit Scoring Case*. Data Mining and Text Analytics 1–8.
- Haloho, R.D., Santoso, S.I., Marzuki, S., 2013. *Analisis Profitabilitas pada Usaha Peternakan Sapi Perah di Kabupaten Semarang*. Pengembangan Humaniora 13, 65–72.
- Haupt, R.L., Haupt, S.E., 2004. *Practical Genetic Algorithms*. Second Edition. ed. John Wiley & Sons, Inc., New Jersey.
- Herenda, D., 1994. *Manual on meat inspection for developing countries*. 119. Food and Agriculture Organization of the United Nations Rome.
- Hidalgo, D., Melin, P., Castillo, O., 2012. *An optimization method for designing type-2 fuzzy inference systems based on the footprint of uncertainty using genetic algorithms*. Expert Systems with Applications 39, 4590–4598.
- Kusumadewi, S., 2003. *Artificial Intelligence (Teknik dan Aplikasinya)*. Graha Ilmu, Yogyakarta.
- Kusumoputro, B., Irwanto, P., 2002. *Penggunaan Algoritma Genetika untuk Optimasi Jaringan Neural Buatan-Fuzzy dan Aplikasinya pada Sistem Penciuman Elektronik*. MAKARA SAINS 6.
- Lee, Y., Lee, J., 2015. *Binary tree optimization using genetic algorithm for multiclass support vector machine*. Elsevier 42, 3843–3851.
- Mahmudy, W.F., 2013. *Algoritma Evolusi*. Program Teknologi Informasi dan Ilmu Komputer (PTIIK), Universitas Brawijaya.
- Maseleno, A., Hasan, M.M., 2014. *Skin Diseases Expert System using Dempster-Shafer Theory*. Intelligent Systems and Applications 5, 38–44.
- Prihastuti, M., 2014. *Aplikasi Sistem Pakar Untuk Pendeteksian dan Penanganan Dini pada Penyakit Sapi dengan Metode Dempster-Shafer Berbasis Web*. PTIIK Univ. Brawijaya 1–10.
- Ressang, A.A., 1986. *Penyakit Viral pada Hewan Mamalia yang Penting di Indonesia*. Penerbit Universitas Indonesia (UI-PRESS), Jakarta.
- Sentz, K., Ferson, S., 2002. *Combination of Evidence in Dempster-Shafer Theory*. SANDIA REPORT.
- Sutojo, T., Mulyanto, E., Suhartono, V., 2011. *Kecerdasan Buatan*. Penerbit ANDI, Yogyakarta.
- Syibli, M., 2014. *Manual Penyakit Pada Mamalia. Subdit Pengamatan Penyakit Hewan Direktorat Kesehatan Hewan Direktorat Jenderal Peternakan dan Kesehatan Hewan Kementerian Pertanian*. Jakarta.

- Tjahajati, I., Husniyati, 2012. *Berbagai Penyakit Pada Sapi*. Citra Aji Prama, Yogyakarta.
- Whittier, W.D., Currin, J., Currin, N., 2009. *Pinkeye in Beef Cattle*. Va. Tech.
- Wiharto, Palgunadi, Y.S., Nugroho, M.A., 2013. *Analisis Penggunaan Algoritma Genetika untuk Perbaikan Jaringan Syaraf Tiruan Radial Basis Function*. Semin. Nas. Teknol. Inf. Dan Komun.
- Zhao, Y., Li, B., 2007. *A New Method for Optimizing Fuzzy Membership Fuction*. International Conference on Mechatronics and Automation.



LAMPIRAN A DAFTAR NAMA PENYAKIT

No.	Kode Penyakit	Nama Penyakit
1	A	Antrak
2	BR	Brucellosis
3	M	Mastitis
4	TB	Tuberkulosis
5	BL	Blackleg
6	PE	Pink Eye
7	JD	Johne Disease
8	S	Salmonelosis
9	BT	Botulisme
10	RK	Radang Kuku
11	V	Vibriosis
12	N	Ngorok
13	K	Kelumpuhan
14	SA	Sindrom Sapi Ambruk
15	DT	Demam 3 Hari
16	BV	Diare Ganas Menahun
17	MC	Malignan Chataral Fever
18	MK	Penyakit Mulut dan Kuku
19	HV	Bovine Herpes Virus
20	R	Ringworm
21	AS	Aspergilosis
22	RP	Retensi Plasenta
23	D	Distokia
24	IS	Indigesti Sederhana
25	KB	Kembung
26	AR	Asidosis Rumen
27	SR	Surra
28	C	Cacingan
29	KM	Kudisan Menular
30	B	Belatungan

LAMPIRAN B DAFTAR GEJALA

Kode	Gejala
1	Aborsi pada 6-9 bulan kebuntingan (trimester terakhir / 3)
2	Aborsi pada bulan keempat sampai akhir kebuntingan
3	Aborsi pada trimester kedua kebuntingan 100 hari setelah infeksi
4	Air susu bercampur darah atau endapan
5	Air susu encer dan rusak
6	Air susu kental dan menguning
7	Air susu pecah dan menggumpal
8	Ambing berwarna kemerahan dan sakit
9	Anemia
10	Anus kotor
11	Bagian perut membesar
12	Batuk
13	Batuk berdahak
14	Bau plasenta yang mulai membusuk
15	Bengkak daerah alat kelamin luar
16	Bengkak daerah dada dan sisi lambung
17	Bengkak daerah leher
18	Bengkak daerah perut
19	Bengkak daerah pinggang
20	Bengkak dan lesi terjadi pada penis dan preputium (sapi jantan)
21	Bengkak kelenjar getah bening
22	Bengkak pada ambing
23	Bengkak pada persendian lutut
24	Bentuk dada terdapat peradangan bronkoparmonia
25	Berbaring pesternum dengan kepala mengarah ke belakang seperti huruf S jika dilihat dari belakang
26	Berhenti merumput
27	Busung daerah dagu dan anggota gerak
28	Busung di bawah dagu
29	Busung di bawah rahang
30	Busung di leher bagian bawah sampai gelambir
31	Busung di tenggorokan
32	Cairan janin yang keluar kelihatan keruh
33	Celah kuku mengeluarkan cairan putih keruh
34	Daerah otot kaki berwarna merah kehitaman dengan gelembung
35	Darah berwarna hitam pekat seperti kecap, bau amis, busuk dan susah beku
36	Dehidrasi

