KLASIFIKASI RESIKO PENYAKIT STROKE DENGAN MENGGUNAKAN ALGORITMA SELF ORGANIZING MAP(SOM)

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:
Septa Hendra Prasetya Utama
NIM:0910683088

PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2016

repository.ub.ac.id
UNIVERSITAS BRAWIJAYA

PENGESAHAN

KLASIFIKASI RESIKO PENYAKIT STROKE DENGAN MENGGUNAKAN ALGORITMA SELF ORGANIZING MAPS (SOM)

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :
Septa Hendra Prasetya Utama
NIM:0910683088

Skripsi ini telah diuji dan dinyatakan lulus pada
22 Agustus 2016
Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I	Dosen Pembimbing II
Imam Cholissodin, S.Si., M.Kom.	Edy Santoso, S.Si., M.Kom.
NIK: 201201 850719 1 001	NIP:19740414 200312 1 004

Mengetahui
Ketua Jurusan Teknik Informatika

Tri Astoto Kurniawan, S.T, M.T, Ph.D
NIP: 19670801 199203 1 001

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, 22 Agustus 2016

Septa Hendra Prasetya Utama
NIM: 0910683088

KATA PENGANTAR

Puji dan syukur Alhamdulillah penulis panjatkan kehadirat Allah SWT , yang telah melimpahkan rahmat dan anugerah-Nya kepada penulis sehingga dapat menyelesaikan skripsi ini yang berjudul "**Klasifikasi Resiko Penyakit Stroke Dengan Menggunakan Algoritma Self Organizing Map (SOM)".**

Pada penyusunan Skripsi ini tidak semata-mata hasil kerja Penulis sendiri,melainkan juga berkat bimbingan dan dorongan dari pihak-pihak yang telah membantu, baik secara materi maupun non materi. Maka dari itu penulis ingin mengucapkan banyak terima kasih yang tak terhingga serta penghargaan yang setinggi tinggi nya kepada orang-orang yang telah membantu penulis secara langsung maupun tidak langsung kepada yang terhormat :

1. Imam Cholissodin, S.Si., M.Kom., selaku Pembimbing I yang telah banyak memberi semangat, bimbingan, dukungan moril, ilmu dan bantuan dalam proses penyelesaian skripsi ini
2. Edy Santoso, S.Si., M.Kom., selaku Pembimbing II yang juga memberikan masukan, bimbingan dan kemudahan dalam penyelesaian proses skripsi ini.
3. Agus Wahyu Widodo, S.T., M.Cs., selaku Ketua Program Studi yang banyak memberikan kemudahan, arahan dan motivasi dalam penyelesaian skripsi ini.
4. Ir. Heru Nurwarsito, M.Kom., selaku Dosen Pembimbing Akademik yang banyak memberikan dukungan moril dan masukan selama masa perkuliahan.
5. Ibu Wiwin Lukitohadi dan Bapak Prasetyo Iskandar atas dukungan moril dan arahan untuk pengajuan judul skripsi sampai dengan masukan selama proses pengerjaan skripsi, hingga selesainya skripsi ini.
6. Orang tua penulis Bapak Achmad Fauzan Adhim dan Ibu Siti Alfiah yang memberikan doa, nasehat, didikan dan kesabarannya.
7. Adikku Ellia Shafira Altama dan Anisa Cahya Muliatama atas keceriaan yang diberikan saat proses pengerjaan skripsi.
8. Ilham Ubaidillah S.Kom, Raviqul Haidir Francasmara S.Kom dan Galih Julianto Purnomo sebagai Pembimbing III, Pembimbing IV dan Pembimbing V.
9. Om I Made Sutharta, Tante Yuniarti, I Putu Arie Bayu Antara dan Kadek Fiantara yang sudah banyak direpotkan. Terima Kasih atas kebaikan dan keramahannya.
10. Teman temanku Keluarga Besar TPL Legend 2009 dan Wisata Kuliner 2009 Tony Gombloh, Bagus Pijet, Yusup Oktofani, Aang Samsudy, Bang

Billy, Pace Aggy, Wildan Nchip, Wong Gendeng Gerrygerro, Prima Tahta, Rio Alif, Putu Arya, Heri Kumis, Martal Hansa Luberan Doripson, Mami Kiki, Alvita, Monic, Ulhaq, Yan. Terima kasih atas segala motivasi, semangat, dukungan moril dan telah menjadi teman yang baik, teman serius maupun teman bersenda gurau selama di Malang. Long live my family.

11. Teman teman penghuni Wisma 46 A Juni, Nanda, Apip , Hilman, Dito.
12. Teman teman alumni SMAN 1 Surabaya XC dan IPA 2 Rani Widya, Dessy, Oggy, Yogi, Irwan, Gandhos, Ove, Adi, Nadia terima kasih atas support dan dukungan yang telah kalian berikan.
13. Semua pihak yang tidak dapat disebutkan satu persatu yang telah terlibat baik secara langsung maupun tidak langsung dalam penulisan skripsi ini.

Penulis menyadari bahwa penulisan skripsi ini jauh dari kata sempurna. Maka dari itu penulis mengharapkan masukan berupa kritik dan juga saran demi tercapainya kesempurnaan dalam penulisan skripsi ini. Semoga skripsi ini dapat memberikan banyak manfaat bagi semua pihak

Malang, 22 Agustus 2016

Septa Hendra Prasetya Utama

ABSTRAK

Dampak yang ditimbulkan seseorang terkena penyakit *stroke* adalah kelumpuhan sebagian organ tubuh, keseluruhan bahkan berujung pada kematian. Tingginya angka penderita stroke diakibatkan kurangnya kesadaran masyarakat pada faktor resiko yang dapat memicu penyakit ini. Hal ini sebenarnya dapat dicegah dengan identifikasi tingkat resiko penyakit *stroke* pada tiap individu secara dini. Selama ini proses deteksi tingkat resiko penyakit *stroke* menggunakan pemeriksaan melalui pendataan umur pasien dan juga cek darah. Dalam proses tersebut didapatkan data umur dan kandungan dalam darah berupa kolesterol total, HDL, LDL dan *trigliserida*. Kendalanya adalah pada proses klasifikasi masih menggunakan cara manual sehingga ketika banyak data pasien yang diidentifikasi masih membutuhkan proses yang relatif lama. Maka dari itu dibutuhkan sebuah pemanfaatan teknologi informasi sebagai solusi untuk mengatasi masalah tersebut. Dalam penelitian ini menggunakan algoritma *self-organizing maps* untuk mengklasifikasikan tingkat resiko penyakit *stroke* dari banyak data klinis perorangan. Dari 200 data klinis yang di ujikan pada pengujian perbandingan data latih dan data uji 50 : 50 dan didapatkan rata-rata akurasi terbaik sebesar 41 % dengan menggunakan learning rate awal 0.1 pengurang learning rate 0.1, learning rate minimum 0.001 dan maksimum iterasi sebesar 10000.

Kata kunci: *Self Organizing Map*, *Stroke*, *Stroke Iskemik*.

repository.ub.ac.id

ABSTRACT

The impact of a person affected by stroke is partial or total body paralysis and even could lead to death. The high number of patients with stroke caused by a lack of public awareness on the risk factors that can trigger this disease.This disease can be prevented by identifying the level of risk of stroke in individuals at early stage. The risk detection of stroke is using through data collection patient's age and blood check. The data obtained from previous process is the contents in the blood such as total cholesterol, HDL, triglycerides and LDL. It could be a problem when there are a lot of data need to be processed because that would take a relatively long time if the identification process is done manually. Therefore the use of information technology could be a solution to overcome said problem. This research use self-organizing maps algorithm to classify the risk level of stroke from many individual's clinical data. There are 200 clinical data which tested on comparative training data and test data 50: 50 using and get the best average accuracy 41% by using the initial learning rate 0.1, deduction learning rate 0.1, learning rate 0.001 minimum and maximum iteration of 10000.

Keywords : Self Organizing Map, Stroke, Stroke Iskemik.

UNIVERSITAS BRAWIJAYA

DAFTAR ISI

PENGESAHAN ii
PERNYATAAN ORISINALITAS iii
KATA PENGANTAR iv
ABSTRAK vi
ABSTRACT vii
DAFTAR ISI viii
DAFTAR TABEL x
DAFTAR GAMBAR xi
DAFTAR LAMPIRAN xii
BAB 1 PENDAHULUAN 1
1.1 Latar belakang 1
1.2 Rumusan masalah 2
1.3 Tujuan 3
1.4 Manfaat 3
1.5 Batasan masalah 3
1.6 Sistematika pembahasan 3
BAB 2 LANDASAN KEPUSTAKAAN 5
2.1 Kajian Pustaka 5
2.2 Stroke 10
2.2.1 Faktor Faktor Yang Mempengaruhi 10
2.2.2 Faktor Faktor Yang Mempengaruhi 11
2.3 Self Organizing Map (SOM) 12
2.3.1 Algoritma SOM 13
2.3.2 Normalisasi Data 14
2.3.3 Learning Rate 14
2.3.4 Evaluasi 15
BAB 3 METODOLOGI 16
3.1 Metode Penelitian 16
3.1.1 Studi Literatur 16
3.1.2 Pengumpulan Data 17

3.1.3 Analisis dan Perancangan Sistem 17
3.1.4 Implementasi Sistem 17
3.1.5 Pengujian dan Analisis Sistem 18
3.1.6 Evaluasi dan Analisis Hasil 18
3.2 Perancangan 19
3.2.1 Deskripsi Umum Sistem 19
3.2.2 Perancangan Proses 20
3.2.3 Perhitungan Manual 27
3.2.4 Perancangan *Interface* 35
3.2.5 Perancangan Uji Coba dan Evaluasi 36
BAB 4 IMPLEMENTASI 40
4.1 Spesifikasi Lingkungan Implementasi 40
4.1.1 Spesifikasi Perangkat Keras 40
4.1.2 Spesifikasi Perangkat Lunak: 40
4.2 Batasan Aplikasi 40
4.3 Implementasi Algoritma 41
4.3.1 Proses Pembelajaran SOM 41
4.4 Implementasi Antar Muka 47
BAB 5 PENGUJIAN DAN ANALISIS 50
5.1 Hasil dan Analisis Pengujian Tingkat Akurasi Terhadap Parameter *Learning rate* awal 50
5.2 Hasil dan Analisis Pengujian Tingkat Akurasi Terhadap Parameter Pengurang *Learning rate*. 51
5.3 Hasil dan Analisis Pengujian Tingkat Akurasi Terhadap Parameter Learning Rate *Minimum.* 52
5.4 Pengujian Tingkat Akurasi Terhadap Jumlah Iterasi *Maksimum* 53
5.5 Pengujian Tingkat Akurasi Terhadap Perbandingan Jumlah Data Latih dan Data Uji 54
BAB 6 PENUTUP 56
6.1 Kesimpulan 56
6.2 Saran 56
DAFTAR PUSTAKA 57

DAFTAR TABEL

Tabel 2.1 Kajian Pustaka 6
Tabel 2.2 Kolesterol Total 11
Tabel 2.3 Kolesterol HDL (*High Density Lipoprotein*) 12
Tabel 2.4 Kolesterol LDL (Low Density-Lipoprotein) 12
Tabel 3.1 Data Latih/Data *Training* 27
Tabel 3.2 Data Testing 28
Tabel 3.3 Mencari Nilai Minimum dan Maksimum 28
Tabel 3.4 Data Hasil Normalisasi 28
Tabel 3.5 Bobot Input 29
Tabel 3.6 Hasil Jarak Antara Data Dengan Bobot 30
Tabel 3.7 Hasil Update Bobot ($\boldsymbol{wij}$) 31
Tabel 3.8 Bobot Data Pengujian ($\boldsymbol{wij}$) 34
Tabel 3.9 Hasil Jarak Dan Klasifikasi Stroke 35
Tabel 3.10 Skenario Pengujian Terhadap Parameter Learning Rate Awal 36
Tabel 3.11 Skenario Pengujian Penentuan Parameter Pengurang *Learning Rate* 37
Tabel 3.12 Skenario Pengujian *Learning Rate Minimum* 38
Tabel 3.13 Skenario Pengujian Jumlah *Iterasi Maksimum* 38
Tabel 3.14 Skenario Pengujian Perbandingan data latih dan data uji 39
Tabel 5.1 Tingkat Akurasi Terhadap Parameter *Learning Rate* Awal 50
Tabel 5.2 Tingkat Akurasi Terhadap Parameter Pengurang *Learning Rate* 52
Tabel 5.3 Tingkat Akurasi Terhadap Parameter *Learning Rate* Minimum 53
Tabel 5.4 Tingkat Akurasi Terhadap Jumlah *Iterasi Maksimum* 54
Tabel 5.5 Tingkat Akurasi Terhadap Perbandingan Data Latih Dan Data Uji 55

DAFTAR GAMBAR

Gambar 2.1 Penyumbatan pembuluh darah di otak penyebab stroke iskemik ... 10
Gambar 3.1 Diagram Alir Penelitian ... 16
Gambar 3.2 Diagram Alir SOM ... 20
Gambar 3.3 Diagram Alir Normalisasi Data ... 21
Gambar 3.4 Diagram Alir Proses Training SOM ... 22
Gambar 3.5 Diagram Alir Proses Testing SOM ... 23
Gambar 3.6 Diagram Alir Proses Inisialisasi Bobot ... 24
Gambar 3.7 Diagram Alir Jarak Antara Data Dengan Bobot ... 25
Gambar 3.8 Proses Update Bobot ... 26
Gambar 3.9 Proses Menghitung Akurasi Terbaik ... 27
Gambar 3.10 Perancangan Interface ... 35
Gambar 5.1 Grafik Pengaruh Parameter *Learning Rate* Awal Terhadap Akurasi ... 51
Gambar 5.2 Grafik Pengaruh Pengurang *Learning Rate* Terhadap Akurasi ... 52
Gambar 5.3 Grafik Pengaruh Learning Rate Minimum Terhadap Akurasi ... 53
Gambar 5.4 Grafik Pengaruh Maksimum Iterasi Terhadap Akurasi ... 54
Gambar 5.5 Pengaruh perbandingan data latih dan data uji terhadap akurasi ... 55

DAFTAR LAMPIRAN

LAMPIRAN DATASET 58

BAB 1 PENDAHULUAN

1.1 Latar belakang

Stroke merupakan salah satu jenis penyakit gangguan fungsi syaraf yang diakibatkan oleh gangguan peredaran darah ke otak dan terjadi secara mendadak. Penyakit *stroke* diakibatkan oleh tersumbatnya pasokan *oksigen* dan nutrisi menuju otak, dan mengakibatkan sel saraf di otak mati dan berakibat pada kelumpuhan sebagian tubuh, keseluruhan bahkan kematian (Seto., et. All., 2014). Faktor resiko *stroke* terbagi menjadi dua yaitu faktor yang tidak dapat diubah dan dapat diubah. Faktor yang tidak dapat diubah terdiri dari usia, jenis kelamin, ras, riwayat keluarga dan riwayat *stroke* sebelumnya. Sedangkan faktor yang dapat diubah diantaranya adalah merokok, *hipertensi*, *obesitas*, dan *diabetes mellitus* (Afrizal., 2015).

Berdasarkan data *World Health Organization* atau WHO di seluruh dunia pada tahun 2002 diperkirakan 5,5 juta orang meninggal akibat *stroke* dan diperkirakan tahun 2020 penyakit jantung dan *stroke* menjadi penyebab kematian di dunia (Projoalisastra., 2009). Pada Tahun 2009 Indonesia merupakan negara dengan jumlah penderita *stroke* terbesar di Asia dan menempati urutan ketiga penyebab kematian setelah penyakit jantung dan *kanker* (Yastroki., 2009). Orang Indonesia yang mengalami serangan *stroke* diperkirakan sekitar 500 ribu setiap tahunnya. Berdasarkan jumlah tersebut, sekitar 2,5% meninggal dunia sementara sisanya mengalami kecacatan dari ringan hingga berat (Gemari., 2009).

Jumlah penderita penyakit stroke di Indonesia tahun 2013 berdasarkan diagnosis Tenaga Kesehatan (Nakes) diperkirakan sebanyak 1.236.825 orang (7,0%), sedangkan berdasarkan diagnosis Nakes gejala seseorang terkena stroke diperkirakan sebanyak 2.137.941 orang (12,1%).Berdasarkan diagnosis/gejala, Provinsi Jawa Barat memiliki estimasi jumlah penderita terbanyak yaitu sebanyak 238.001 orang (7,4%) dan 533.895 orang (16,6%) sedangkan provinsi Papua Barat memiliki jumlah penderita paling sedikit yaitu sebanyak 2.007 orang (3,6%) dan 2.995 orang (5,3%) (Pusat., 2013).