Kode	Gejala
37	Demam / panas ($> 39^{\circ}\text{C}$) minimal 3 hari
38	Demam / panas $> 39^{\circ}\text{C}$
39	Demam / panas tinggi $> 40^{\circ}\text{C}$
40	Demam ringan ($38 - 39^{\circ}\text{C}$)
41	Demam ringan atau tinggi bergantian
42	Denyut jantung terasa lemah dan lebih cepat
43	Diare
44	Diare bercampur darah
45	Diare berlendir
46	Diare encer akut
47	Duduk / tidak mampu berdiri / kelemahan anggota gerak
48	Duduk memanjangkan kaki belakang seperti posisi kaki katak
49	Edema pada rahang bawah
50	Erosi pada selaput lendir hidung, lidah, bibir, gusi dan bagian belakang maksila
51	Erupsi kulit daerah leher dan bahu melingkar berbentuk cincin berwarna putih keputihan
52	Feses encer
53	Feses lembek sampai berair dan berbau busuk
54	Feses normal (berbentuk bulat) sampai keras dan jumlahnya sedikit
55	Flank bagian kiri atas membesar
56	Frekuensi kencing sering dan mengejan
57	Gaya berjalan kaku
58	Gelisah / depresi
59	Gemetar
60	Sapi menggosok-gosokkan badan pada dinding kandang hingga luka lecet
61	Hidung sapi kering
62	Hidung tersumbat kerak
63	Hiperlakrimasi
64	Hiperthermia / suhu badan meningkat
65	Hipomotilitas pada rumen sampai atoni
66	Infertilitas (Infestasi larva di saluran reproduksi / pasca calving)
67	Inflamasi pada bagian dalam mata
68	Jalan sempoyongan, kejang dan berputar-putar
69	Jika sapi mati dan dilakukan nekropsi akan ditemukan benjolan putih pada paru
70	Kadang terjadi radang kulit (penbalan dan pengelupasan kulit)
71	Kaki terasa dingin saat diraba
72	Kebuntingan kembar
73	Keguguran 1-3 kali
74	Keguguran pada bulan ke-4 sampai akhir kebuntingan

Kode	Gejala
75	Keguguran pada kebuntingan 5 – 8 bulan
76	Kejang-kejang
77	Kekakuan anggota gerak sampai pincang
78	Keluar air mata (Thelazia)
79	Keluar darah dari lubang alami (mulut, hidung, telinga, dubur dan kemaluan)
80	Keluar discharge ringan
81	Keluar ingus
82	Keluar leleran / cairan dari hidung dan mata
83	Keluar leleran atau serous dari lubang hidung
84	Keluar leleran mukopurulon dari hidung
85	Kelumpuhan otot-otot pernafasan
86	Kelumpuhan pada periode 3 hari setelah melahirkan
87	Kelumpuhan sebelum mati
88	Kelumpuhan temporer / permanen
89	Kematian dalam 3-4 hari dengan posisi sapi tidak berubah
90	Kematian dini pada embrio
91	Kematian mendadak dalam waktu singkat
92	Kerak pada permukaan kulit berwarna keabuan
93	Kerusakan jaringan pada tempat infeksi larva
94	Konjungtivitis dengan criminal discharge / discharge mata
95	Konsistensi isi rumen padat atau keras
96	Kontraksi pada rumen berkurang atau tidak ada / atoni
97	Kontraksi uterus yang lemah
98	Kornea mata keruh atau keputihan (Jika parah menjadi buta)
99	Kuku berbau busuk
100	Kuku membusuk berwarna merah kehitaman
101	Kulit kotor
102	Kulit kuku mengelupas
103	Lemah / lesu
104	Lepuh atau luka disekitar batas atas kuku dan ambing susu
105	Lepuh atau tukak atau tonjolan bulat pada gusi, lidah dan pangkal lidah
106	Lepuh-lepuh bernanah pada kulit
107	Luka atau keruh pada kornea mata
108	Lumpuh dengan pembengkakan otot (Usia 6 bulan – 2 tahun)
109	Malas bergerak
110	Mata dan hidung mengeluarkan leleran kental dan mukopurulon
111	Menderita kelumpuhan progresif
112	Mengeluarkan air liur berlebihan / Hipersaliva
113	Moncong kering dan pecah-pecah terisi nanah
114	Nafas berat

Kode	Gejala
115	Nafsu makan hilang / berkurang / Anorexia
116	Otot-otot gemetar/ semoyongan
117	Pada kelopak mata sapi terlihat seperti benang putih kecil cacing-cacing thelazia bergerak di bola mata
118	Paha gemetar
119	Panas saat ambing diraba
120	Pedet mati berwarna biru kecoklatan
121	Pembengkakan limpa glandula
122	Pembengkakan mata
123	Pencabalan dan kekakuan kulit dapat lokal sampai meluas
124	Pendarahan pada keropeng
125	Peningkatan sensitivitas terhadap cahaya
126	Penurunan angka konsepsi
127	Penurunan berat badan
128	Peradangan lidah dan mulut bagian dalam
129	Peradangan mulut dan lepuhan dipermukaan lidah
130	Peradangan pada buah pelir (sapi jantan)
131	Peradangan pada hidung (<i>red nose</i>)
132	Peradangan pada saluran sperma (sapi jantan)
133	Permukaan kulit meninggi, berbisik / berbungkul-bungkul pada daerah keropeng
134	Pernafasan cepat dan dangkal
135	Perut terasa keras / kencang saat diraba
136	Perut terasa seperti ada udara dan bunyinya seperti tong kosong saat dipukul / ditepuk
137	Pincang saat berjalan
138	Produksi susu menurun (pada sapi perah atau sapi yang laktasi)
139	Rambut kusam dan berdiri
140	Rambut patah atau rontok
141	Rambut patah / rontok / kebotaka pada lesi
142	Rambut terasa kasar
143	Refleks pupil jelek
144	Respirasi cepat
145	Roboh / jatuh / ambruk
146	Rumen penuh dan keras
147	Rumen tidak terlalu penuh dan berisi banyak cairan
148	Ruminasi berhenti
149	Ruminasi di bawah normal / tidak ada ruminasi
150	Sakit di daerah perut
151	Sakit pada paha apabila dipalpasi
152	Sapi menjadi buta
153	Sapi menjadi pincang hingga lumpuh

Kode	Gejala
154	Sapi terlihat takut terkena cahaya / sinar matahari
155	Sekitar celah kuku membengkak
156	Selaput fetus menggantung di luar alat kelamin ≥ 12 jam
157	Selaput lendir menguning
158	Selaput lendir mulut kemerahan
159	Sering berkedip
160	Sering kencing
161	Sering terlihat merangkak untuk mengubah posisi
162	Skrotum bengkak dan membesar (hernia) pada sapi jantan
163	Sulit melahirkan karena ukuran fetus yang terlalu besar
164	Sulit melahirkan karena ukuran jalan kelahiran yang kecil
165	Susah bernafas
166	Susah buang kotoran
167	Susah untuk berdiri
168	Terasa sakit saat ambing disentuh / palpasi
169	Terdapat luka-luka pada kaki depan dan belakang
170	Tidak mampu menopang berat badan
171	Tinja / kotoran berbau busuk
172	Tubuh kurus
173	Tumbuh benjolan pada kuku yang menimbulkan rasa sakit
174	Ulkus hingga ke iris
175	Ulkus pada kornea / menyebar
176	Uterus yang melintir atau terputar
177	Vagina vulva bengkak, timbul papul erosi dan ulcer (sapi betina)
178	Vulva bengkak dan kemerahan
179	Peradangan (berdarah) luka di tempat infestasi larva

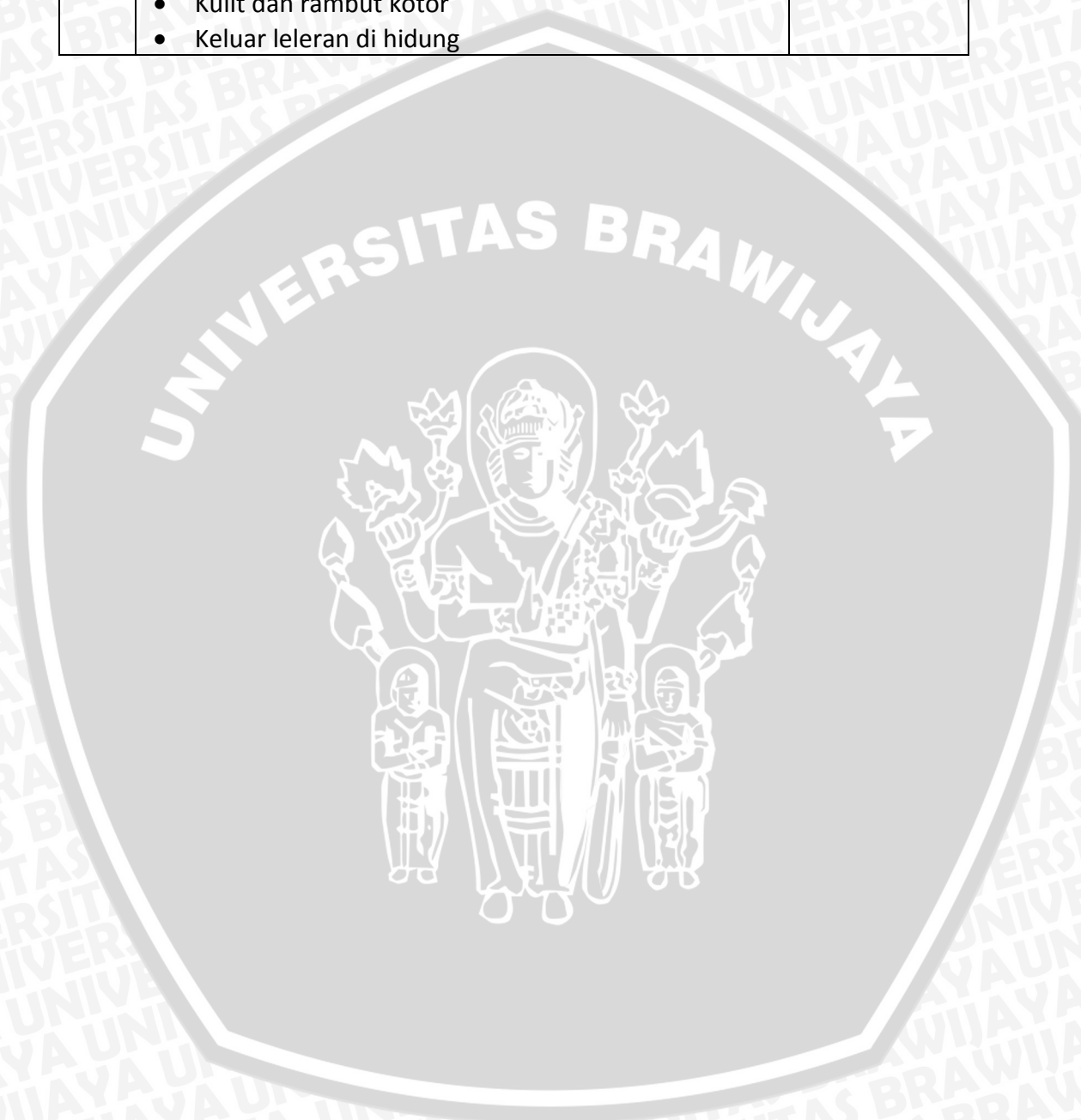
LAMPIRAN C DATA KASUS PENYAKIT SAPI

No.	Gejala	Diagnosa
1	- Suhu tubuh : 38,2°C - Rambut rontok - Diare 3 hari - Diare encer berwarna putih - Nafsu makan menurun - Frekuensi nafas 76x/menit, Pulsus 80x/menit	Ascariasis (Cacingan)
2	- Suhu tubuh : 39,3 - Rambut rontok - Konjungtivitas mata normal - Keluar leleran dari mata dan hidung - Tidak dapat berdiri - Lesu - Frekuensi nafas 40x/menit, pulsus 132x/menit (tinggi)	Bovine Ephemeral Fever (Demam 3 hari)
3	- Suhu tubuh : 39°C - Tubuh lesu - Tubuh kurus - Rambut kusam, berdiri, rontok dan teraba kasar - Frekuensi nafas 32x/menit, pulsus 100x/menit (normal)	Helminthiasis (Cacingan)
4	- Nafsu makan menurun - Suhu tubuh meningkat - Lesu - Malas bergerak	Indigesti Sederhana
5	- Suhu tubuh : 39 °C - Peradangan luka di tempat infestasi larva - Frekuensi nafas 44x/menit, pulsus 96x/menit (tinggi)	Myasis (Belatungan)
6	- Suhu tubuh : 39,7 °C - Tubuh kurus - Lesu - Rambut rontok dan kusam - Diare encer berwarna kekuningan - Dehidrasi - Anoreksia - Frekuensi nafas 64x/menis, pulsus 140x/menit (tinggi)	Enteritis (Cacingan)

No.	Gejala	Diagnosa
7	- Suhu tubuh : 39,7 °C - Nafsu makan menurun - Diare - Hidung berair - Ambruk/ tidak dapat berdiri - Keluar leleran dari mata - Rambut rontok - Frekuensi nafas 36x/menit, pulsus 88x/menit (tinggi)	Bovine Ephemeral Fever, Indigesti dan Ascariasis
8	- Nafsu makan tidak ada - Lesu - Tinja berbentuk bulatan keras - Frekuensi nafas 32x/menit, pulsus 68x/menit	Indigesti Sederhana
9	- Suhu tubuh : 41 °C - Ambing bengkak dan kemerahan - Terasa sakit saat ambing disentuh - Nafsu makan menurun - Malas bergerak - Cermin hidung kering - Palpasi gladula mammae bengkak - Frekuensi nafas 48x/menit, pulsus 76x/menit (tinggi)	Mastitis
10	- Suhu tubuh : 39 °C - Lesu - Anus kotor - Feses encer - Keluar leleran serous dari hidung - Nafsu makan hilang - Hipersaliva - Frekuensi nafas 25x/menit, pulsus 48x/menit	Bovine Ephemeral Fever (Demam 3 hari)
11	- Tinja berbentuk bulat keras - Nafsu makan tidak ada - Lesu - Frekuensi nafas 32x/menit, pulsus 68x/menit	Indigesti Sederhana
12	- Suhu tubuh : 39 °C - Nafsu makan hilang - Keluar leleran hidung - Hipersaliva - Lesu - Anus kotor - Feses encer	Rhinitis