Dengan data statistik yang didapatkan dapat disimpulkan bahwa penyakit stroke memiliki potensi memperbesar jumlah penderita atau pasien yang rentan terkena *stroke* dan yang positif terkena *stroke* bertambah setiap tahunnya. Penyakit *stroke* bersifat mendadak dan bisa terjadi sewaktu-waktu, maka dibutuhkan sebuah sistem untuk membantu seseorang dalam mengidentifikasi tanda-tanda yang dapat menyebabkan orang tersebut memiliki potensi terkena *stroke*.Penanganan yang lambat merupakan salah satu penyebab tingginya angka kematian para penderita *stroke*.Penanganan secara *Gold Standart Procedure* atau prosedur wajib menggunakan teknologi *Computed Tomograph scene* terkendala masalah biaya yang mahal dan tidak semua Rumah Sakit memilikinya terutama di Indonesia (Seto., et. al., 2014). Maka dari itu dibutuhkan sistem komputerisasi

yang dibuat bertujuan untuk memberikan prediksi serta mengarahkan kepada *user* agar dapat mengupayakan dirinya terhindar dari penyakit *stroke*. Dalam hal ini bukan hanya orang medis saja yang bisa mengetahui kondisi dan tingkat seseorang terkena penyakit *stroke* akan tetapi orang selain tenaga medis juga dapat mengetahui berdasarkan dari hasil tes kesehatan yang diolah secara otomatis didalam sebuah sistem. Dalam pengolahan data sendiri bisa dilakukan pengklasifikasian dengan metode/algoritma tertentu. Salah satunya adalah algoritma *self organizing maps.*

Penelitian terhadap penyakit *stroke* pernah dilakukan oleh Afrizal Laksita Akbar(2015). Atribut yang digunakan pada penelitian tersebut adalah umur, *cholesterol* total, *High Density-Lipoprotein* (HDL), *Low Density-Lipoprotein* (LDL) dan *Trigliserida*.Hasil yang didapatkan pada penelitian tersebut adalah pengklasifikasian seseorang masuk kedalam kelompok yang Normal,Rentan dan stroke berdasarkan nilai masing-masing atribut. Pada penelitian tersebut dihasilkan akurasi sebesar 0,8939 pada saat panjang fold 5 dengan nilai parameter *epsilon* 0,001 dan *cost* 10 dan standar deviasi terbaik dengan menggunakan *cross validation* dimana dihasilkan 0,1005 pada saat panjang fold 10 dengan nilai parameter *epsilon* 0,001 dan *cost* 10 (Afrizal., 2015).

Self-organizing Maps adalah sebuah metode yang pertama kali dikembangkan oleh Profesor Teuvo Kohonen. Kelebihan metode ini adalah Kecepatan pengolahan data training dan keakuratan dalam proses pengklasifikasian (Gregorius., et. All., 2012). Penelitian dengan metode *self organizing maps* di bidang Kesehatan pernah dilakukan oleh Wahyudi Setiawan dari Universitas Trunojoyo dengan Objek pengenalan *retina* untuk mendeteksi *retinopati diabetika* dengan menggunakan metode *self organizing maps*. Penelitian tersebut menggunakan data training berupa data citra retina yang diolah menggunakan *Operator Laplacia*n. Kemudian dilakukan *ekstraksi* ciri Menggunakan Metode *Principal Component Analysis*. Dalam penelitian tersebut didapatkan hasil berupa *recognizittion rate* sebesar 85% untuk kemungkinan terbaik dan 50% untuk kemungkinan terburuk.

Berdasarkan pada penelitian tersebut Penulis menggunakan algoritma Self-Organizing Map (SOM) untuk mengklasifikasikan data-data yang merupakan tanda-tanda atau gejala-gejala penyakit *stroke,* kemudian data tersebut di proses kedalam suatu bentuk sistem cerdas yang berfungsi untuk mengklasfikasikan resiko seseorang terkena penyakit *stroke*.

1.2 Rumusan masalah

Berdasarkan latar belakang yang telah dikemukakan di atas, maka dirumuskan masalah sebagai berikut

1. Bagaimana implementasi *algoritma self-organizing map* dapat *mengklasifikasikan* seseorang memiliki potensi terkena penyakit *stroke*?
2. Bagaimana tingkat akurasi sistem dalam *mengklasifikasikan* potensi seseorang terkena penyakit *stroke* ?

1.3 Tujuan

Tujuan:

1. Mengimplementasikan *Algoritma Self Organizing Map* untuk *mengklasifikasikan* seseorang memiliki *potensi* terkena penyakit *stroke.*
2. Mengetahui tingkat *akurasi* dari hasil *implementasi Algoritma Self-Organizing Map* untuk *mengklasifikasikan potensi* seseorang terkena penyakit *stroke.*

1.4 Manfaat

Manfaat yang diharapkan dari penelitian ini antara lain :

1. Dapat mengetahui tingkat *resiko* seseorang terkena penyakit *stroke* secara dini.
2. Dapat *mengklasifikasikan* seseorang *berpotensi* terkena penyakit *stroke* dengan *efektif* dan *efisien.*

1.5 Batasan masalah

Agar rumusan permasalahan penelitian berjudul "Klasifikasi Resiko Penyakit *Stroke* Dengan Menggunakan Algoritma *Self-Organizing Map* (SOM)" lebih fokus, maka penulis menentukan batasan masalah sebagai berikut:

1. Data penelitian pada skripsi ini diambil dari Laboratorium Klinik Sejahtera di Kota Probolinggo Jawa Timur.
2. Data dan parameter yang digunakan pada perhitungan berupa parameter karakteristik penyakit stroke berdasarkan hasil Laboratorium darah Kota Probolinggo antara lain : umur, *kolesterol* total, HDL(*High-Density Lipoprotein*), LDL(*Low-Density Lipoprotein*) dan *Trigliserida*.
3. Stroke yang akan dideteksi pada penelitian ini adalah *stroke* yang disebabkan oleh sumbatan (*stroke iskemik*).
4. Algoritma yang digunakan pada penelitian ini adalah *Self-Organizing Map* (SOM).

1.6 Sistematika pembahasan

Sistematika penyusunan laporan ditunjukkan untuk memberikan gambaran dan uraian dari penelitian skripsi secara garis besar yang meliputi beberapa bab, sebagai berikut.

BAB 1 PENDAHULUAN

Bab iniberisi uraian mengenai latar belakang,rumusan masalah, batasan masalah tujuan dan manfaat, serta sistematika penyusunan laporan penelitian skripsi.

BAB 2 LANDASAN KEPUSTAKAAN

Bab ini berisi pengantar, materi kajian pustaka dan teori-teori yang menjadi referensi dalam pelaksanaan penelitian klasifikasi resiko penyakit stroke dengan menggunakan algoritma SOM.

BAB 3 METODOLOGI PENELITIAN DAN PERANCANGAN

Bab ini berisi tentang metode dan langkah kerja yang dilakukan dalam proses perancangan dan implementasi pada pelaksanaan penelitian Klasifikasi resiko penyakit stroke menggunakan algoritma SOM.

BAB 4 IMPLEMENTASI

Bab ini merupakan uraian dan penjelasan implementasi sistem yang berasal dari dasar-dasar teori, metode yang digunakan dan perancangan yang telah dibuat untuk kemudian dibuat menjadi sebuah sistem untuk menyelesaikan permasalahan studi kasus yang diteliti

BAB 5 PENGUJIAN DAN ANALISIS

Dalam bab ini akan dijelaskan mengenai pengujian program dan analisa hasil penelitian.

BAB 6 PENUTUP

Memuat Kesimpulan yang diperoleh dari pembuatan dan pengujian perangkat lunak yang dikembangkan dalam penelitian ini serta saran untuk pengembangan lebih lanjut.

BAB 2 LANDASAN KEPUSTAKAAN

Bab ini membahas tentang kajian pustaka dan dasar teori yang digunakan untuk menunjang penulisan skripsi mengenai Identifikasi penyakit *stroke* dengan menggunakan algoritma *Self Organizing Map*. Beberapa Dasar teori yang dimaksud meliputi Stroke, dan *Self Organizing Map* (SOM).

2.1 Kajian Pustaka

Kajian pustaka pada penelitian ini adalah metode jaringan syaraf tiruan *self organizing map* untuk klasifikasi kualitas tanaman nilam pada citra daun. Kartika Purwandari dari Fakultas Ilmu Komputer Universitas Brawijaya pernah melakukan penelitian menggunakan metode *self organizing maps* untuk klasifikasi kualitas tanaman nilam pada citra daun. Objek penelitian dari kasus ini adalah daun nilam. Daun yang diambil merupakan jenis daun nilam yang berumur 6 bulan sebanyak 50 daun yang berkualitas baik dan 50 daun yang berkualitas jelek. Daun yang memiliki kualitas baik adalah daun yang kecil,tebal dan berwarna sedikit merah kekuningan. Tujuan dari penelitian tersebut adalah untuk mengklasifikasikan antara daun nilam yang berkualitas baik dan daun nilam yang berkualitas buruk.

Tahapan pertama dalam penelitian tersebut adalah input citra daun hasil pemindaian dengan ukuran 200x500 pixel. Kemudian tahap selanjutnya adalah *preprocessing* yang berfungsi mengubah citra menjadi citra *grayscale* dan biner. Proses selanjutnya adalah mengekstraksi fitur dan hasilnya di klasifikasi kedalam algoritma *self organizing maps*. Hasil rata-rata terbaik yang didapatkan dalam pengujian tersebut adalah sebesar 85% dengan menggunakan data latih sebanyak 60, data uji sebanyak 40, *learning rate* awal = 0.1, pengurang *learning rate*=0.8 *learning rate minimum* = 0.001, dan jumlah iterasi maksimum = 12.

Perbedaaan yang dibuat Penulis pada penelitian ini adalah objek data penelitian. Data yang digunakan merupakan data pasien penyakit *stroke*.Data tersebut berbentuk tabel dan terdiri dari umur, *kolesterol* total, kadar *kolesterol* HDL (*High Density-Lipoprotein*), LDL (Low Density-Lipoprotein), *Trigliserida*. Dengan menggunakan algoritma SOM diharapkan proses klasifikasi pasien yang beresiko terkena penyakit stroke dapat diketahui secara akurat.

Tabel 2.1 Kajian Pustaka

No	Judul	Objek	Metode	Hasil
1.	Implementasi Metode Jaringan Syaraf Tiruan *Self Organizing Maps* (SOM) Untuk Klasifikasi Kualitas Tanaman Nilam pada Citra Daun	Objek: • Daun Nilam Input Kriteria: • Citra daun nilam berumur 6 bulan	Metode: *Self Organizing Maps* Proses: 1. Input Citra daun hasil pemindaian dengan ukuran 200x250 piksel 2. Preprocessing citra (mengubah menjadi citra grayscale dan biner) 3. Ekstraksi Fitur (rata-rata merah, hijau, biru dan luas daun) 4. Klasifikasi dengan SOM	• Hasil klasifikasi kualitas daun nilam meliputi kelas baik dan kurang baik
2.	Sistem Pengenalan Retina Menggunakan Self Organizing Map untuk Mend	Objek • *Retinopati Diabetika* Input Kriteria • Citra *Retina*	Metode : *Self Organizing Maps* (SOM) Proses: 1. Pengambilan Data citra retina yang diolah dengan menggunakan *Operator Laplacian.*	• Dihasilkan *Recognizition Rate* sebesar 85% untuk kemungkinan terbaik dan 50% untuk kemungkinan terburuk.

	eteksi Retinopati Diabetika.		2. Ekstraksi Fitur menggunakan metode *Principal Component Analysis* (PCA). *3.* Data biner hasil PCA digunakan sebagai data masukan pada proses pelatihan SOM.	
3.	Klasifikasi Stroke Berdasarkan Kelainan Patologis dengan *Learning Vector Quantization*.	Objek • *Stroke* Input Kriteria : • Data Klinis terdiri dari 32 fitur berisi tentang hasil pemeriksaan fisik, gejala yang dirasakan pasien, riwayat penyakit dan pemeriksaan Laboratorium darah.	Metode : *Learning Vector Quantization* Proses: 1. Input data pasien hasil rekam medis. 2. Selanjutnya dataset yang telah masuk ke basis data, sesuai kebutuhan sistem dibagi menjadi 2. Untuk pembelajaran (*training*) *LVQ* dan untuk uji coba agar tidak saling intervensi. 3. Proses Klasifikasi data dengan	• Hasil penelitian ini memiliki tingkat akurasi sebesar 96 %. Uji diagnosis ditunjukkan dengan nilai Area Under Curve (AUC) sebesar 0,952 yang tergolong dalam kategori excellent. • Keputusan pasien menderita *strokeinfark*atau *hemmorhagic*.

			menggunakan LVQ. 4. Pengujian Akurasi 5. Pengujian dan Klasifikasi 6. Hasil	
4.	Implementasi Algoritma SVM (*Support Vector Machine*) untuk mengetahui tingkat resiko penyakit stroke.	Objek • Stroke Iskemik Input Kriteria • Umur • *Cholesterol total* • High Density-Lipoprotein (HDL) • Low Density-Lipoprotein(LDL) • *Trigliserida*	Metode : *Support Vector Mchine,Simplified Sequential Minimal Optimization(Simplified SMO), K-fold Cross Validation.* Proses: 1. Input data hasil rekam medis 2. Set Iterasi. 3. Proses klasifikasi di dalam sistem menggunakan metode *SVM*untuk membuat hyperplane/ garis pemisah yang optimal. 4. Optimasi menggunakan *Simplified SMO*. 5. Hasil Optimasi di proses menggunak	• Keputusan pasien berada dalam kategori normal, rentan dan *stroke*. • Akurasi terbaik adalah pengujian *crossvalidation*di mana dihasilkan akurasi sebesar 0,8939 pada saat panjang fold 5 dengan nilai parameter *epsilon* 0,001 dan *cost* 10. • Standar *deviasi* terbaik dengan menggunakan *cross validation* dimana dihasilkan standar *deviasi* terbaik 0,1005 pada saat panjang fold 10 dengan nilai parameter *epsilon* 0,001 dan *cost* 10

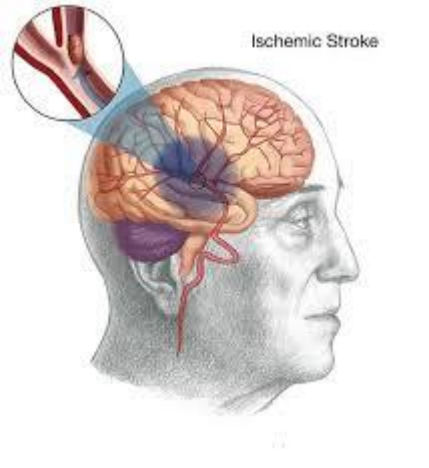
			an K-fold *CrossValidation* untuk menghitung akurasi.	

2.2 Stroke

Stroke atau *Cerebrovascular disease* menurut *World Health Organization* (WHO) adalah tanda tanda klinis yang berkembang cepat akibat gangguan fungsi otak *fokal* atau *global* karena adanya sumbatan atau pecahnya pembuluh darah di otak dengan gejala yang berlangsung selama 24 jam atau lebih (Seto., et. All., 2014). Ada 2 macam jenis stroke yaitu stroke *hemoragik* dan stroke *iskemik* (nonhemoragik).

Stroke *hemoragik* merupakan jenis *stroke* yang disebabkan karena pecahnya pembuluh darah di otak. Sedangkan *stroke Iskemik* disebabkan karena adanya penyumbatan aliran darah di otak. Hampir 80% penderita *stroke* di Indonesia adalah *stroke iskemik* (Alfred., 2007).

Faktor resiko *stroke* dibagi menjadi dua, yaitu faktor resiko yang tidak dapat dirubah dan faktor resiko yang dapat dirubah.Faktor yang tidak dapat dirubah adalah usia, jenis kelamin, ras , riwayat keluarga, dan riwayat stroke sebelumnya. Sedangkan faktor yang dapat dirubah disebabkan karena gaya hidup adalah *hipertensi,diabetes melitus, dislipidemia, merokok, dan obesitas*.