No.	Gejala	Diagnosa
13	• Frekuensi nafas 32x/menit, pulsus 48x/menit • Suhu tubuh : 37,2 °C • Lesu • Ambruk • Rambut berdiri • Frekuensi nafas 64x/menit, pulsus 60x/menit • Gerak rumen tidak terasa • Flank kiri membesar • Anus kotor • Tidak dapat berdiri	Bloat (Kembung)
14	• Bulu berdiri • Tubuh kurus • Lesu • Suhu tubuh : 38,6 °C • Feses tidak berbentuk • Anus kotor • Frekuensi nafas 36x/menit, pulsus 80x/menit (normal)	Fasiolasis (Cacingan)
15	• Ambing sapi bengkak • Air susu berwarna kekuningan • Tubuh kurus • Suhu tubuh kurus • Suhu tubuh : 38,6 °C • Rambut kusam dan rontok • Ambing kuartir bagian kiri depan bengkak kemerahan • Saat ambing dipalpasi terasa sakit • Air susu encer dan rusak • Frekuensi nafas 36x/menit, pulsus 60x/menit	Mastitis
16	• Suhu tubuh : 39 °C • Lesu • Frekuensi nafas 40x/menit, pulsus 72x/menit • Suara perkusi rumen terdengar pekak • Feses lunak kehitaman • Anus kotor	Indigesti Sederhana
17	• Suhu tubuh : 38,8 °C • Ambruk • Tidak sanggup berdiri lagi • Setelah melahirkan tidak dapat bangun • Nafsu makan menurun • Kekakuan kaki kanan terutama bagian kanan karena	Sindrom Sapi Ambruk

No.	Gejala	Diagnosa
	tertindih lama	
18	• Suhu tubuh : 38,9 °C • Nafsu makan menurun • Lemah • Kulit dan rambut kotor • Keluar leleran di hidung	Indigesti Sederhana



LAMPIRAN D DATA LATIH

No.	Gejala	Diagnosa
1.	- Suhu tubuh : 41°C - Ambing bengkak dan kemerahan - Terasa sakit saat ambing disentuh - Nafsu makan menurun - Malas bergerak - Cermin hidung kering - Palpasi gladula mammae bengkak	Mastitis
2.	- Suhu tubuh : 39,5°C - Nafsu makan berkurang - Bengkak daerah pinggang dan kelenjar getah bening - Bengkak daerah perut - Keluar darah dari hidung dan mulut - Susah bernafas	Antrak
3.	- Batuk berkepanjangan - Bengkak kelenjar getah bening - Susah bernafas	Tuberkulosis
4.	- Inflamasi pada bagian dalam mata - Peningkatan sensitivitas terhadap cahaya - Ulkus pada kornea - Buta - Uterus terputar	Pink Eye
5.	- Nafsu makan menurun - Diare encer akut - Berat badan menurun - Produksi susu menurun - Kotoran berbau busuk - Tubuh kurus	Johne Disease
6.	- Suhu tubuh : 39,5°C - Nafsu makan berkurang - Diare bercampur darah - Lemah - Produksi susu menurun	Salmonelosis
7.	- Kerak berwarna abu pada permukaan kulit - Lepuh-lepuh bernanah pada kulit - Rambut mudah patah	Kudisan Menular
8.	- Suhu tubuh : 38,2°C - Rambut rontok - Diare 3 hari	Cacingan

No.	Gejala	Diagnosa
	• Diare encer berwarna putih • Nafsu makan menurun	
9.	• Suhu tubuh : 40°C • Nafsu makan menurun • Keluar leleran dari hidung • Keluar discharge ringan • Sering kencing • Susah bernafas	Bovine Herpes Virus
10.	• Pendarahan pada keropeng • Rambut rontok • Respirasi cepat	Ringworm
11.	• Kontraksi uterus lemah • Sulit melahirkan karena ukuran fetus yang terlalu besar • Sulit melahirkan karena ukuran jalan kelahiran yang kecil • Uterus terputar	Distokia
12.	• Suhu tubuh : 39,3°C • Nafsu makan berkurang • Limpa glandula membesar • Susah bernafas • Darah keluar dari mulut, hidung dan telinga	Brucellosis
13.	• Berhenti merumput • Daerah otot kaki berwarna merah kehitaman dengan gelembung • Lumpuh dengan pembengkakan otot • Penurunan berat badan • Sakit pada paha apabila dipalpasi	Blackleg
14.	• Suhu tubuh : 38,8 °C • Ambruk • Tidak sanggup berdiri lagi • Setelah melahirkan tidak dapat bangun • Nafsu makan menurun • Kekakuan kaki kanan terutama bagian kanan karena tertindih lama	Sindrom Sapi Ambruk
15.	• Suhu tubuh : 40,2°C • Nafsu makan hilang • Susah bernafas • Keluar ingus • Busung di leher bagian bawah sampai gelambir • Busung di bawah dagu	Ngorok

No.	Gejala	Diagnosa
	Busung di tenggorokan	
16.	- Susah bernafas - Nafas berat - Mengeluarkan air liur berlebihan - Mengalami kelumpuhan progresif - Kelumpuhan otot-otot pernafasan - Gaya berjalan kaku - Duduk memanjangkan kaki kebelakang seperti posisi kaki katak	Botulisme
17.	- Pincang - Kuku membusuk berwarna merah kehitaman - Celah kuku bengkak dan mengeluarkan cairan putih keruh - Kuku berbau busuk - Kulit kuku mengelupas - Sekitar celah kuku membengkak - Tumbuh benjolan pada kuku yang menimbulkan rasa sakit	Radang Kuku
18.	- Suhu tubuh : 39,3 - Rambut rontok - Konjungtivitas mata normal - Keluar leleran dari mata dan hidung - Tidak dapat berdiri - Lesu	Demam Tiga Hari
19.	- Suhu tubuh : 38,8°C - Tidak mampu berdiri - Nafsu makan berkurang - Gelisah - Ruminasi berhenti	Kelumpuhan
20.	- Kematian dini pada embrio - Aborsi pada bulan keempat - Penurunan angka konsepsi	Vibriosis
21.	- Suhu tubuh : 37,2 °C - Lesu - Ambruk - Rambut berdiri - Gerak rumen tidak terasa - Flank kiri membesar - Anus kotor - Tidak dapat berdiri	Kembung (Bloat)
22.	Suhu tubuh : 39,3°C selama 3 hari	Diare Ganas

No.	Gejala	Diagnosa
	- Suhu tubuh naik turun - Nafsu makan berkurang - Diare berlendir - Diare bercampur darah	Menahun
23.	- Suhu tubuh : 39,6°C - Aborsi pada bulan ke-7 kebuntingan - Batuk berdahak - Keluar leleran dari lubang hidung - Pernafasan cepat dan dangkal	Aspergilosis
24.	- Nafsu makan menurun - Suhu tubuh meningkat - Lesu - Malas bergerak	Indigesti Sederhana
25.	- Suhu tubuh : 39,5°C - Kulit kuku mengelupas - Lepuh di sekitar batas atas kuku dan ambing susu - Tukak bulat pada gusi, lidah dan pangkal lidah - Malas bergerak - Nafsu makan berkurang - Peradangan lidah dan mulut bagian dalam - Pincang saat berjalan	Penyakit Mulut dan Kuku
26.	- Suhu tubuh : 39 °C - Peradangan luka di tempat infestasi larva	Belatungan
27.	- Suhu tubuh : 40,3°C - Kornea mata keruh - Mata dan hidung mengeluarkan leleran kental dan mukopurulon - Kelumpuhan sebelum mati - Mengeluarkan air liur berlebihan - Moncong kering dan pecah-pecah terisi nanah - Nafsu makan hilang - Otot-otot gemetaran - Peradangan mulut dan lepuhan di permukaan lidah - Selaput lendir mulut kemerahan	Malignan Chataral Fever
28.	- Suhu tubuh : 39,8°C - Denyut jantung terasa lemah dan lebih cepat - Diare - Feses keras dan jumlahnya sedikit - Suhu tubuh meningkat - Lemas - Malas bergerak	Asidosis Rumen

No.	Gejala	Diagnosa
	- Nafsu makan berkurang - Rumen penuh dank eras - Ruminasi berhenti - Sakit di daerah perut	
29.	- Suhu tubuh : 39,8°C - Nafsu makan berkurang - Selaput fetus menggantung di luar alat kelamin selama lebih dari 12 jam - Gelisah - Bau plasenta yang mulai membusuk - Kontraksi uterus yang melemah - Produksi susu menurun - Respirasi cepat - Vulva bengkak dan kemerahan	Retensi Plasenta
30.	- Suhu tubuh : 40,4°C - Anemia - Busung di daerah dagu dan anggota gerak - Lemah - Jalan sempoyongan, kejang dan berputar-putar - Keluar cairan dari hidung dan mata - Nafsu makan berkurang - Rambut patah atau rontok - Selaput lendir menguning	Surra
31.	- Suhu tubuh : 38,6 °C - Ambing sapi bengkak - Air susu berwarna kekuningan - Tubuh kurus - Suhu tubuh kurus - Rambut kusam dan rontok - Ambing kuartir bagian kiri depan bengkak kemerahan - Saat ambing dipalpasi terasa sakit - Air susu encer dan rusak	Mastitis
32.	- Suhu tubuh : 39,6°C - Bengkak daerah perut - Bengkak daerah pinggang - Keluar darah dari mulut, hidung, telinga, dubur dan kemaluan - Kematian mendadak dalam waktu singkat - Nafsu makan berkurang - Susah bernafas	Antrak

No.	Gejala	Diagnosa
	Pembengkakan limpa glandula	
33.	- Batuk - Kelenjar getah bening membengkak - Jika sapi mati dan dilakukan nekropsi akan ditemukan benjolan putih pada paru - Produksi susu menurun - Susah bernafas - Tubuh kurus	Tuberkulosis
34.	- Hiperlakrimasi - Inflamasi pada bagian dalam mata - Peningkatan sensitifitas terhadap cahaya - Sapi menjadi buta - Ulkus pada kornea (menyebar) - Uterus terputar	Pink Eye
35.	- Diare encer akut - Edema pada rahang bawah - Penurunan berat badan - Produksi susu menurun - Kotoran berbau busuk - Tubuh kurus	Johne Disease
36.	- Suhu tubuh : 39,4°C - Diare bercampur darah - Lemah - Nafsu makan berkurang - Produksi susu menurun	Salmonelosis
37.	- Sapi menggosok-gosokkan badan ke dinding hingga terluka - Kerak pada permukaan kulit berwarna kemerahan - Lepuh-lepuh bernanah pada kulit - Rambut patah	Kudisan Menular
38.	- Suhu tubuh : 39°C - Tubuh lesu - Tubuh kurus - Rambut kusam, berdiri, rontok dan teraba kasar	Cacingan
39.	- Suhu tubuh : 40,3°C - Keluar discharge ringan - Keluar leleran dari lubang hidung - Keluar leleran mukopurulon dari hidung - Peradangan pada hinging - Pembengkakan pada vagina vulva, timbul papul erosi dan ulcer	Bovine Herpes Virus

No.	Gejala	Diagnosa
	Vulva bengkak dan kemerahan	
40.	- Erupsi kulit daerah leher dan bahu melingkar berbentuk cincin berwarna keputihan - Nafsu makan berkurang - Pendarahan pada keropeng - Permukaan kulit meninggi, berbukul-bukul pada daerah keropeng - Rambut rontok - Tubuh kurus	Ringworm
41.	- Kebuntingan kembar - Kontraksi uterus yang lemah - Sulit melahirkan karena ukuran fetus yang terlalu besar - Sulit melahirkan karena ukuran jalan kelahiran yang kecil - Uterus yang terputar	Distokia
42.	- Suhu tubuh : 38,9°C - Keguguran pada kebuntingan 7 bulan - Kelumpuhan permanen - Nafsu makan berkurang - Peradangan pada buah pelir (sapi jantan) - Peradangan pada saluran sperma (sapi jantan) - Produksi susu menurun - Skrotum bengkak dan membesar (sapi jantan)	Brucellosis
43.	- Berhenti merumput - Daerah otot kaki berwarna merah kehitaman dengan gelembung - Lumpuh dengan pembengkakan otot - Penurunan berat badan - Sakit pada paha apabila dipalpasi	Blackleg
44.	- Suhu tubuh : 38,7°C - Tidak mampu berdiri - Kaki terasa dingin saat diraba - Kelumpuhan pada periode 3 hari setelah melahirkan - Nafsu makan berkurang - Ambruk - Sering terlihat merangkak untuk mengubah posisi - Susah untuk berdiri - Tidak mampu menopang berat badan	Sindrom Sapi Ambruk
45.	- Suhu tubuh : 40,8°C - Bentuk dada terdapat peradangan bronkoparmonia	Ngorok