Gambar 2.1 Penyumbatan pembuluh darah di otak penyebab stroke iskemik
Sumber : www.neuroendomke.com

2.2.1 Faktor Faktor Yang Mempengaruhi

Stroke diklasifikasikan menjadi dua yaitu *stroke* sumbatan (iskemik) dan *stroke* pendarahan (*hemorrargik*) (Dewi., 2012) yaitu:

1. *Stroke* sumbatan (iskemik), *stroke iskemik* terjadi apabila pembuluh darah ke otak tersumbat. Kondisi yang mendasari stroke iskemik adalah penumpukan lemak yang melapisi dinding pembuluh darah (disebut *asteroklerosis*). Kolesterol, *homocyteine* dan zat lainnya dapat melekat pada dinding *arteri*, membentuk zat lengket yang disebut *plak*. Seiring waktu, *plak* menumpuk. Hal ini sering membuat darah sulit mengalir dengan baik dan menyebabkan

bekuan darah (*trombus*). *Stroke Iskemik* dibedakan berdasarkan penyebab sumbatan arteri:

a) *Stroke trombotik*, Sumbatan disebabkan *trombus* yang berkembang didalam arteri otak yang sudah sangat sempit.

b) *Stroke embolik*, Sumbatan disebabkan *trombus*, gelembung udara atau pecahan lemak (*emboli*) yang terbentuk dibagian tubuh lain seperti jantung dan pembuluh aorta di dada dan leher, yang terbawa aliran darah darah ke otak. Kelainan jantung yang disebut *fibrilasi atrium* dapat menciptakan kondisi dimana *trombus* yang terbentuk di jantung terpompa dan beredar menuju otak.

2. Stroke pendarahan(hemorargik), disebakan oleh pembuluh darah yang bocor atau pecah di dalam atau di sekitar otak sehingga menghentikan suplai darah ke jaringan otak yang dituju. Selain itu, darah membanjiri dan memampatkan jaringan otak sehingga mengganggu atau mematikan fungsinya. Dua jenis stroke *hemoragik* :

   a) Perdarahan *intraserebral* adalah pendarahan di dalam otak yang disebabkan oleh *trauma* (cedera otak) atau kelainan pembuluh darah (*aneurisma* atau *angioma*). Jika tidak disebabkan oleh salah satu kondisi tersebut, paling sering disebakan oleh tekanan darah tinggi

   b) Perdarahan *subarachnoid* adalah perdarahan dalam ruang *subarachnoid*, ruang diantara lapisan dalam (*Pia mater*) dan lapisan tengah (*arachnoid mater*) dari jaringan selaput otak (*meninges*). Penyebab paling umum adalah pecahnya tonjolan (*aneurisma*) dalam arteri.

2.2.2 Faktor Faktor Yang Mempengaruhi

1. Umur

   Semakin bertambah umur seseorang maka semakin rentan terkena *stroke*. Menurut penelitian Aji Seto Arifianto dilihat dari usia pasien yang pernah terkena *stroke* didapatkan data pasien yang terkena stroke kurang dari 45 tahun sebanyak 26 pasien, usia 45-55 tahun sebanyak 105 pasien. Usia 56-65 tahun sebanyak 126 pasien.Sedangkan usia diatas 65 tahun ada 116 pasien (Seto., et. All., 2014).

2. Kolesterol Total

   Kolesterol total adalah kadar keseluruhan *kolesterol* yang beredar pada tubuh manusia. Zat ini merupakan zat lemak atau *lipid amfipatik* . Kadar *lipid amfipatik* yang terlalu tinggi dapat mempengaruhi kesehatan. Meski *kolesterol* tinggi tidak menyebabkan gejala apapun namun apabila jumlahnya terlampau tinggi dapat membahayakan kondisi kesehatan (Alodokter., 2015).

Tabel 2.2 Kolesterol Total

200 mg/dL atau kurang	Kadar yang diinginkan

200 – 239 mg/Dl	Batas kadar kolesterol tinggi
240 mg/dL atau lebih	Terlalu tinggi

Sumber: (A.M.A., 2015)

3. Kolesterol HDL (*High-Density Lipoprotein*)

HDL atau *lipoprotein* dengan kepadatan tinggi disebut kolesterol baik karena HDL mengangkut *kolesterol* dari sel-sel dan kembali ke hati. Di dalam hati *kolesterol* akan dihancurkan atau dikeluarkan tubuh melalui kotoran. Semakin tinggi kadar HDL seseorang maka potensi seseorang terkena penyakit *stroke* semakin rendah (Harsono., 2015).

Tabel 2.3 Kolesterol HDL (*High Density Lipoprotein*)

Kurang dari 35 mg/dL	Terlalu rendah
Lebih dari 35 mg/dL	Menguntungkan terutama diatas 60 mg/dL

Sumber : (A.M.A., 2015)

4. Kolesterol LDL (*Low Density-Lipoprotein*)

LDL atau disebut dengan *lipoprotein* dengan kepadatan yang rendah. LDL adalah kebalikan dari HDL. Jika Kadar LDL terlalu tinggi maka seseorang berpotensi terkena *stroke*, sedangkan jika kadar LDL nya rendah maka seseorang memiliki resiko yang rendah terkena *stroke*. LDL dikenal juga sebagai kolesterol jahat (Harsono., 2015)

Tabel 2.4 Kolesterol LDL (Low Density-Lipoprotein)

Kurang dari 100mg/Dl	Diinginkan
100-129 mg/Dl	Mendekati optimal/melebihi optimal
130-159 mg/Dl	Batas tinggi
160-189 mg/Dl	Tinggi
190 mg/Dl	Sangat tinggi

Sumber (A.M.A, 2015)

5. Trigliserida

Trigliserida adalah sejenis lemak yang bermanfaat sebagai sumber energi. Ketika kita makan – makanan yang berlebihan dari yang dibutuhan oleh tubuh, kelebihan kalori tersebut disimpan sebagai trigliserida dalam sel-sel lemak. Contohnya pada saat berpuasa kita tidak makan seharian karena masih memiliki kalori yang cukup pada tubuh dikarenakan apa yang kita makan ketika sahur. *Trigliserida* dalam kadar normal sangat diperlukan oleh tubuh.

2.3 Self Organizing Map (SOM)

Self Organizing Map (SOM) pertama kali diperkenalkan oleh Teuvo Kohonen dari University of Helsinki. Jaringan SOM Kohonen menggunakan metode pembelajaran unsupervised yaitu metode yang secara otomatis mengatur

keluarannya sesuai dengan aturan yang dimiliki. Jadi jika kondisi yang ditetapkan sudah terpenuhi maka proses akan berhenti (Kristanto., 2004).

Jaringan SOM merupakan jaringan yang mempunyai 2 lapisan (*layer*) , yaitu lapisan *input* dan lapisan *output.* Setiap neuron dalam lapisan masukan terhubung dengan lapisan keluaran dan dalam lapisan keluaran akan mempresentasikan kelas (cluster) dari masukan yang diberikan (Irwansyah., et. All., 2015).

Pada jaringan SOM tidak menggunakan fungsi aktivasi. Misalkan pada *iterasi* tertentu,*vektor* contoh w menjadi pemenang. Pada *iterasi* berikutnya, *vektor* w dan *vektor* – *vektor* sekitarnya akan dimodifikasi bobotnya(Kartika., 2015). Pada algoritma *Self Organizing Maps* bobot terkecil dijadikan pemenang dan perbaharuan bobot akan dilakukan pada bobot pemenang tersebut.

2.3.1 Algoritma Pembelajaran

Self Organizing Maps (SOM) merupakan cabang ilmu dari Jaringan Syaraf Tiruan (JST). Didalam jaringan syaraf tiruan terdapat proses pembelajaran yang digunakan untuk memperkecil tingkat *error* dalam mengenali suatu objek (Kartika., 2015). Ada 2 jenis sistem pembelajaran dalam Jaringan syaraf tiruan yaitu *supervised learning* dan *unsupervised learning*(Irwansyah., et. All., 2015).

1. Pembelajaran Terawasi (*Supervised Learning*).

   Pada proses pembelajaran dalam sistem ini, sistem diberikan pengetahuan terlebih dahulu yang digunakan untuk memetakan *input* menjadi output yang diinginkan. Proses pembelajaran ini berhenti apabila kondisi *error* dan kondisi yang diinginkan belum tercapai. Kondisi *error* diperlukan dalam setiap pemrosesan agar data maksimal yang diinginkan dapat tercapai.

2. Pembelajaran Tanpa Terawasi (*Unsupervised Learning*)

   Metode *unsupervised learning* adalah sebuah metode yang tidak memerlukan target keluaran. Selama proses pembelajaran dalam metode ini hasil yang diharapkan tidak dapat ditentukan. Tujuan dari metode ini adalah untuk mengelompokkan unit-unit yang sama ke dalam kelas tertentu. Algoritma yang termasuk dalam kelas ini adalah *Self Organizing Maps* (SOM)

2.3.2 Algoritma SOM

Dalam Jaringan SOM, tahap-tahap yang digunakan adalah sebagai berikut :

1. Menetapkan :
   a. *Input* data yang dinormalisasi
   b. Jumlah kelas
2. Inisialisasi :
   a. Bobot *input* (w_{ij}) dengan nilai *random*.
   b. Set parameter *learning rate* (α) awal.

c. Set pengurang *learning rate* (α)

d. Set parameter minimum *learning rate* (α)

3. Set Iterasi = 0
4. Kerjakan jika *learning rate*<*minimum learning rate*

a. Pilih data secara acak, misalnya data terpilih data ke-j

b. Cari jarak antara data ke-j dengan tiap bobot input ke-i (D_j) :

(D_j) = $\sum_i^m (w_{ij} - X_i)^2$.....(2.1)

c. Cari bobot yang terkecil atau (pemenang).

d. *Update* bobot yang terkecil dengan persamaan:

w_{ij}(baru) = w_{ij} (lama) + α [X_i-w_{ij} (lama)].....(2.2)

e. Iterasi = Iterasi+1

5. Perbarui learning rate (α)

a. Proses pembaharuan atau *update learning rate* (α) berhenti apabila nilai *learning rate* (α) kurang dari *minimal learning rate* (α) yang sudah ditentukan.

2.3.3 Normalisasi Data

Untuk melakukan proses pembelajaran pada klasifikasi resiko seseorang terkena penyakit *stroke* diperlukan proses *normalisasi* data. Proses *normalisasi* data ini perlu dilakukan agar data yang akan diproses memiliki *range* yang sama yaitu antara 0-1 (Purwandari., 2015).

Terdapat bermacam-macam metode untuk melakukan *normalisasi* data. Pada penelitian ini menggunakan metode *min-max normalisasi*. *Normalisasi* ini bertujuan untuk mendapatkan data yang lebih kecil tanpa mengubah karakteristik dari data tersebut. Untuk melakukan *normalisasi* data dapat menggunakan persamaan berikut :

$X_{norm} = \frac{X_i - X_{min}}{X_{max} - X_{min}}$.....(2.3)

Dengan :

X_{norm} = Data baru hasil normalisasi

X_i = Nilai Data yang lama

X_{max} = Nilai *maximum* dari X_i

X_{min} = Nilai *minimum* dari X_i

2.3.4 Learning Rate

Learning rate menentukan perubahan bobot dalam setiap iterasi karena setiap penambahan iterasi, nilai *learning rate* diubah sesuai dengan nilai

pengurangnya.Nilai dari *learning* berada pada *range* 0 sampai dengan 1 (Kartika., 2015). Semakin besar nilai *learning rate*, maka ketelitian dari jaringan akan semakin kecil karena bobot yang diubah hanya sedikit sehingga kemiripannya semakin jauh. Namun jika *learning rate* semakin kecil, ketelitian dari jaringan akan semakin besar karena bobot diubah semakin lama sehingga semakin mendekati nilai yang *konvergen*/sama. Penelitian untuk pemilihan learning rate terbaik pernah dilakukan, dengan range antara 0.00001 sampai dengan 100 didapatkan hasil berupa akurasi pelatihan data terbaik sebesar 69,46 % dengan 30.000 kali proses pelatihan, sedangkan dari rentang 100 sampai dengan 1 didapatkan akurasi pelatihan terbaik sebesar 56, 14 % (Wilson., et.all., 2011).

Rumus untuk menggunakan *update learning rate* :

$\alpha_t = \alpha_0 * \Delta\alpha$.....(2.4)

Dengan α_t adalah *learning rate* yang baru,dengan α_0 adalah *learning rate* yang lama, dan $\Delta\alpha$ adalah besar pengurang *learning rate.*

2.3.5 Evaluasi

Tahap evaluasi pada penelitian ini adalah bertujuan untuk mengetahui tingkat akurasi dari penggunaan algoritma SOM. Perhitungan pada tahap ini adalah dengan cara menghitung jumlah data uji yang kelasnya diprediksi secara benar dibandingkan dengan data uji yang kelasnya diprediksi salah.

Tahap evaluasi dilakukan dengan tujuan untuk mengukur keberhasilan sistem yang telah dibuat. Salah satu konsep evaluasi yang bisa digunakan adalah akurasi. Akurasi merupakan nilai dari perbandingan jumlah data yang sesuai dengan jumlah seluruh data yang ada. Nilai akurasi direpresentasikan dalam bentuk *presentase*. Untuk mengukur hasil evaluasi yang dihasilkan dari sistem dapat menggunakan persamaan berikut ini:(Kartika, 2015)

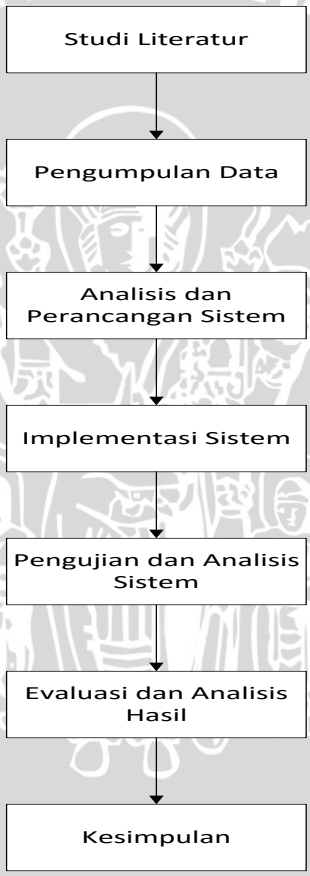
$$Akurasi = \frac{jumlah\ data\ yang\ benar}{jumlah\ seluruh\ data\ yang\ ada} \text{x } 100\% \quad \text{.....(2.5)}$$

BAB 3 METODOLOGI

Pada Bab Metodologi akan dibahas mengenai metode yang digunakan dan perancangan yang akan dibuat meliputi langkah-langkah penelitian hingga gambaran penggunaan algoritma pada implementasi metode *Self Organizing Maps* untuk mengiklasifikasi resiko seseorang terkena penyakit *stroke*.

3.1 Metode Penelitian

Bagian sub-bab ini menjelaskan tahapan atau langkah – langkah yang digunakan dalam penelitian ini berupa diagram alir. Diagram alir tahapan pada penelitian ini ditunjukkan Gambar 3.1



Gambar 3.1 Diagram Alir Penelitian

3.1.1 Studi Literatur

Studi literatur digunakan dalam mempelajari dasar-dasar teori dan literatur terkait dengan penelitian yang dilakukan. Pada penelitian kali ini data penyakit stroke diolah menggunakan algoritma *Self Organizing Maps* (SOM). Teori dan *literatur* tersebut diperoleh dari buku, jurnal, *ebook*, *internet*, penelitian sebelumnya, dan sumber pustaka lain yang dapat dipertanggung jawabkan.

3.1.2 Pengumpulan Data

Pengumpulan data pada penelitian ini menggunakan data *primer*. Data *primer* adalah data yang didapatkan secara langsung dari sumber penelitian. Data yang digunakan dalam penelitian ini berasal dari penelitian sebelumnya yaitu "Implementasi Algoritma SVM (*Support Vector Machine*)untuk mengetahui tingkat penyakit stroke". Penelitian tersebut mengambil data penyakit *stroke* dari hasil tes darah yang diambil dari laboratorium Klinik Sejahtera Probolinggo. Jumlah data yang digunakan dalam penelitian ini adalah 200 data, dimana data tersebut dikelompokkan menjadi 3(tiga) kelas yang berbeda yaitu kelas normal,kelas rentan dan kelas stroke. Fitur yang digunakan ada 5(lima) yaitu umur,*Kolesterol* total, *Kolesterol* HDL (*High Density-Lipoprotein*), *Kolesterol* LDL (*Low Density-Lipoprotein*) dan *Trigliserida*.

3.1.3 *Analisis* dan Perancangan Sistem

Pada analisis dan perancangan sistem membahas mengenai semua hal terkait yang diperlukan dalam proses pembuatan sistem. Sistem yang akan dibangun adalah sebuah sistem perangkat lunak untuk *mengklasifikasi* resiko seseorang terkena penyakit *stroke* menggunakan *algoritma Self Organizing Maps* (SOM). Secara keseluruhan, kebutuhan yang digunakan dalam pembuatan sistem ini meliputi:

1. Kebutuhan perangkat keras :
   - Laptop dengan processor Intel (R) Core (TM) 2 Duo CPU T6400 @ 2.00GHz (2CPUs),~2.0GHz
   - Memori (RAM) 3GB
2. Kebutuhan perangkat lunak :
   - Windows 7 Professional 32-bit
   - Microsoft Office 2013
   - Sublime Text 3 for PHP developers
3. Data yang dibutuhkan, meliputi:
   - Data pasien penyakit stroke Laboratorium Klinik Sejahtera Probolinggo

Perancangan sistem adalah tahap dimana penulis memulai tahap merancang suatu sistem serta menerapkan teori-teori dari pustaka dan data sampel kemudian digabungkan dengan ilmu yang didapat untuk mengimplementasikan serta mengembangkan suatu sistem klasifikasi resiko penyakit stroke dengan menggunakan algoritma *self organizing map*.

3.1.4 Implementasi Sistem

Implementasi Sistem adalah proses pembuatan program aplikasi sesuai dengan analisis dan perancangan sistem yang dilakukan pada tahap sebelumnya. Dimana proses klasifikasi resiko seseorang terkena penyakit *stroke*

diimplementasikan ke dalam program aplikasi berbasis *web* dengan menggunakan metode SOM yang selanjutnya dioperasikan untuk diuji dan dianalisis pada data yang ada.Implementasi perangkat lunak ini menggunakan bahasa pemrograman PHP, Sublime Text2 for PHP Developers. Implementasi sistem tersebut meliputi :

- Pembuatan antar muka
- Memasukkan data berupa nilai batasan seseorang tekena penyakit stroke ke dalam Komputer untuk diolah menjadi informasi yang berguna bagi sistem.
- Menerapkan algoritma *Self Organizing Maps* dalam aplikasi yang telah dibuat menggunakan bahasa pemrograman PHP.
- *Output* berupa klasifikasi seseorang dikategorikan kedalam kelas normal,kelas rentan,kelas *stroke*.

3.1.5 Pengujian dan Analisis Sistem

Pada tahap ini pengujian dan analisis, permasalahan klasifikasi dan tingkat akurasi yang telah di implementasikan dalam bentuk program aplikasi akan di uji tingkat keberhasilannya. Terdapat beberapa skenario pengujian yang dilakukan meliputi :

1. Pengujian untuk mengetahui pengaruh nilai laju pembelajaran awal terhadap tingkat akurasi.
2. Pengujian untuk mengetahui pengaruh tingkat akurasi terhadap parameter pengurang laju pembelajaran.
3. Pengujian untuk mengetahui pengaruh laju pembelajaran minimum terhadap tingkat akurasi.
4. Pengujian untuk mengetahui pengaruh tingkat akurasi terhadap jumlah *iterasi* maksimum.
5. Pengujian untuk mengetahui pengaruh perbandingan data latih dan data uji terhadap tingkat akurasi.

3.1.6 Evaluasi dan Analisis Hasil

Tahap evaluasi dilakukan setelah proses pengujian sistem , evaluasi dilakukan untuk mempertimbangkan tingkat keberhasilan program aplikasi dan implementasi metode dalam menyelesaikan permasalahan objek yang diteliti. Fungsi dari Evaluasi dan analisis hasil ini adalah apakah sistem yang dibuat dengan metode yang digunakan kinerjanya lebih optimal sesuai dengan tujuan penelitian ini. Kemudian tahap akhir dari seluruh rangkaian proses ini adalah penarikan kesimpulan terhadap hasil yang diperoleh dari proses pengujian dan analisis yang telah dievaluasi. Selanjutnya kesimpulan dan saran dapat digunakan pembaca sebagai acuan perbaikan dan pengembangan sistem dan aplikasi selanjutnya.

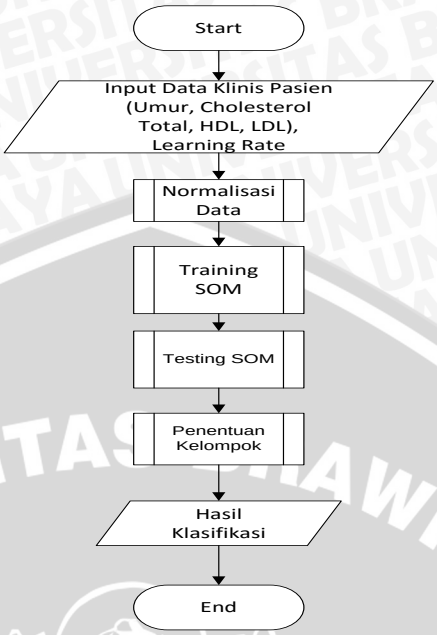
3.2 Perancangan

Dalam bab perancangan akan dibahas mengenai deskripsi umum sistem, perancangan proses, perhitungan manual, perancangan interface dan perancangan uji coba dan evaluasi.

3.2.1 Deskripsi Umum Sistem

Sistem yang dibuat dalam skripsi kali ini merupakan sistem untuk melakukan klasifikasi resiko seseorang terkena penyakit *stroke*. Data yang digunakan dari penelitian ini didapatkan dari penelitian sebelumnya yaitu tentang implementasi algoritma SVM (*Support Vector Machine*) untuk mengetahui tingkat resiko penyakit *stroke*. Pada penelitian tersebut data didapatkan dari Laboratorium Klinik Sejahtera Kota Probolinggo. Data tersebut terdiri dari Umur, Kolesterol Total, Kolesterol HDL (*High-DensityLipoprotein*), Kolesterol LDL (*LowDensityLipoprotein*), *Trigliserida*. Dari ke lima data tersebut Penulis menjadikannya sebagai fitur dalam sistem yang dibuat dalam penelitian kali ini. Lima data tersebut juga merupakan data *input* dalam sistem ini. Data tersebut berupa angka atau nilai dari masing masing fitur yang dibuat. Data Inputan tersebut kemudian diproses kedalam sistem yang menggunakan algoritma SOM. Hasil dari proses tersebut nanti berupa pengelompokan atau pengklasifikasian seseorang ke dalam tiga kelas yaitu tidak terkena stroke atau normal, rentan dan stroke. Selain hasil berupa 3 kelas tersebut terdapat juga tingkat akurasi sistem yang berguna untuk mengetahui tingkat keakuratan data *training* dengan sistem yang dibuat.

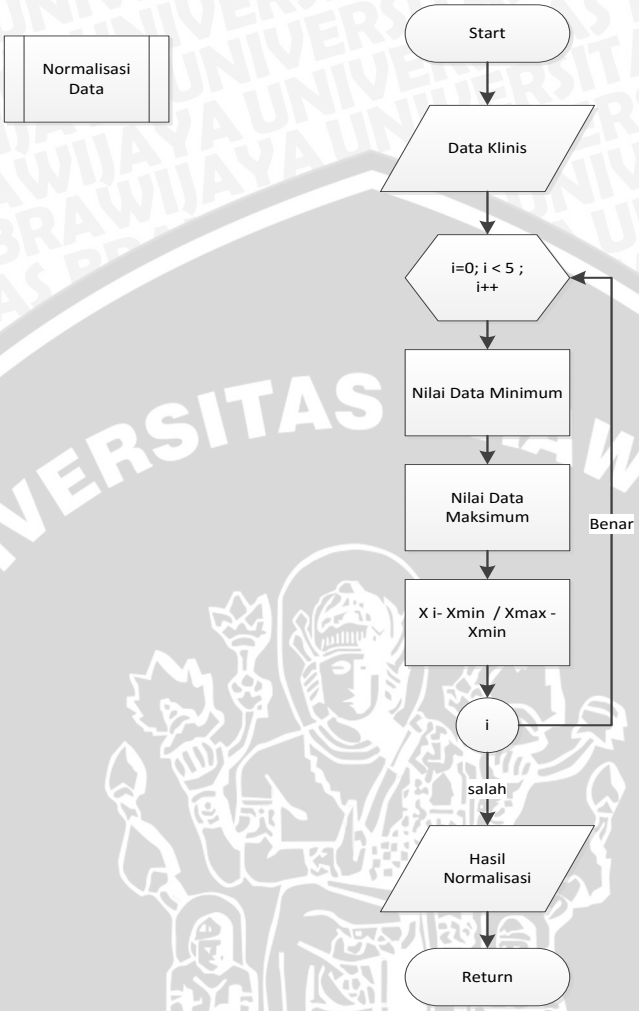
3.2.2 Perancangan Proses



Gambar 3.2 Diagram Alir SOM

Pada gambar 3.2 dijelaskan bahwa data masukan dari sistem berupa data klinis pasien yang terdiri dari umur, *cholesterol,* HDL dan LDL dan juga *learning rate*. Data tersebut digunakan sebagai masukan untuk proses *training* dan *testing* menggunakan SOM. Dari proses tersebut pada akhirnya diperoleh hasil keluaran berupa hasil *identifikasi* seseorang tidak terkena *stroke* atau normal, rentan , stroke dan tingkat akurasi data.

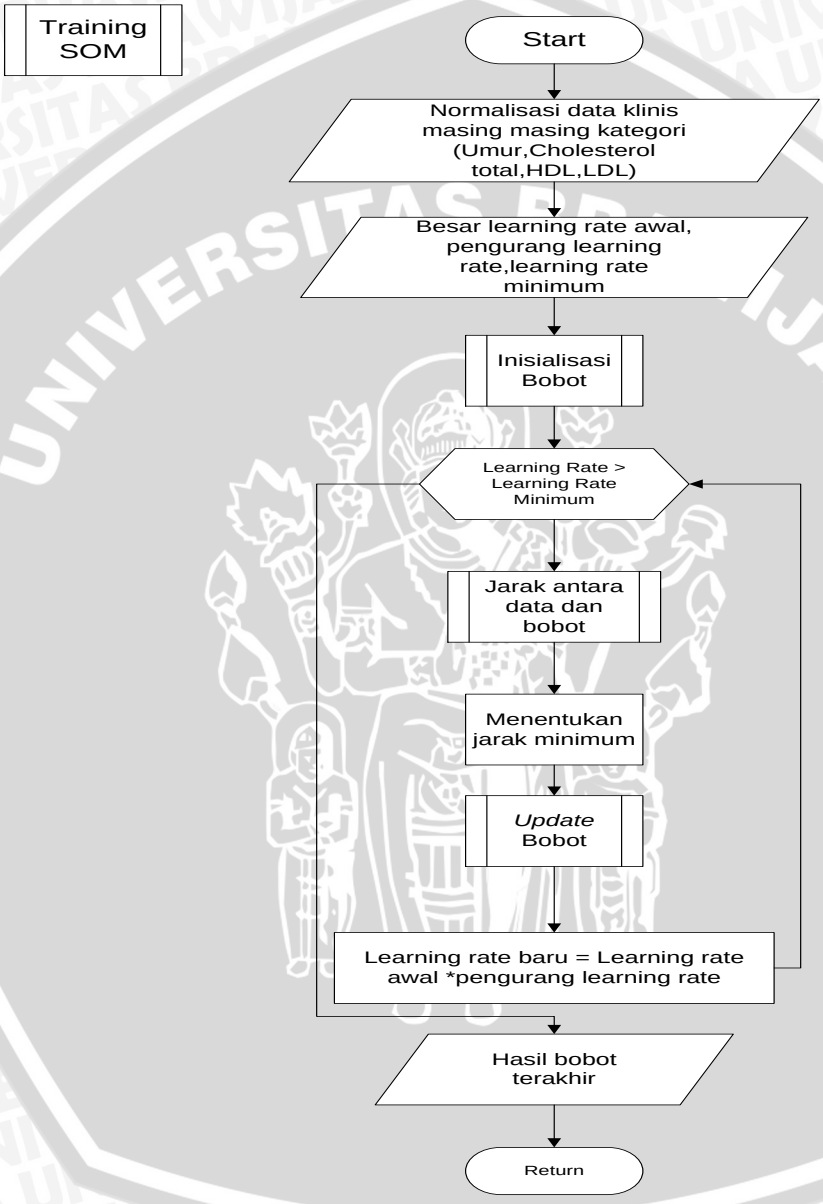
3.2.2.1 Normalisasi Data



Gambar 3.3 Diagram Alir Normalisasi Data

Pada proses Gambar 3.3, untuk melakukan *normalisasi* pada data yang akan dilakukan *training* dan *testing*, langkah awal adalah mengambil nilai dari data klinis, yaitu gabungan antara data *training* dan data *testing*. Untuk masing masing kelas dari data klinis yaitu Umur, *Cholesterol*, HDL, LDL dan *Trigliserida* ditentukan nilai maksimum dan minimum. Setelah didapatkan nilai maksimum dan minimum, setiap nilai data pada masing-masing kelas dilakukan perhitungan. Perhitungan yang dilakukan adalah menggunakan *min max normalisasi*. Dimana perhitungannya adalah data asli (x) dikurangi dengan data minimum (x min) kemudian dibagi dengan data maksimum (x max) dikurangi dengan data minimum (x min). Sehingga dari proses tersebut dihasilkan dataset baru dengan nilai yang sudah di *normalisasi.*

3.2.2.2 Proses Training SOM

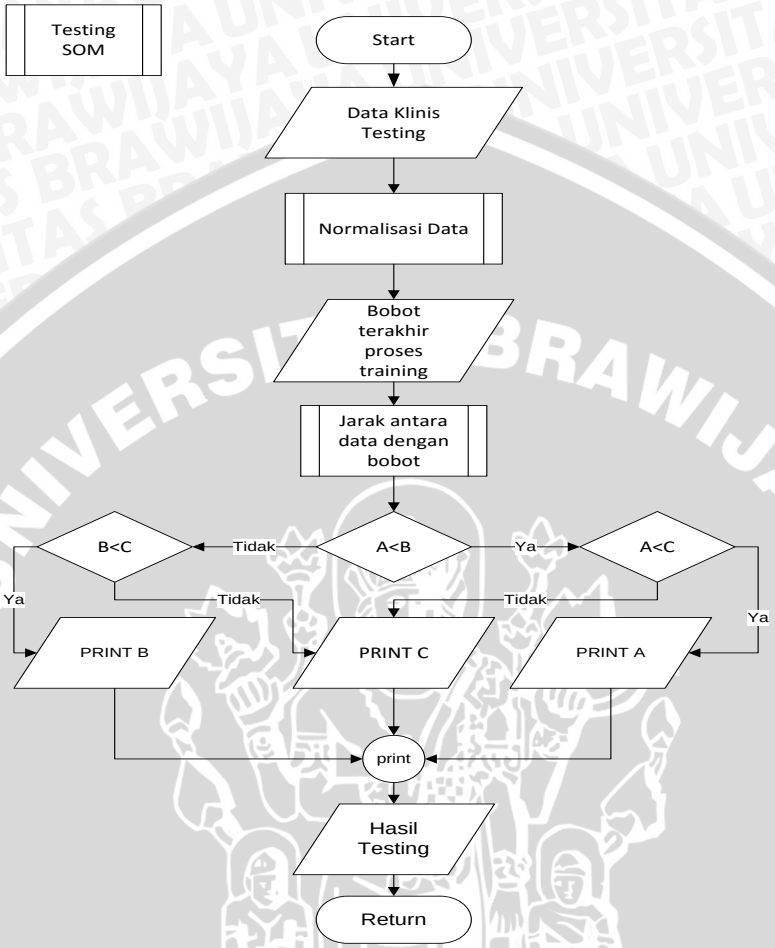


Gambar 3.4 Diagram Alir Proses Training SOM

Pada Gambar 3.3 dijelaskan mengenai proses *training* SOM. Terdapat 3 proses yang dilakukan oleh sistem yaitu proses *inisialisasi* bobot, proses menghitung jarak antara bobot dan proses *update* bobot. Data masukan dari proses ini berupa nilai-nilai dari data klinis (umur, cholesterol total,HDL, LDL)yang sudah melalui proses *normalisasi.* Kemudian data tersebut diproses yang

menghasilkan keluaran berupa hasil bobot terakhir yang akan digunakan untuk proses selanjutnya.