No.	Gejala	Diagnosa
	• Busung di bawah dagu • Busung di leher bagian bawah sampai gelambir • Busung di tenggorokan • Gemetar • Keluar ingus • Nafsu makan berkurang • Susah bernafas • Tubuh kurus	
46.	• Tidak mampu berdiri • Duduk memanjangkan kaki belakang seperti posisi katak • Gaya berjalan kaku • Menderita kelumpuhan progresif • Mengeluarkan liur yang berlebihan • Nafas berat • Susah bernafas	Botulisme
47.	• Celah kuku mengeluarkan cairan putih keruh • Kuku berbau busuk • Kuku membusuk berwarna merah kehitaman • Kulit kuku mengelupas • Pincang hingga lumpuh • Sekitar celah kuku membengkak • Tumbuh benjolan pada kuku yang menimbulkan rasa sakit	Radang Kuku
48.	• Suhu tubuh : 39,7 °C • Nafsu makan menurun • Diare • Hidung berair • Ambruk/ tidak dapat berdiri • Keluar leleran dari mata • Rambut rontok	Demam Tiga Hari
49.	• Suhu tubuh : 38,9 °C • Dehidrasi • Kelemahan anggota gerak • Depresi • Kelumpuhan pada periode 3 hari setelah melahirkan • Nafsu makan berkurang • Ruminasi berhenti	Kelumpuhan
50.	• Aborsi pada bulan keempat • Kematian dini pada embrio • Penurunan angka konsepsi	Vibriosis

No.	Gejala	Diagnosa
51.	• Tidak mampu berdiri • Flank pada bagian kiri atas membesar • Depresi • Hidung kering • Nafsu makan hilang • Sekitar celah kuku membengkak • Perut kembung berisi udara • Rambut kusam dan berdiri • Sering kencing • Susah bernafas	Kembung
52.	• Suhu tubuh : 39,4°C • Aborsi pada bulan ke 8 kebuntingan • Batuk berdahak • Keluar leleran dari lubang hidung • Daerah otot kaki berwarna merah kehitaman dengan gelembung	Aspergilosis
53.	• Nafsu makan menurun • Suhu tubuh meningkat • Lesu • Malas bergerak	Idigesti Sederhana
54.	• Suhu tubuh : 39,7°C • Kulit kuku mengelupas • Lepuh sekitar batas atas kuku dan ambing susu • Tonjolan bulat pada gusi, lidah dan pangkal lidah • Malas bergerak • Nafsu makan berkurang • Pincang saat berjalan	Penyakit Mulut dan Kuku
55.	• Suhu tubuh : 39,7°C • Anemia • Gelisah • Infestasi larva di saluran reproduksi • Nafsu makan berkurang • Penurunan berat badan • Peradangan pada luka di tempat infestasi larva	Belatungan
56.	• Suhu tubuh : 40,3°C • Terjadi radang kulit • Keluar leleran dari hidung dan mata • Kelumpuhan sebelum mati • Kornea mata keruh • Mata dan hidung mengeluarkan leleran kental dan mukopurulon	Malignan Chataral Fever

No.	Gejala	Diagnosa
	- Mengeluarkan air liur yang berlebihan - Moncong kering dan pecah-pecah berisi nanah - Nafsu makan hilang - Otot-otot gemetar - Peradangan mulut dan lepuhan dipermukaan lidah - Selaput lendir mulut kemerahan - Tubuh kurus	
57.	- Suhu tubuh : 39,7°C - Dehidrasi - Denyut jantung terasa lemah dan lebih cepat - Diare - Nafsu makan berkurang - Sakit di daerah perut	Asidosis Rumen
58.	- Suhu tubuh : 39,5°C - Depresi - Nafsu makan berkurang - Bau plasenta yang mulai membusuk - Kontraksi uterus yang lemah - Pernafasan yang cepat dan dangkal - Produksi susu menurun - Respirasi cepat - Vulva bengkak dan kemerahan	Retensi Sederhana
59.	- Suhu tubuh : 38,7°C - Suhu tubuh naik turun - Diare bercampur darah - Diare berlendir - Erosi pada selaput lendir hidung, lidah bibir, gusi dan bagian belakang maksila - Mengeluarkan air liur yang berlebihan - Nafsu makan hilang	Diare Ganas Menahun
60.	- Suhu tubuh : 40,3°C - Anemia - Busung di daerah dagu dan anggota gerak - Jalan sempoyongan, kejang dan berputar-putar - Keluar cairan dari hidung dan mata - Nafsu makan berkurang - Rambut patah dan rontok - Selaput lendir menguning - Tubuh kurus	Surra
61.	- Suhu tubuh : 39,7 °C - Tubuh kurus	Cacingan

No.	Gejala	Diagnosa
	• Lesu • Rambut rontok dan kusam • Diare encer berwarna kekuningan • Dehidrasi • Anoreksia	
62.	• Suhu tubuh : 39,7 °C • Nafsu makan menurun • Diare • Hidung berair • Ambruk/ tidak dapat berdiri • Keluar leleran dari mata • Rambut rontok	Cacingan
63.	• Nafsu makan tidak ada • Lesu • Tinja berbentuk bulatan keras	Indigesti Sederhana
64.	• Suhu tubuh : 39 °C • Lesu • Anus kotor • Feses encer • Keluar leleran serous dari hidung • Nafsu makan hilang • Hipersaliva	Demam Tiga Hari
65.	• Tinja berbentuk bulat keras • Nafsu makan tidak ada • Lesu	Indigesti Sederhana
66.	• Suhu tubuh : 38,6 °C • Bulu berdiri • Tubuh kurus • Lesu • Feses tidak berbentuk • Anus kotor	Cacingan
67.	• Suhu tubuh : 39 °C • Lesu • Suara perkusi rumen terdengar pekak • Feses lunak kehitaman • Anus kotor	Indigesti Sederhana
68.	• Suhu tubuh : 38,9 °C • Nafsu makan menurun • Lemah • Kulit dan rambut kotor • Keluar leleran di hidung	Indigesti Sederhana

No.	Gejala	Diagnosa
69.	- Suhu tubuh : 39,5°C - Nafsu makan berkurang - Bengkak daerah pinggang dan kelenjar getah bening - Bengkak daerah perut - Keluar darah dari hidung dan mulut - Susah bernafas	Antrak
70.	- Batuk berkepanjangan - Bengkak kelenjar getah bening - Susah bernafas	Tuberkulosis
71.	- Inflamasi pada bagian dalam mata - Peningkatan sensitivitas terhadap cahaya - Ulkus pada kornea - Buta - Uterus terputar	Pink Eye
72.	- Nafsu makan menurun - Diare encer akut - Berat badan menurun - Produksi susu menurun - Kotoran berbau busuk - Tubuh kurus	Johne Disease
73.	- Suhu tubuh : 39,5°C - Nafsu makan berkurang - Diare bercampur darah - Lemah - Produksi susu menurun	Salmonelosis
74.	- Kerak berwarna abu pada permukaan kulit - Lepuh-lepuh bernanah pada kulit - Rambut mudah patah	Kudisan Menular
75.	- Pendarahan pada keropeng - Rambut rontok - Respirasi cepat	Ringworm
76.	- Kontraksi uterus lemah - Sulit melahirkan karena ukuran fetus yang terlalu besar - Sulit melahirkan karena ukuran jalan kelahiran yang kecil - Uterus terputar	Distokia
77.	- Suhu tubuh : 39,3°C - Nafsu makan berkurang - Limpa glandula membengkak - Susah bernafas	Brucellosis

No.	Gejala	Diagnosa
	Darah keluar dari mulut, hidung dan telinga	
78.	- Berhenti merumput - Daerah otot kaki berwarna merah kehitaman dengan gelembung - Lumpuh dengan pembengkakan otot - Penurunan berat badan - Sakit pada paha apabila dipalpasi	Blackleg
79.	- Suhu tubuh : 40,2°C - Nafsu makan hilang - Susah bernafas - Keluar ingus - Busung di leher bagian bawah sampai gelambir - Busung di bawah dagu - Busung di tenggorokan	Ngorok
80.	- Susah bernafas - Nafas berat - Mengeluarkan air liur berlebihan - Mengalami kelumpuhan progresif - Kelumpuhan otot-otot pernafasan - Gaya berjalan kaku - Duduk memanjangkan kaki kebelakang seperti posisi kaki katak	Botulisme
81.	- Pincang - Kuku membusuk berwarna merah kehitaman - Celah kuku bengkak dan mengeluarkan cairan putih keruh - Kuku berbau busuk - Kulit kuku mengelupas - Sekitar celah kuku membengkak - Tumbuh benjolan pada kuku yang menimbulkan rasa sakit	Radang Kuku
82.	- Suhu tubuh : 38,8°C - Tidak mampu berdiri - Nafsu makan berkurang - Gelisah - Ruminasi berhenti	Kelumpuhan
83.	- Kematian dini pada embrio - Aborsi pada bulan keempat - Penurunan angka konsepsi	Vibriosis
84.	- Tidak mampu berdiri - Flank bagian atas kiri membesar	Kembung

No.	Gejala	Diagnosa
	• Gelisah • Hidung sapi kering • Nafsu makan berkurang • Perut terasa keras • Roboh • Sering kencing • Susah bernafas	
85.	• Suhu tubuh : 39,3°C selama 3 hari • Suhu tubuh naik turun • Nafsu makan berkurang • Diare berlendir • Diare bercampur darah	Diare Ganas Menahun
86.	• Suhu tubuh : 39,6°C • Aborsi pada bulan ke-7 kebuntingan • Batuk berdahak • Keluar leleran dari lubang hidung • Pernafasan cepat dan dangkal	Aspergilosis
87.	• Suhu tubuh : 39,5°C • Kulit kuku mengelupas • Lepuh di sekitar batas atas kuku dan ambing susu • Tukak bulat pada gusi, lidah dan pangkal lidah • Malas bergerak • Nafsu makan berkurang • Peradangan lidah dan mulut bagian dalam • Pincang saat berjalan	Penyakit Mulut dan Kuku
88.	• Suhu tubuh : 40,3°C • Kornea mata keruh • Mata dan hidung mengeluarkan leleran kental dan mukopurulon • Kelumpuhan sebelum mati • Mengeluarkan air liur berlebihan • Moncong kering dan pecah-pecah terisi nanah • Nafsu makan hilang • Otot-otot gemetaran • Peradangan mulut dan lepuhan di permukaan lidah • Selaput lendir mulut kemerahan	Malignan Chataral Fever
89.	• Suhu tubuh : 39,8°C • Denyut jantung terasa lemah dan lebih cepat • Diare • Feses keras dan jumlahnya sedikit • Suhu tubuh meningkat	Asidosis Rumen

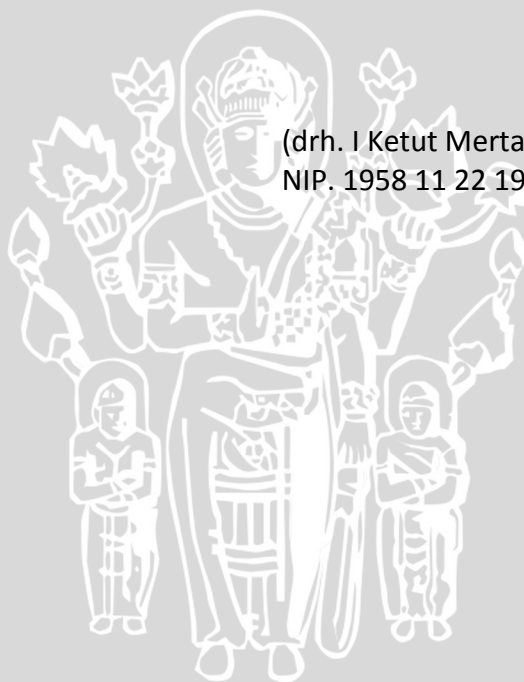
No.	Gejala	Diagnosa
	• Lemah • Malas bergerak • Nafsu makan berkurang • Rumen penuh dank eras • Ruminasi berhenti • Sakit di daerah perut	
90.	• Suhu tubuh : 39,8°C • Nafsu makan berkurang • Selaput fetus menggantung di luar alat kelamin selama lebih dari 12 jam • Gelisah • Bau plasenta yang mulai membusuk • Kontraksi uterus yang melemah • Produksi susu menurun • Respirasi cepat • Vulva bengkak dan kemerahan	Retensi Plasenta
91.	• Suhu tubuh : 40,4°C • Anemia • Busung di daerah dagu dan anggota gerak • Lemah • Jalan sempoyongan, kejang dan berputar-putar • Keluar cairan dari hidung dan mata • Nafsu makan berkurang • Rambut patah atau rontok • Selaput lendir menguning	Surra
92.	• Suhu tubuh : 38,9°C • Keguguran pada kebuntingan 7 bulan • Kelumpuhan permanen • Nafsu makan berkurang • Peradangan pada buah pelir (sapi jantan) • Peradangan pada saluran sperma (sapi jantan) • Produksi susu menurun • Skrotum bengkak dan membesar (sapi jantan)	Brucellosis
93.	• Berhenti merumput • Daerah otot kaki berwarna merah kehitaman dengan gelembung • Lumpuh dengan pembengkakan otot • Penurunan berat badan • Sakit pada paha apabila dipalpasi	Blackleg
94.	• Suhu tubuh : 40,8°C • Bentuk dada terdapat peradangan bronkoparmonia	Ngorok

No.	Gejala	Diagnosa
	• Busung di bawah dagu • Busung di leher bagian bawah sampai gelambir • Busung di tenggorokan • Gemetar • Keluar ingus • Nafsu makan berkurang • Susah bernafas • Tubuh kurus	
95.	• Tidak mampu berdiri • Duduk memanjangkan kaki belakang seperti posisi katak • Gaya berjalan kaku • Menderita kelumpuhan progresif • Mengeluarkan liur yang berlebihan • Nafas berat • Susah bernafas	Botulisme
96.	• Celah kuku mengeluarkan cairan putih keruh • Kuku berbau busuk • Kuku membusuk berwarna merah kehitaman • Kulit kuku mengelupas • Pincang hingga lumpuh • Sekitar celah kuku membengkak • Tumbuh benjolan pada kuku yang menimbulkan rasa sakit	Radang Kuku
97.	• Suhu tubuh : 38,9°C • Dehidrasi • Kelemahan anggota gerak • Depresi • Kelumpuhan pada periode 3 hari setelah melahirkan • Nafsu makan berkurang • Ruminasi berhenti	Kelumpuhan
98.	• Aborsi pada bulan keempat • Kematian dini pada embrio • Penurunan angka konsepsi	Vibriosis
99.	• Tidak mampu berdiri • Flank pada bagian kiri atas membesar • Depresi • Hidung kering • Nafsu makan hilang • Sekitar celah kuku membengkak • Perut kembung berisi udara	Kembung