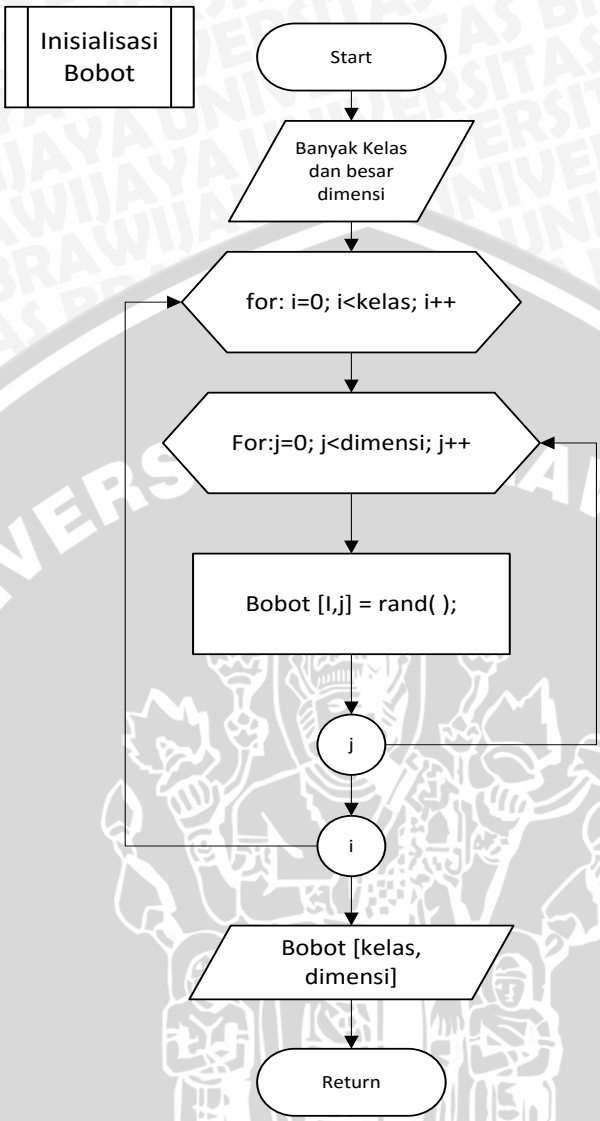
3.2.2.3 Proses*Testing* SOM



Gambar 3.5 Diagram Alir Proses Testing SOM

Pada Gambar 3.5 dijelaskan mengenai proses *testing* SOM. Terdapat 2 proses yang dilakukan oleh sistem yaitu proses *normalisasi* data dan proses perhitungan jarak antara data dengan bobot. Pada proses ini jarak yang dipilih adalah jarak dengan nilai paling kecil tanpa melibatkan proses *update* bobot. Pada gambar 3.5 proses jarak antara data dengan bobot penjelasannya adalah apabila A (normal) jaraknya kurang dari B(rentan) maka berlanjut ke proses perhitungan jarak A(normal) kurang dari C(stroke) jika bernilai benar maka hasil testing dicetak A (normal).

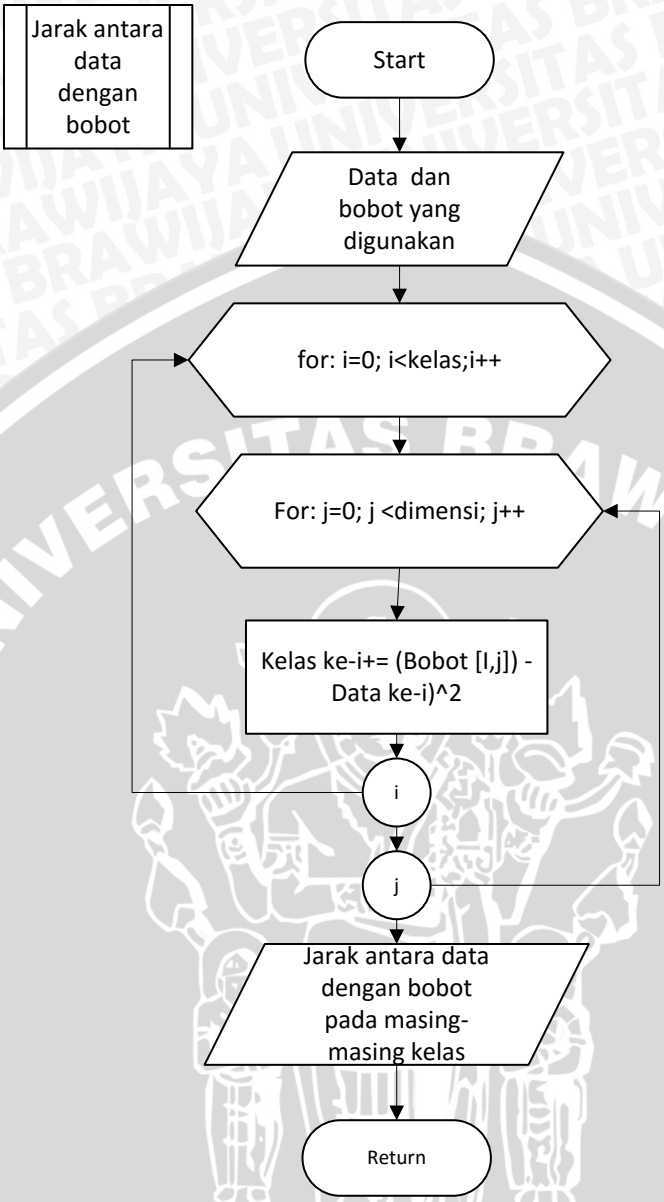
3.2.2.4 Proses Inisialisasi Bobot



Gambar 3.6 Diagram Alir Proses Inisialisasi Bobot

Pada Gambar 3.7 , untuk melakukan inisialisasi bobot tahap training sebagai data masukan adalah berupa banyak kelas dan banyak dimensi yang digunakan. Selanjutnya penentuan nilai bobot yaitu secara random dengan panjang sesuai dengan jumlah kelas dan jumlah dimensi yang ditentukan. Selama proses masih bernilai benar, maka bobot akan terus dihasilkan dan melakukan perulangan sampai dengan jumlah *iterasi* yang ditentukan dan dihasilkan keluaran berupa nilai bobot yang dapat digunakan untuk proses selanjutnya.

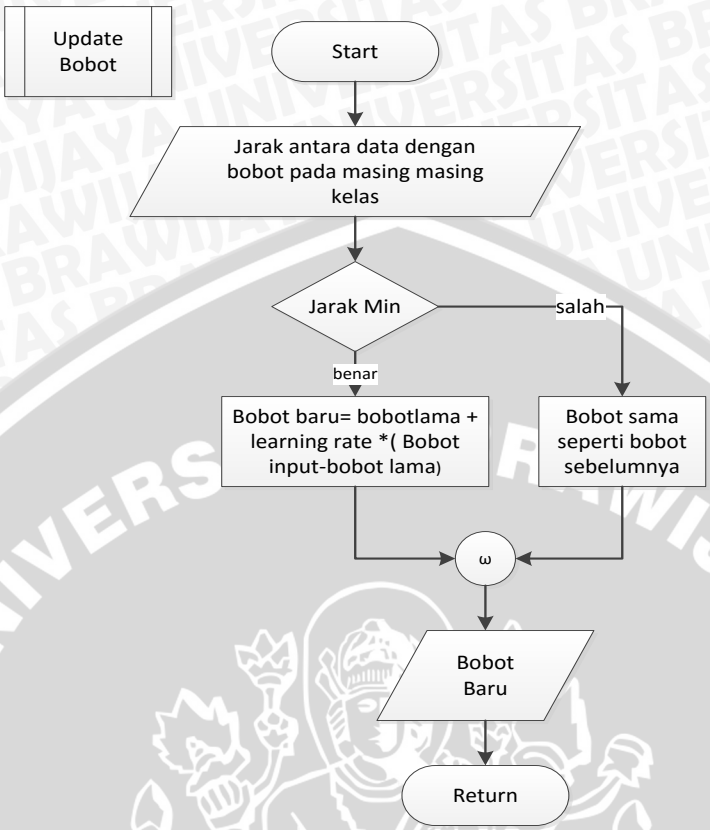
3.2.2.5 Proses Jarak Antara Data Dengan Bobot



Gambar 3.7 Diagram Alir Jarak Antara Data Dengan Bobot

Pada Gambar 3.8, untuk mencari jarak antara data dengan bobot tahap awal yang digunakan untuk masukan adalah berupa data *training* (*Xi*) yang sudah dinormalisasi dan bobot yang digunakan. Ketika proses perulangan benar, maka nilai kelas dikurangkan dengan bobot lalu dipangkatkan dua. Untuk jarak pada kelas yang sama dilakukan penjumlahan dari proses pengurangan sebelumnya. Proses tersebut dilakukan juga pada kelas lainnya. Dari proses ini, akan dihasilkan keluaran berupa jarak antara data dengan bobot pada masing-masing kelas yang akan digunakan untuk proses selanjutnya.

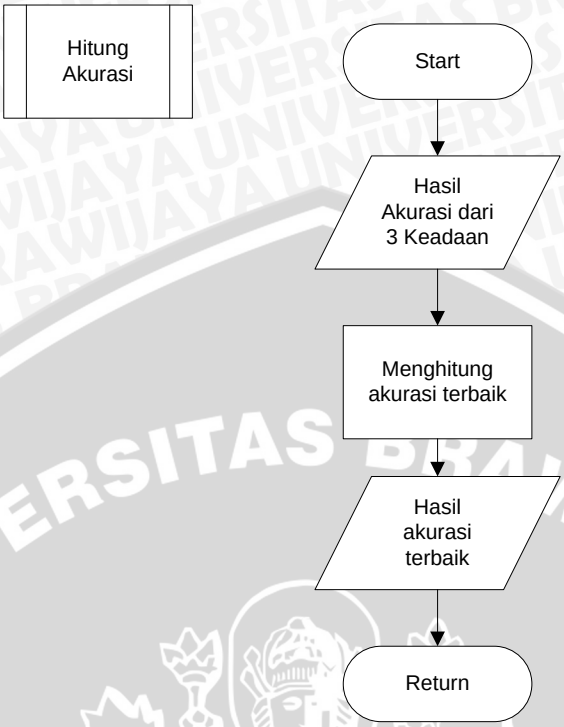
3.2.2.6 Proses Update Bobot



Gambar 3.8 Proses Update Bobot

Pada Gambar 3.9 untuk melakukan *update* bobot yang digunakan untuk masukan adalah berupa jarak antara data dengan bobot pada masing masing kelas. Selanjutnya adalah mencari jarak minimum pada masing-masing kelas. Jika jarak pada kelas tersebut merupakan jarak minimum pada masing masing kelas , maka dilakukan proses *update* bobot pada kelas tersebut yaitu dengan menggunakan persamaan (2.2), namun jika salah maka bobot akan sama seperti bobot sebelumnya. Hasil keluaran dari proses ini adalah berupa bobot baru yang akan digunakan pada proses selanjutnya.

3.2.2.7 Proses Menghitung Akurasi Terbaik



Gambar 3.9 Proses Menghitung Akurasi Terbaik

Pada Gambar 3.13 dijelaskan mengenai proses menghitung akurasi terbaik. Masukan dari proses ini yaitu berupa hasil akurasi yang memperhatikan 3 keadaan dimana kelas bisa saja tertukar. Kelas tersebut terdiri dari kelas normal,rentan dan stroke. Dari hasil akurasi tersebut dicari mana hasil akurasi yang terbaik dengan cara mencari hasil akurasi yang paling maksimum.

3.2.3 Perhitungan Manual

Perhitungan manual berfungsi untuk mengetahui perkiraan hasil manual sebelum sistem dibuat, sehingga dapat diketahui hasilnya sesuai atau tidak antara hasil hitung manual dengan sistem yang telah dibuat. Langkah pertama yang dilakukan adalah menentukan data latih/data *training* dan data uji/data *testing*.

Tabel 3.1 Data Latih/Data *Training*

ID	Umur	*Cholesterol*	HDL	LDL	*Trigliserida*
7	49	171	40,1	104,1	134
9	49	184	38,4	119,2	133
121	42	242	42,5	162,5	185
122	25	210	40,1	134,1	179
161	79	340	31,2	208,9	162
162	38	287	51,6	172,2	31,6

Tabel 3.2 Data Testing

ID	Umur	*Cholesterol*	HDL	LDL	*Trigliserida*
5	40	253	46,5	179,3	136
130	57	168	40,8	88,8	192
170	41	173	37,3	113,5	281

Setelah data *training* dan data *testing* ditentukan kemudian data tersebut digabung untuk proses *normalisasi* data.

3.2.3.1 Normalisasi Data

Proses Selanjutnya adalah menentukan nilai minimum dan maksimum pada masing masing *fitur*. Pada *fitur* umur tabel 3.3 misalnya didapatkan nilai minimum 25 dan maksimum 79.

Tabel 3.3 Mencari Nilai Minimum dan Maksimum

ID	Umur	*Cholesterol*	HDL	LDL	*Trigliserida*	Kelompok
7	49	171	40,1	104,1	134	1
9	49	184	38,4	119,2	133	1
121	42	242	42,5	162,5	185	2
122	25	210	40,1	134,1	179	2
161	79	340	31,2	208,9	162	3
162	38	287	51,6	172,2	316	3
5	40	253	46,5	179,3	136	1
130	57	168	40,8	88,8	192	2
170	41	173	37,3	113,5	281	3
Min	25	168	31,2	88,8	133	1
Max	79	340	51,6	208,9	316	3

Proses selanjutnya adalah melakukan *normalisasi* pada data training dan data testing menggunakan persamaan (2.3). Hasil *normalisasi* data ditunjukkan pada tabel 3.4. Contoh perhitungan pada fitur umur, didapatkan nilai minimum yaitu 25 dan nilai maksimum 79. Maka Untuk fitur umur pada ID 7 didapatkan hasil *normalisasi* sebagai berikut:

$$X_{Norm} = \frac{X_i - X_{min}}{X_{max} - X_{min}} = \frac{49 - 25}{79 - 25} = 0{,}44444444$$

Tabel 3.4 Data Hasil Normalisasi

ID	Umur	*Cholesterol*	HDL	LDL	*Trigliserida*
7	0,44444444	0,01744186	0,43627	0,12739	0,0067567
9	0,44444444	0,093023256	0,35294	0,25312	0
121	0,31481481	0,430232558	0,55392	0,61365	0,2841530
122	0	0,244186047	0,43627	0,37718	0,2513661
161	1	1	0	1	0,1584699

ID	Umur	*Cholesterol*	HDL	LDL	*Trigliserida*
162	0,24074074	0,691860465	1	0,69442	1
5	0,27777777	0,494186047	0,75	0,75353	0,0163934
130	0,59259259	0	0,4705	0	0,3224043
170	0,296296296	0,029069767	0,29902	0,205662	0,8087431

Langkah selanjutnya adalah menentukan parameter awal yang digunakan dalam pelatihan, yaitu:
Learning rate awal = 0,6
Pengurang Learning rate =0,5
Learning rate minimum = 0,1

3.2.3.2 *Inisialisasi* Bobot

Proses selanjutnya adalah *inisialisasi* bobot. Tujuan dari *inisialisasi* adalah menentukan nilai awal bobot,parameter *learning rate* dan *iterasi maksimum*.

- Langkah pertama yang dilakukan adalah menentukan bobot *input*(w_{ij}) dengan nilai *random*.

Tabel 3.5 Bobot Input

(wij) Random	*Umur*	*Cholesterol*	*HDL*	LDL	*Trigliserida*
w_1	0,16	0,35	0,63	0,77	0,55
w_2	0,34	0,54	0,23	0,28	0,38
w_3	0,45	0,76	0,98	0,69	0,79

- Langkah kedua adalah menghitung jarak antara data dengan bobot dari masing masing kelas dapat menggunakan persamaan (2.1). Hasil dari perhitungan jarak tersebut dilakukan sampai iterasi ke-3 yang ditunjukkan pada tabel 3.6. Berikut adalah contoh dari perhitungan jarak pada data ke-1:
  X_1= 0,444444444, X_2=0,01744186, X_3=0,436275, X_4 =0,12739, X_5 = 0,0067567 .