No.	Gejala	Diagnosa
	• Rambut kusam dan berdiri • Sering kencing • Susah bernafas	
100.	• Suhu tubuh : 39,4°C • Aborsi pada bulan ke 8 kebuntingan • Batuk berdahak • Keluar leleran dari lubang hidung • Daerah otot kaki berwarna merah kehitaman dengan gelembung	Aspergilosis

Tabanan, 2 Oktober 2015

(drh. I Ketut Mertanadi)
NIP. 1958 11 22 1991 03 1 002



LAMPIRAN E NILAI DENSITAS OPTIMAL

[illegible]

	A	BR	M	TB	BL	PE	JD	S	BT	RK	V	N	K	SA	DT
21	0	0	0	0,81	0	0	0	0	0	0	0	0	0	0	0
22	0	0,192	0,488	0	0	0	0	0	0	0	0	0	0	0	0
23	0	0,647	0	0	0	0	0	0	0	0	0	0	0	0	0
24	0	0	0	0	0	0	0	0	0	0	0	0,838	0	0	0
25	0	0	0	0	0	0	0	0	0	0	0	0	0,857	0	0
26	0	0	0	0	0,768	0	0	0	0	0	0	0	0	0	0
27	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
28	0	0	0	0	0	0	0	0	0	0	0	0,951	0	0	0
29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0	0	0	0,773	0	0	0
31	0	0	0	0	0	0	0	0	0	0	0	0,834	0	0	0
32	0	0,465	0	0	0	0	0	0	0	0	0	0	0	0	0
33	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
34	0	0	0	0	0,884	0	0	0	0	0	0	0	0	0	0
35	0,642	0	0	0	0	0	0	0	0	0	0	0	0	0	0
36	0	0	0	0	0	0	0	0	0	0	0	0	0,577	0	0
37	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0,94
38	0,543	0	0,495	0	0	0	0	0,912	0	0	0	0	0	0	0,884
39	0,787	0	0	0	0	0	0	0	0	0	0	0,839	0	0	0
40	0	0,536	0,542	0	0	0	0	0	0	0	0	0	0,502	0,447	0
41	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
42	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
43	0,262	0	0	0	0	0	0	0	0	0	0	0	0	0	0
44	0	0	0	0	0	0	0	0,803	0	0	0	0	0	0	0

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

	BV	MC	MK	HV	R	AS	RP	D	IS	KB	AR	SR	C	KM	B
49	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
50	0,823	0	0	0	0	0	0	0	0	0	0	0	0	0	0
51	0	0	0	0	0,846	0	0	0	0	0	0	0	0	0	0
52	0	0	0	0	0	0	0	0	0	0	0	0	0,554	0	0
53	0	0	0	0	0	0	0	0	0,854	0	0	0	0	0	0
54	0	0	0	0	0	0	0	0	0,94	0	0,895	0	0	0	0
55	0	0	0	0	0	0	0	0	0	0,801	0	0	0	0	0
56	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
57	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
58	0,476	0	0	0	0	0	0,615	0	0	0,806	0	0	0,89	0	0,521
59	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
60	0	0	0	0	0	0	0	0	0	0	0	0	0	0,859	0
61	0	0	0	0	0	0	0	0	0	0,85	0	0	0	0	0
62	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
63	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
64	0	0	0	0	0	0	0	0	0	0	0,06	0	0	0	0
65	0	0	0	0	0	0	0	0	0,935	0	0	0	0	0	0
66	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0,812
67	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
68	0	0	0	0	0	0	0	0	0	0	0	0,766	0	0	0
69	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
70	0	0,629	0	0	0	0	0	0	0	0	0	0	0	0	0
71	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
72	0	0	0	0	0	0	0	0,766	0	0	0	0	0	0	0

	BV	MC	MK	HV	R	AS	RP	D	IS	KB	AR	SR	C	KM	B
73	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
74	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
75	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
76	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
77	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
78	0	0	0	0	0	0	0	0	0	0	0	0	0,776	0	0
79	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
80	0	0	0	0,852	0	0	0	0	0	0	0	0	0	0	0
81	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
82	0	0,839	0	0	0	0	0	0	0	0	0	0,809	0	0	0
83	0,388	0	0	0,84	0	0,834	0	0	0	0	0	0	0	0	0
84	0	0	0	0,744	0	0	0	0	0	0	0	0	0	0	0
85	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
86	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
87	0	0,858	0	0	0	0	0	0	0	0	0	0	0	0	0
88	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
89	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
90	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
91	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
92	0	0	0	0	0	0	0	0	0	0	0	0	0	0,822	0
93	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0,888
94	0	0	0	0,749	0	0	0	0	0	0	0	0	0	0	0
95	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
96	0	0	0	0	0	0	0	0	0,829	0	0	0	0	0	0

[illegible]

	BV	MC	MK	HV	R	AS	RP	D	IS	KB	AR	SR	C	KM	B
121	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
122	0	0	0	0	0	0	0	0	0	0	0	0	0,839	0	0
123	0	0	0	0	0	0	0	0	0	0	0	0	0	0,88	0
124	0	0	0	0	0,963	0	0	0	0	0	0	0	0	0	0
125	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
126	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
127	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0,628
128	0	0	0,857	0	0	0	0	0	0	0	0	0	0	0	0
129	0	0,853	0	0	0	0	0	0	0	0	0	0	0	0	0
130	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
131	0	0	0	0,873	0	0	0	0	0	0	0	0	0	0	0
132	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
133	0	0	0	0	0,814	0	0	0	0	0	0	0	0	0	0
134	0	0	0	0	0	0,807	0,667	0	0	0	0,837	0	0	0	0
135	0	0	0	0	0	0	0	0	0	0,897	0	0	0	0	0
136	0	0	0	0	0	0	0	0	0,874	0,729	0	0	0	0	0
137	0	0	0,794	0	0	0	0	0	0	0	0	0	0	0	0
138	0	0	0	0	0	0	0,592	0	0	0	0	0	0	0	0,472
139	0	0	0	0	0	0	0	0	0	0,465	0	0	0,842	0	0
140	0	0	0	0	0,855	0	0	0	0,654	0	0	0,422	0,902	0,871	0
141	0	0	0	0	0,81	0	0	0	0	0	0	0	0	0	0
142	0	0	0	0	0	0	0	0	0	0	0	0	0,769	0	0
143	0	0	0	0	0	0	0	0	0	0	0,887	0	0	0	0
144	0	0	0	0	0	0	0,452	0	0	0	0	0	0	0	0

[illegible]

[illegible]