D (1) = $\sum_{i=0}^{m}(w_{ij-}Xi)^2$ = (0,16 - 0,44444444)^2 + (0,35 - 0,01744186)^2 + (0,63 - 0,436275)^2 + (0,77 - 0,12739)^2 + (0,55 - 0,0067567)^2= 0,93708924

D (2) = $\sum_{i=0}^{m}(w_{ij-}Xi)^2$ = (0,34 - 0,44444444)^2 + (0,54 - 0,01744186)^2 + (0,23 - 0,436275)^2 + (0,28 - 0,12739)^2 + (0,38 - 0,0067567)^2= 0,489123984

D (3) = $\sum_{i=0}^{m}(w_{ij-}Xi)^2$ = (0,45 - 0,44444444)^2 + (0,76 - 0,01744186)^2 + (0,98 - 0,436275)^2 + (0,69 - 0,12739)^2 + (0,79 - 0,0067567)^2= 1,777057

Tabel 3.6 Hasil Jarak Antara Data Dengan Bobot

Iterasi ke-i	*Learning Rate*	Data ke- i	Jarak pada kelas ke-j		
			1	2	3
0	0.6	1	0.937089024	0.489123984	1.777057
		2	0.793369736	0.048089938	1.653054
		3	0.131311112	0.33204768	0.570274
		4	0.218343126	0.257588877	1.152180699
		5	1.810693943	1.787293226	1.81543021
		6	0.9423511	1.794057583	0.092951982
1	0.3	1	0.431095653	0.935110831	1.962
		2	0.153968115	0.677380466	1.847031619
		3	0.165398173	0.463469447	0.680930476
		4	0.090424562	0.968955712	1.180911436
		5	1.759549667	0.285966916	2.187649065
		6	1.187890488	1.937679984	0.014872317
2	0.15	1	0.23688365	1.26918143	2.006802265
		2	0.11481588	0.96573982	1.893858264
		3	0.15413114	0.64417594	0.711092797
		4	0.08320611	1.26565121	1.194688634
		5	1.75362799	0.14012379	2.270123887
		6	1.26390897	2.00010857	0.007287435

3.2.3.3 Update Bobot

Update bobot w_{ij}(baru) dilakukan pada data yang memiliki jarak minimum berdasarkan hasil perhitungan sebelumnya. Dalam melakukan *update* bobot dapat menggunakan persamaan (2.2). Hasil dari perhitungan *update* bobot sampai iterasi ke 3 dapat dilihat pada tabel 3.6. Sebagai contoh jarak pada data ke-1 dihasilkan jarak minimum pada kelas ke 2, maka bobot untuk kelas 2 *diupdate*, sedangkan bobot untuk kelas 1 dan 3 sama seperti sebelumnya. Contoh perhitungan *update* bobot pada data ke-1:

w_{ij}(baru) = w_{ij} (lama) + α [X_i-w_{ij} (lama)] = 0,34 + 0,6 *(0,44444444 - 0,34) =0.402666664

w_{ij}(baru) = w_{ij}(lama) + α [X_i-w_{ij} (lama)] = 0,54 + 0,6 * (0.01744186 - 0,54) = 0.226465116

w_{ij}(baru) = w_{ij}(lama) + α [X_i-w_{ij} (lama)] = 0,23 + 0,6 * (0.436275

- 0,23) = 0.353765

w_{ij}(baru) = w_{ij}(lama) + α [X_i-w_{ij} (lama)] = 0,28 + 0,6 * (0.127394
- 0,28) = 0.18844

w_{ij}(baru) = w_{ij}(lama) + α [X_i-w_{ij} (lama)] = 0,38 + 0,6 * (0.00675676

- 0,38) = 0.15605 Hasil Update Bobot (w_{ij})

Tabel 3.7 Hasil Update Bobot ($\boldsymbol{w_{ij}}$)

Iterasi ke-i	*Learning rate*	Data ke-i	Kelas ke-j		
			1	2	3
0	0.6	1	0.16	0.40267	0.45
			0.35	0.22647	0.76
			0.63	0.35377	0.98
			0.77	0.18844	0.69
			0.55	0.15605	0.79
		2	0.16	0.42773	0.45
			0.35	0.1464	0.76
			0.63	0.35327	0.98
			0.77	0.22725	0.69
			0.55	0.06242	0.79
		3	0.25289	0.42773	0.45
			0.39814	0.1464	0.76
			0.58435	0.35327	0.98
			0.67619	0.22725	0.69

			0.39049	0.06242	0.79
		4	0.10116	0.42773	0.45
			0.30577	0.1464	0.76
			0.49551	0.35327	0.98
			0.49679	0.22725	0.69
			0.30702	0.06242	0.79
		5	0.10116	0.77109	0.45
			0.30577	0.65856	0.76
			0.49551	0.14131	0.98
			0.49679	0.6909	0.69
			0.30702	0.12005	0.79
		6	0.10116	0.77109	0.3244444
			0.30577	0.65856	0.7191163
			0.49551	0.14131	0.992
			0.49679	0.6909	0.6926528
			0.30702	0.12005	0.916
1	0.3	1	0.20414	0.77109	0.3244444
			0.21927	0.65856	0.7191163
			0.47774	0.14131	0.992
			0.38597	0.6909	0.6926528
			0.21694	0.12005	0.916
		2	0.27623	0.77109	0.3244444
			0.1814	0.65856	0.7191163
			0.4403	0.14131	0.992
			0.34612	0.6909	0.6926528
			0.15186	0.12005	0.916
		3	0.28781	0.77109	0.3244444
			0.25605	0.65856	0.7191163
			0.47439	0.14131	0.992
			0.42638	0.6909	0.6926528
			0.19155	0.12005	0.916
		4	0.20147	0.77109	0.3244444
			0.25249	0.65856	0.7191163
			0.46295	0.14131	0.992
			0.41162	0.6909	0.6926528
			0.20949	0.12005	0.916
		5	0.20147	0.83977	0.3244444
			0.25249	0.76099	0.7191163
			0.46295	0.09892	0.992
			0.41162	0.78363	0.6926528
			0.20949	0.13158	0.916

		6	0.20147	0.83977	0.2993333
			0.25249	0.76099	0.7109395
			0.46295	0.09892	0.9944
			0.41162	0.78363	0.6931833
			0.20949	0.13158	0.9412
2	0.15	1	0.23791	0.83977	0.2993333
			0.21723	0.76099	0.7109395
			0.45895	0.09892	0.9944
			0.36899	0.78363	0.6931833
			0.17908	0.13158	0.9412
		2	0.26889	0.83977	0.2993333
			0.1986	0.76099	0.7109395
			0.44305	0.09892	0.9944
			0.35161	0.78363	0.6931833
			0.15222	0.13158	0.9412
		3	0.27578	0.83977	0.2993333
			0.23335	0.76099	0.7109395
			0.45968	0.09892	0.9944
			0.39091	0.78363	0.6931833
			0.17201	0.13158	0.9412
		4	0.23441	0.83977	0.2993333
			0.23497	0.76099	0.7109395
			0.45617	0.09892	0.9944
			0.38885	0.78363	0.6931833
			0.18391	0.13158	0.9412
		5	0.23441	0.8638	0.2993333
			0.23497	0.79684	0.7109395
			0.45617	0.08408	0.9944
			0.38885	0.81609	0.6931833
			0.18391	0.13561	0.9412
		6	0.23441	0.8638	0.2905444
			0.23497	0.79684	0.7080777
			0.45617	0.08408	0.99524
			0.38885	0.81609	0.693369
			0.18391	0.13561	0.95002

3.2.3.4 Update Learning rate

Untuk mendapatkan hasil dari *learning rate* yang baru dapat dihitung dengan persamaan (2.4). Contoh perhitungannya adalah sebagai berikut : $\alpha_t = \alpha_0 * \Delta\alpha = 0.6 * 0.5 = 0.3$

Karena nilai *learning rate* minimum adalah 0.1, maka perulangan akan terus dilakukan sampai nilai *learning rate* kurang dari minimal *learning* yang sudah ditentukan. Pada iterasi ketiga tidak perlu dilakukan update learning rate lagi. Hal ini dikarenakan pada iterasi ke 3, nilai learning rate kurang dari nilai learning rate minimum yang sudah ditentukan yaitu 0.1 dengan nilai learning rate minimum pada iterasi ketiga sebesar 0.075. Setelah hasil dari minimal terpenuhi, bobot terakhir dari hasil pelatihan disimpan untuk digunakan proses pengujian.

3.2.3.5 Pengujian Data/Data *Testing*

Data yang akan digunakan untuk data uji adalah data pada tabel 3.2. Data pengujian juga melewati proses *normalisasi* menggunakan persamaan (2.3). Hasil dari normalisasi data ditunjukkan oleh tabel 3.3. Bobot yang digunakan untuk proses pengujian adalah bobot terakhir dari proses pelatihan yang ditunjukkan pada tabel 3.7 . Tabel 3.8 adalah inisialisasi bobot yang digunakan untuk proses pengujian.

Tabel 3.8 Bobot Data Pengujian **(w_{ij})**

Inisialisasi Bobot		
Kelas 1	Kelas 2	Kelas 3
0.234413329	0.863800533	0.290544444
0.234971303	0.7968432	0.708077674
0.456169	0.084078	0.99524
0.388855	0.816085	0.693369
0.183912918	0.135610444	0.95002

Selanjutnya menghitung jarak dari masing masing data menggunakan persamaan (2.1). Data (Xi) yang digunakan merupakan data *testing* yang sudah dinormalisasi. Jika hasil jarak pada kelas 1 lebih kecil data masuk kedalam kelas 1, jika hasil jarak pada kelas 2 lebih kecil maka data masuk kedalam kelas 2 dan jika hasil jarak pada kelas 3 lebih kecil maka data masuk ke dalam kelas 3. Hasil perhitungan secara keseluruhan ditunjukkan pada Tabel 3.9 . Berikut adalah contoh perhitungan untuk data-1:

X_1= 0.234413329, X_2= 0.234971303, X_3= 0.456169022, X_4 = 0.388854672, X_5 = 0.183912918.

D (1) = $\sum_{i=0}^{m}(w_{ij} - Xi)$ = (0.234413329 - 0.278)^2+ (0.234971303 - 0.494186047) ^{2} + (0.456169022 - 0.75) ^{2} + (0.388854672 - 0.753538718) ^{2}+(0.183912918 - 0.016393443)^2 = 0.31646663

D (2) = $\sum_{i=0}^{m}(w_{ij-} Xi)$ = (0.863800533 - 0.278)^2+ (0.7968432 - 0.494186047)^2 + (0.084078 - 0.75)^2 + (0.816085 - 0.753538718)^2+(0.1135610444 - 0.016393443)^2 = 0.896600296

D (3) = $\sum_{i=0}^{m}(w_{ij-} Xi)$ = (0.290544444 - 0.278)^2+ (0.708077674 - 0.494186047)^2 + (0.99524 - 0.75)^2 + (0.693369 - 0.753538718)^2+(0.95002 - 0.016393443)^2 = 0.981334212

Dari hasil jarak tersebut, jarak pada kelas 1 lebih kecil, maka data-1 masuk ke dalam kelas 1 yaitu normal.

Tabel 3.9 Hasil Jarak Dan Klasifikasi Stroke

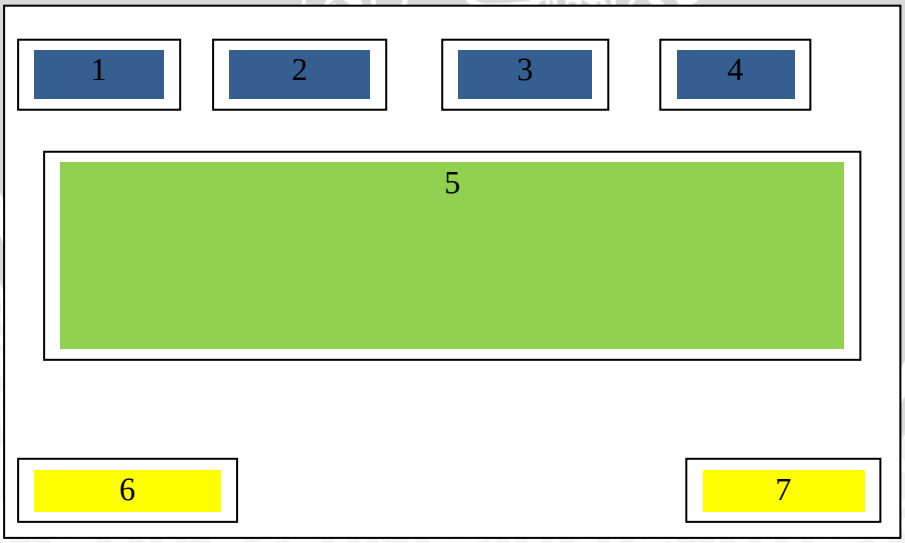
Data ke i	Jarak pada kelas ke-j			Jarak Minimum	Hasil Klasifikasi	Kelas Sebenarnya
	1	2	3			
1	0.31646663	0.896600296	0.981334212	Kelas 1	Normal	Normal
2	0.35409965	1.558789401	0.35409965	Kelas 1	Normal	Stroke
3	0.4948933	1.78346081	1.203625008	Kelas 1	Normal	Rentan

Dari hasil Klasifikasi diatas, maka dapat dihitung tingkat akurasi yang dihasilkan menggunakan persamaan (2-5). Contoh perhitungan akurasi diatas

akurasi = $\frac{1}{3} x\ 100\%$ = 30%

3.2.4 Perancangan *Interface*

Perancangan Interface untuk klasifikasi resiko penyakit stroke terdiri dari 1 form dengan 4 tab yang terdiri dari profil, load data, manualisasi dan pengujian.



Gambar 3.10 Perancangan Interface

Penjelasan Gambar 3.10:

1. Tab Profil

2. Tab Load data
3. Tab manualisasi
4. Tab pengujian
5. Judul Sistem
6. Nama (*copyrigt*)
7. Jurusan (*copyright*)

3.2.5 Perancangan Uji Coba dan Evaluasi

Pada bagian ini akan menjelaskan mengenai pengujian data. Dalam pengujian ini, sistem menggunakan data klinis penyakit stroke sebanyak 200 data sebagai *data load*. Pengujian yang dilakukan sebanyak 5 pengujian yaitu pengujian terhadap besar parameter *learning rate* awal, parameter pengurang *learning rate*, *learning rate minimum*, jumlah *iterasi maksimum*, dan *rasio* perbandingan banyak data latih dan data uji.

3.2.5.1 Pengujian Parameter *Learning rate* awal

Besar parameter *learning rate* awal yang dilakukan pada proses pelatihan memiliki pengaruh terhadap besar akurasi yang dihasilkan sistem oleh karena itu perlu dilakukan pengujian dengan menggunakan parameter ini. Tujuannya adalah untuk mengetahui pengaruh parameter learning rate awal terhadap hasil akurasi dan banyak *iterasi* yang dijalankan selama proses pelatihan. Besar learning rate awal yang digunakan adalah 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, dan 0.9 . Masing-masing pengujian dilakukan sebanyak 10 kali percobaan. Dari hasil percobaan tersebut maka dapat diketahui rata-rata akurasi yang dihasilkan. Skenario pengujian terhadap parameter *learning rate* awal ditunjukkan pada *Tabel 3.10.*

Tabel 3.10 Skenario Pengujian Terhadap Parameter Learning Rate Awal

Learning Rate Awal	Hasil Akurasi Awal					Rata - rata akurasi %
	Percobaan ke-i					
	1	2	3	4	5	
0.1						
0.2						
0.3						
0.4						
0.5						
0.6						
0.7						
0.8						
0.9						

3.2.5.2 Pengujian Parameter Pengurang *Learning rate*

Besar pengurang *learning rate*yang digunakan pada proses pelatihan juga mempengaruhi akurasi yang dihasilkan dari sistem yang dibuat. Tujuan pengujian ini adalah untuk mengetahui pengaruh nilai *learning rate* yang digunakan terhadap hasil akurasi dan jumlah *iteras*i yang dijalankan selama proses pelatihan. Parameter pengurang *learning rate* yang digunakan adalah 0.1, 0.2,0.3,0.4,0.5,0.6,0.7,0.8,dan 0.9 . Masing-masing pengujian dengan parameter tersebut dilakukan percobaan sebanyak 10 kali. Dari hasil tersebut maka dapat diketahui rata-rata akurasi yang dihasilkan. Skenario pengujian terhadap penentuan parameter pengurang learning rate ditunjukkan pada *Tabel 3.11*.

Tabel 3.11 Skenario Pengujian Penentuan Parameter Pengurang *Learning Rate*

Pengurang *Learning Rate*	Nilai Akurasi (%) Percobaan ke-i					Rata-rata akurasi
	1	2	3	4	5	
0.1						
0.2						
0.3						
0.4						
0.5						
0.6						
0.7						
0.8						
0.9						

3.2.5.3 Pengujian *Learning rate Minimum*

Besar *learning rate minimum* berpengaruh pada proses pelatihan juga mempengaruhi akurasi yang dihasilkan oleh sistem. Tujuan dari pengujian ini adalah untuk mengetahui pengaruh nilai *learning rate minimum* terhadap hasil akurasi dan jumlah *iterasi* yang dijalankan selama proses pelatihan. Parameter Learning rate minimum yang digunakan adalah 0.01, 0.001 , 0.0001 ,0.00001, 0.000001 (Purwandari., 2015). Diambil *learning rate* dengan nilai-nilai tersebut dikarenakan pada nilai tersebut sudah mendapatkan nilai akurasi yang baik, sehingga dapat dilakukan analisa. Untuk masing masing *learning rate minimum* dilakukan percobaan sebanyak 10 kali. Skenario pengujian penentuan parameter *learning rate minimum* ditunjukkan pada *Tabel 3.12*.

Tabel 3.12 Skenario Pengujian *Learning Rate Minimum*

Learning Rate Minimum	Nilai Akurasi (%) Percobaan ke-i					Rata - rata akurasi %
	1	2	3	4	5	
0.01						
0.001						
0.0001						
0.00001						
0.000001						

3.2.5.4 Pengujian Jumlah Iterasi Maksimum

Pengujian jumlah *iterasi maksimum* bertujuan untuk mengetahui pengaruh jumlah *iterasi maksimum* terhadap tingkat *akurasi* yang dihasilkan pada proses pelatihan. Jumlah *iterasi maksimum* yang digunakan adalah 3,5,7,10,12 dan 15. Diambil nilai tersebut dikarenakan tingkat *akurasi* yang dihasilkan sudah tinggi sehingga dapat dilakukan analisa. Masing-masing pengujian dilakukan sebanyak 10 kali percobaan. Dari percobaan tersebut dihasilkan rata-rata akurasi .

Tabel 3.13 Skenario Pengujian Jumlah *Iterasi Maksimum*

Iterasi Maksimum	Nilai Akurasi (%) Percobaan ke-i					Rata - rata akurasi %
	1	2	3	4	5	
100						
1000						
10000						
100000						
1000000						

3.2.5.5 Pengujian Perbandingan Jumlah Data Latih dan Data Uji

Pengujian ini bertujuan untuk mengetahui pengaruh banyaknya jumlah data latih dan data uji terhadap akurasi yang dihasilkan selama dilakukannya proses pelatihan.Perbandingan yang digunakan adalah 50 : 50, 60 : 40, 70 : 30, 80 : 20, 90 : 10. Masing-masing pengujian terhadap data dilakukan sebanyak 10 kali percobaan. Dari hasil tersebut dapat diketahui akurasi terbaik pada perbandingan data latih data uji yang mana dan dapat diketahui rata-rata akurasi yang dihasilkan. Skenario pengujian perbandingan antara data latih dan data uji ditunjukkan pada tabel 3.14 .

Tabel 3.14 Skenario Pengujian Perbandingan data latih dan data uji

Rasio	Nilai Akurasi (%)					Rata - rata akurasi %
	Percobaan ke-i					
	1	2	3	4	5	
50 : 50						
60 : 40						
70 : 30						
80 : 20						
90 : 10						

BAB 4 IMPLEMENTASI

Bab ini membahas mengenai implementasi sistem yang akan dibuat untuk Klasifikasi resiko penyakit *stroke* menggunakan algoritma *self organizing map* berdasarkan analisis data perancangan yang dilakukan pada bab sebelumnya.Pada Bab ini bahasan implementasi meliputi *spesifikasi sistem*, batasan *sistem*, implementasi algoritma dalam sistem dan implementasi antarmuka.

4.1 Spesifikasi Lingkungan Implementasi

Spesifikasi Lingkungan Implementasi yang akan dibahas dalam sub-bab ini terkait dengan spesifikasi perangkat lunak dan juga perangkat keras yang digunakan dalam mengimplementasikan aplikasi yang digunakan dalam penelitian ini.

4.1.1 Spesifikasi Perangkat Keras

Spesifikasi perangkat keras yang digunakan dalam penelitian kali ini adalah sebagai berikut:

1. Processor Intel Core 2 Duo CPU T 5250@1.50 G.Hz
2. Memory 3 GB
3. Monitor 13''

4.1.2 Spesifikasi Perangkat Lunak:

Perangkat lunak yang digunakan untuk mengembangkan sistem dan penelitian ini terdiri dari :

1. Sistem Operasi Windows 7 Home Premium 32 bit
2. Sublime Text 3 for PHP Developers
3. Microsoft Office Visio 2007

4.2 Batasan Aplikasi

Pada sub-bab ini dibahas batasan sistem yang memfasilitasi user dalam mengakses klasifikasi resiko penyakit stroke. Berikut merupakan batasan-batasan sistem antara lain :

1. Perangkat lunak klasifikasi resiko penyakit *stroke* dirancang dan dijalankan menggunakan *web base application*.
2. Format file yang digunakan pada penelitian ini adalah *.html*, *.php* dan *.css*
3. Keluaran yang dihasilkan sistem adalah pada tahapan manualisasi *output* berupa proses data *training*,data *testing*,proses *normalisasi*,proses *inisialisasi* data, jarak antara data dengan

bobot,proses *update* bobot sedangkan pada proses pengujian *output* berupa tingkat akurasi pada masing-masing pengujian.

4.3 Implementasi Algoritma

Pada sub-bab ini akan dibahas mengenai implementasi dari algoritma yang digunakan dalam sistem klasifikasi resiko penyakit stroke ini. Algoritma yang digunakan meliputi algoritma *Self Organizing Map* (SOM),dan tingkat akurasi yang dihasilkan.

4.3.1 Proses Pembelajaran SOM

4.3.1.1 Load Data

Langkah awal pada implementasi algoritma Self Organizing Maps (SOM) adalah load data. Proses ini berfungsi untuk memilih dan menentukan data yang akan diproseske dalam proses *training* dan *testing* data.*Source code* yang digunakan untuk proses load data ditunjukkan pada *Source Code 4.1.*

```
1   $atraining = [7,9,121,122,161,162];
2   $atesting = [5,130,170];
3   $training = getDataIn($atraining);
4   $testing = getDataIn($atesting);

    // remove class (array 5)
5   $i=0;foreach ($training as $value) {
    //unset($training[$i][5]);
6   unset($testing[$i][5]);$i++;
            }

7   function getDataIn($in){
8      $m = app\models\Dataset::find()->where(['in','id',$in])->all();
9      $training = [];
10  foreach($m as $em){
11         $training[] = [
12  $em->umur, $em->kolesterol, $em->hdl,$em->ldl, $em->trigliserida,
    $em->kelompok
           ];
13     } return $training;
    }
```

Source Code 4.1 Load Data

Baris 1-2 : Inisialisasi Variabel yang menampung nilai kumpulan ID yang digunakan untuk memanggil data set di database.

Baris 3-4 : Inisialisasi Variabel yang menampung hasil dari pemanggilan data set untuk training dan testing.

Baris 5-6 : Untuk menghilangkan pelabelan pada proses testing.

Baris 7-13 : Fungsi yang digunakan untuk memanggil dataset di database sesuai dengan parameter yang minta.

4.3.1.2 Normalisasi Data

Langkah selanjutnya adalah *normalisasi* data. Pada proses ini data training dan data testing dilakukan proses *normalisasi* dengan menggunakan persamaan (2.3). Source code untuk melakukan normalisasi data ditunjukkan pada *Source Code 4.2*.

```
1 $normal = minmaxfix($training, $testing);
2       $minmaxtraining = $normal['training'];
3       $minmaxtesting = $normal['testing'];
```

Source Code 4.2 Normalisasi Data

Baris 1-3 : Variabel normal pada baris 1 menyimpan nilai dari minmax, hasil dari proses normalisasi. Variabel normal pada baris 2 menyimpan nilai dari minmax dari proses normalisasi training data.Pada baris 3 variabel normal menyimpan nilai dari minmax dari proses normalisasi testing data.

4.3.1.3 Inisialisasi Bobot

Langkah selanjutnya adalah *inisialisasi* bobot.Pada proses *inisialisas*i bobot menggunakan bobot awal random yang digunakan untuk melakukan proses *training* SOM. *Source code* pada implementasi program *inisialisasi* bobot ditunjukkan pada *Source Code 4.3.*

```
1  class SOM
2  {
3  private function random_value(){
4  return abs((double)(rand())/(32767/2) - 1);
     }

5  private function initBobot($criteria, $class){
6  for ($j=0; $j <$class ; $j++) {
7  for ($i=0; $i <$criteria ; $i++) {
8  $init[$j][$i] = $this->random_value();
                    }
   }
9          return($init);
     }
```

Source Code 4.3 Inisialisasi Bobot

Baris 2-4 : Menentukan nilai bobot secara *random*.
Baris 5-8 : Proses *inisialisasi* berdasarkan kelas dan kriteria.

4.3.1.4 *Training* SOM

Langkah selanjutnya adalah proses *training* SOM. Proses ini dapat dijalankan setelah melalui proses *load data* dan normalisasi data. Source code untuk melakukan proses *training* SOM ditunjukkan pada *Source Code 4.4* .

```
1  $initw = $w = $this->initBobot($criteria,$class);
2        $lrate = $config['learningrate_init'];
3        $i=0;
4        while(true){

            // each data
5           foreach ($training as $j => $data) {
6              $d = $this->getDistance($data, $w); $var[$i][$j]['d'] = $d;
7              $w = $this->updateWeight($data, $w, $d, $lrate);
8  $var[$i][$j]['w'] = $w;
            }

   // update learning rate
9  $lrate = $lrate * $config['learningrate_delta']; $var[$i]['lrate'] = $lrate;

10  if($lrate < $config['learningrate_min'] or $i > $config['max_iterasi'])
   break;
   $i++;}
```

Source Code 4.4 Training SOM

Baris 1 : Proses *inisialisasi* bobot.
Baris 2 : Proses *inisialisasi* nilai *learning rate*.
Baris 3-8 : Proses perulangan pada data *training* untuk perhitungan jarak antara data dengan bobot.Hasil dari perhitungan tersebut digunakan untuk prosese selanjutnya yaitu update bobot.
Baris 9-10 : *Update learning rate*.

4.3.1.5 Jarak Antara Data dengan Bobot

Dalam proses *training* dan *testing* SOM terdapat proses perhitungan jarak antara data dengan bobot. Untuk mencarai jarak antara data dengan bobot menggunakan persamaan (2.1). *Source code* untuk proses perhitungan Jarak antara data dengan bobot ditunjukkan pada *Source Code 4.5.*

```
1  private function getDistance($data, $w)
   {
2  foreach($w as $class => $ew){
3  $total =0;
```

```
4  foreach ($data as $criteria => $edata) {
5  if(isset($ew[$criteria]))
6  $total += pow($ew[$criteria] - $data[$criteria],2);
       }
7  $distance[$class] = ($total);
     }
8  return $distance;}
```

Source Code 4.5 Jarak Antara Data Dengan Bobot

Baris 1-8 : Proses pengambilan jarak menggunakan data dan bobot.
Baris 6 : Proses penghitungan jarak antara data dengan bobot. Proses penghitungan jarak menggunakan persamaan (2.1).

4.3.1.6 *Update* Bobot

Pada proses *training* terdapat proses *update* bobot.Sedangkan untuk proses testing tidak menggunakan proses *update* bobot.Untuk melakukan proses *update* bobot dapat menggunakan persamaan (2.2). *Source code* untuk proses *update* bobot ditunjukkan pada *Source Code* 4.6 .

```
1  private function updateWeight($data, $w, $d, $lrate){
2  $minclass = min($d);
3  $key = array_search ($minclass, $d);
4  $oldw = $w[$key];
5  foreach ($oldw as $category => $ew) {
6  $neww[] = ( $oldw[$category] + $lrate*( $data[$category] -
7  $oldw[$category] ) );
       }
8  $w[$key] = $neww;
9   return $w;
   }
```

Source Code 4.6 Update Bobot

Baris 1-9 : Proses update bobot dengan menggunakan kriteria data, *delta* dan *learning rate*.
Baris 3 : Mencari array pada kelas minimum dan delta yang digunakan sebagai parameter untuk *update* bobot.
Baris 6-9 : Proses update bobot baru menggunakan bobot lama ditambah dengan *learning rate* dikalikan dengan data pada masing-masing kategori kemudian di kurangi bobot lama dikalikan data pada masing masing kategori kemudian dihasilkan bobot baru.

4.3.1.7 *Testing* SOM

Setelah dilakukan proses *training* SOM proses selanjutnya adalah proses *testing* SOM. Tahapan proses ini sama dengan proses *training* akan tetapi pada proses *testing* SOM tidak menggunakan *inisialisasi* bobot dan *update* bobot. Untuk bobot awal yang digunakan pada proses *testing* SOM, menggunakan bobot terakhir pada proses *training* SOM. *Source code* untuk melakukan proses testing SOM ditunjukkan pada *Source Code 4.7*.

```
1  foreach ($testing as $j => $data) {
2   $d = $this->getDistance($data, $w); $test['d'][$j] = $d;
3   $classresult[] = $this->Classify($d, $config);
   }
4   $test['classresult'] = $classresult;

5  return ['var'=>$var,'initw'=>$initw, 'test'=>$test];
   }
```

Source Code 4.7 *Testing* SOM

Baris 1 : Memanggil data *testing*.
Baris 2 : Memanggil fungsi menghitung jarak antara data dengan bobot pada data *testing*.
Baris 3-5 : Hasil yang digunakan sebagai penentuan kelas normal,rentan,stroke.

4.3.1.8 Akurasi

Implementasi dari proses akurasi dilakukan setelah proses pembelajaran SOM yaitu proses *training* dan *testing* SOM. Proses perhitungan tingkat akurasi pada penelitian ini menggunakan persamaan (2.5). Proses tersebut menghasilkan tingkat akurasi yang dihasilkan dari proses klasifikasi selama proses pembelajaran SOM. *Source Code* untuk menghitung tingkat akurasi ditunjukkan pada *Source Code* 4.8.

```
1  $class_real = [];
2          foreach($eachuji['testing'] as $key => $testingvalue) {
3             $class_real[] = $config['class'][$testingvalue[5]-1];
           }

4          $class_uji = [];
5          foreach ($return['test']['classresult'] as $value) {
6             $class_uji[] = $value;
           }

7          $jml_data = count($class_uji);
8          $valid=0;
```

```
9           for($ak=0;$ak<$jml_data;$ak++){
10            if($class_uji[$ak] == $class_real[$ak]) $valid++;
            }


11          $akurasi[$paramuji_key][$ea] = $valid/$jml_data*100;

          }
        }

12       $rata_uji[] = array_sum($akurasi[$paramuji_key])/count($ujidata);

       }


13     $rata_uji_master[$step] = $rata_uji;

14     $returnindex = array_keys($rata_uji, max($rata_uji));

15     $scenario_result_index[$step] = $returnindex[0];
       // override config
16     $config[$step] = $scenario_pengujian [$step] [ $returnindex[0] ];

17     //echo "<pre>";
18     //print_r($rata_uji);echo "</pre>";
19     //echo "<h2>Parameter : ".$step.", hasil : ".$param_uji[ $returnindex[0]
20   ]." dengan akurasi : ".max($rata_uji)."</h2>";
21     $akurasitotal[$step] = $akurasi;
     }

22   //echo "<pre>";
23   //print_r($akurasitotal);die();
```

Source Code 4.8 Akurasi

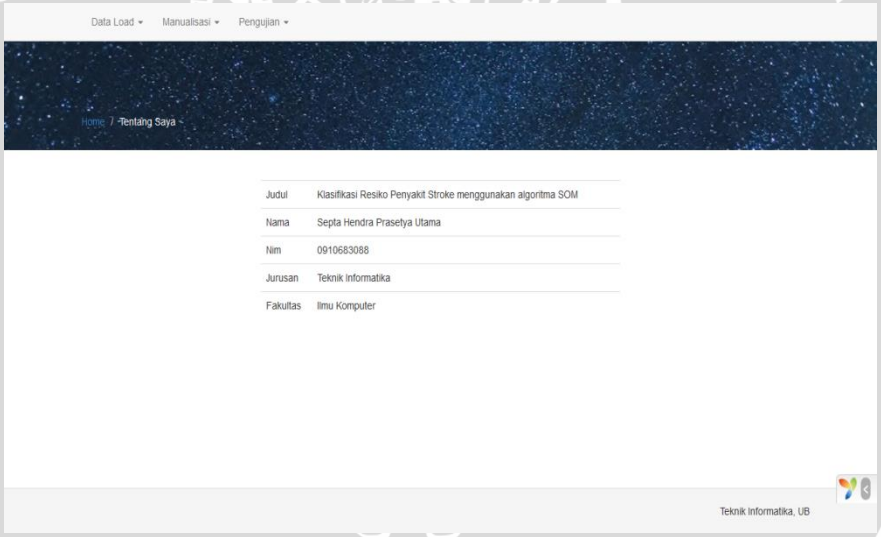
Baris 1-3 : Pengambilan data kelas asli dari data *load*.
Baris 4-6 : Pengambilan data dari proses pengujian.
Baris 7-11 : Membandingkan data asli dengan data pengujian kemudian hasil dari perbandingan tersebut dicocokan kebenarannya dengan menggunakan rumus (2.5).
Baris 12 : Hasil akurasi dari beberapa kali percobaan tersebut kemudian di rata-rata.
Baris 13-23 : Hasil dan rata-rata akurasi terbaik digunakan untuk proses pengujian selanjutnya.

4.4 Implementasi Antar Muka

Pada sub bab ini menjelaskan implementasi antar muka/ interface dari program yang dibuat yaitu klasifikasi resiko penyakit stroke dengan menggunakan algoritma *self organizing maps*. Implementasi antarmuka terdiri dari :

1. Implementasi Halaman Utama

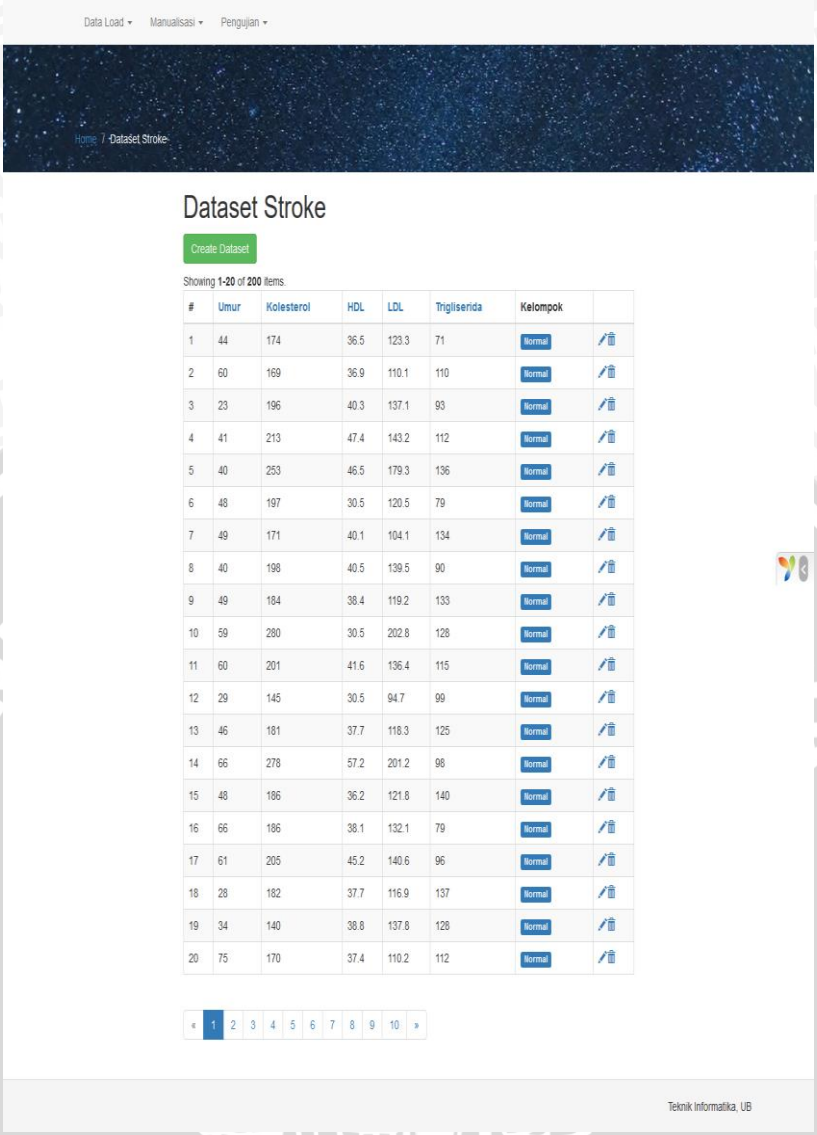
Pada Halaman utama ini terdiri dari biodata penulis dan juga menu *navigasi*. Menu *navigasi* terdiri dari *data load* yang berfungsi untuk membuat dan menyimpan data yang digunakan untuk proses *manualisasi* dan juga proses pengujian.Menu *manualisasi* adalah salah satu menu *navigasi* yang berisi perhitungan manual menggunakan metode algoritma *self organizing maps* (SOM). Di dalam manualisasi terdapat proses *normalisasi* data, *training data testing* data dan juga hasil pengklasifikasian pasien *stroke* kedalam kelas normal, rentan atau *stroke*. Menu *navigasi* Pengujian berisi 5 pengujian terdiri dari Pengujian tingkat akurasi terhadap parameter *learning rate* awal,pengujian tingkat akurasi terhadap parameter pengurang *learning rate*, pengujian tingkat akurasi terhadap parameter *learning rate* minimum, pengujian tingkat akurasi terhadap jumlah *iterasi* maksimum, pengujian tingkat akurasi terhadap perbandingan data latih dan data uji.



Gambar 4.1 Halaman Utama

2. Implementasi *Data Load*

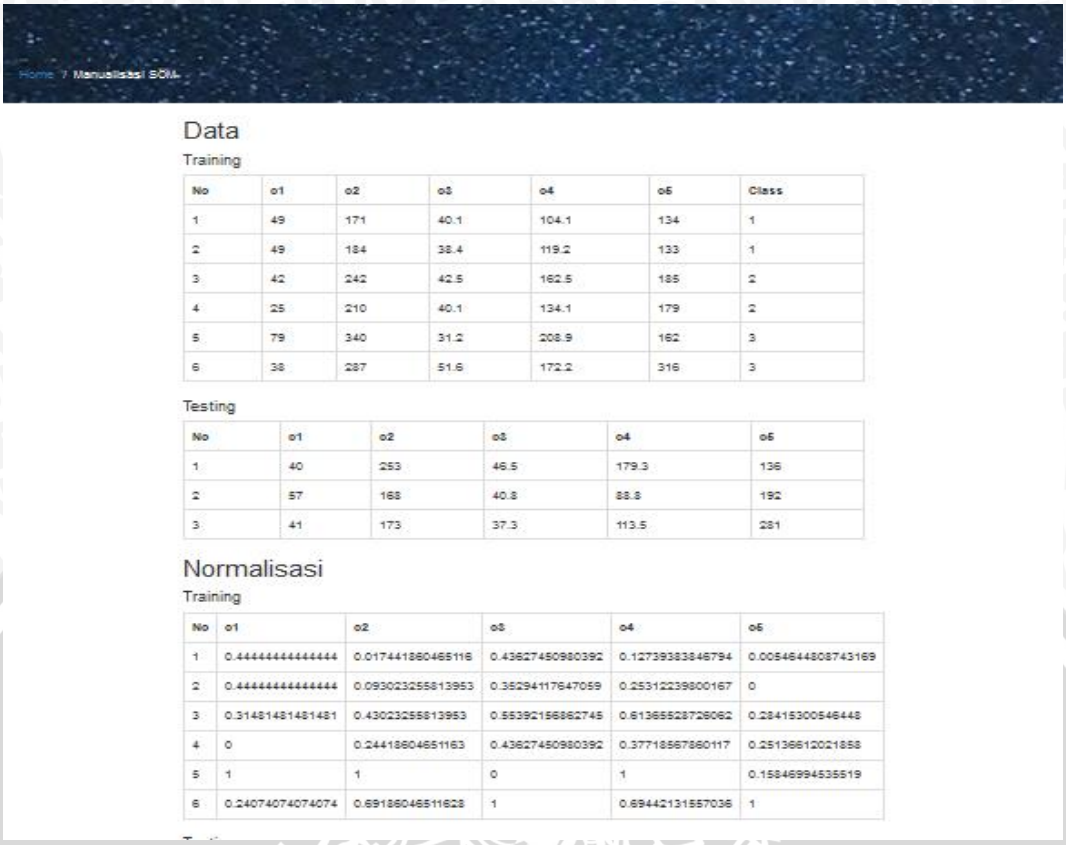
Implementasi pada *menu navigasi* data *load* berisi dataset *pasien stroke* yang didapatkan dari data Laboratorium Klinik Sejahtera Probolinggo. Penulis mengubah data tersebut menjadi ke dalam bentuk database yang digunakan untuk proses pengambilan data pada sistem klasifikasi resiko penyakit *stroke* menggunakan algoritma *self organizing maps* (SOM).



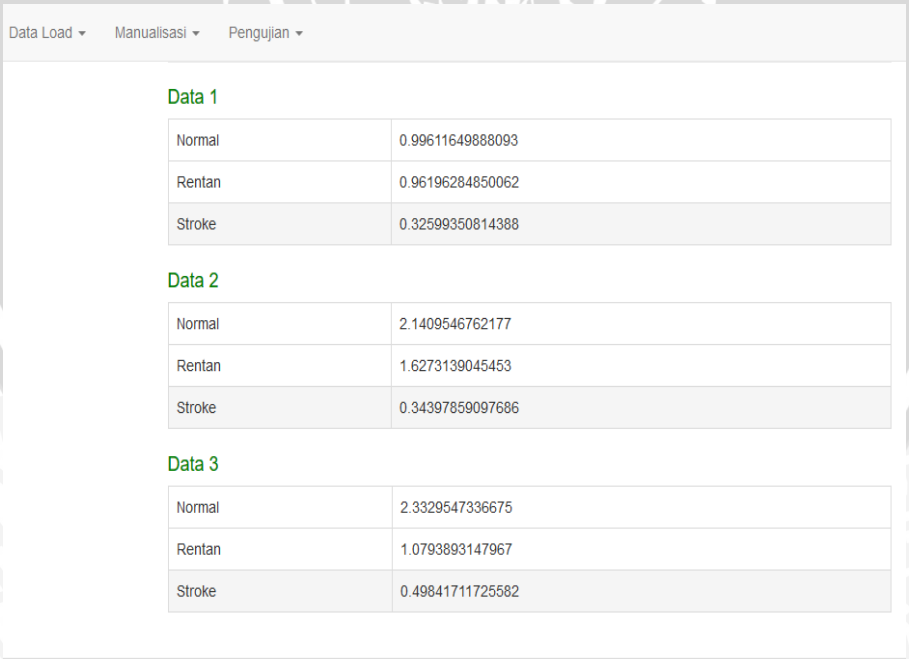
Gambar 4.2 Halaman Data *Load*

3. Implementasi *Manualisasi*

Pada menu *navigasi manualisasi* terdapat proses *normalisasi* data kemudian proses *training* data untuk mencari jarak terkecil yang digunakan untuk *update* bobot. Dalam menu *navigasi* ini juga terdapat hasil bobot terakhir pada tahapan proses *training* data yang digunakan untuk proses pencarian jarak terkecil pada *testing* data. Jarak terkecil digunakan sebagai hasil akhir atau pemenang pada rangkaian proses ini. Fungsi menu navigasi *manualisasi* ini adalah untuk mengimplementasikan perhitungan *manual* pada penelitian ini kedalam bentuk *aplikasi* berbasis *web*.



Gambar 4.3 Halaman Manualisasi Pada Tahap Normalisasi



Gambar 4.4 Halaman Manualisasi Pada Tahapan *Testing* Data

BAB 5 PENGUJIAN DAN ANALISIS

Pada bab ini membahas pengujian dan analisis hasil implementasi algoritma Self Organizing map (SOM) untuk klasifikasi resiko penyakit stroke. Proses dari pengujian ini telah ditentukan pada skenario sebelumnya yaitu meliputi pengujian terhadap parameter *learning rate* awal, pengurang *learning rate*, *learning rate* minimum, *iterasi* maksimum, dan pengujian tingkat akurasi terhadap perbandingan jumlah data latih dan data uji.

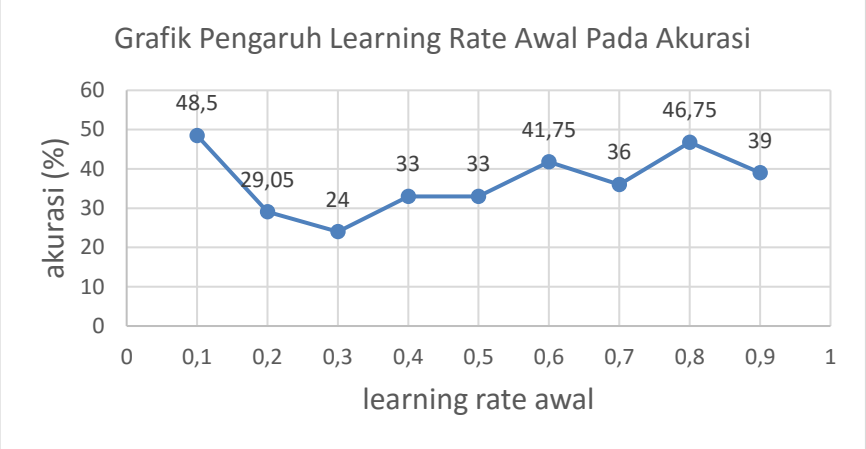
5.1 Hasil dan Analisis Pengujian Tingkat Akurasi Terhadap Parameter *Learning rate* awal

Pada sub-bab ini membahas tentang hasil dan analisis pengujian tingkat akurasi terhadap parameter learning rate awal pada algoritma *Self-Organizing Maps* (SOM). Tujuan dari pengujian ini adalah untuk mengetahui pengaruh *learning rate* awal terhadap rata-rata akurasi yang dihasilkan dari lima kali percobaan dengan nilai *learning rate* awal 0.6 pengurang *learning rate*=0.5 *learning rate* minimum 0.000001 dengan perbandingan antara data latih dan data uji 60 : 40. Data yang digunakan dalam pengujian ini adalah 200 data. Data tersebut terdiri dari 118 kelas 1 atau normal,37 kelas 2 atau rentan dan 45 kelas 3 atau stroke. Maksimum *iterasi* yang digunakan pada pengujian ini adalah 100000. Hasil pengujian tingkat akurasi berdasarkan pengaruh nilai *learning rate* awal ditunjukkan tabel 5.1.

Tabel 5.1 Tingkat Akurasi Terhadap Parameter *Learning Rate* Awal

Learning Rate Awal	Hasil Akurasi Awal					Rata - rata akurasi %
	Percobaan ke-i					
	1	2	3	4	5	
0.1	23.75	60	35	62.5	61.25	**48.5**
0.2	17.5	21.5	28.75	55	22.5	29
0.3	46.25	20	15	16.25	22.5	24
0.4	31.25	21.25	37.5	53.75	21.25	33
0.5	22.5	22.5	35	47.5	37.5	33
0.6	23.75	28.75	43.75	51.25	61.25	41.75
0.7	20	60	41.25	20	38.75	36
0.8	45	30	35	62.5	61.25	46.75
0.9	22.5	52.5	33.75	52.5	33.75	39

Pada tabel 5.1 didapatkan rata-rata akurasi terbaik 48.5 % dengan *learning rate* awal 0.1 . Hasil tersebut dengan demikian akan digunakan untuk pengujian selanjutnya dengan nilai 0.1 digunakan sebagai nilai *learning rate* awal.



Gambar 5.1 Grafik Pengaruh Parameter *Learning Rate* Awal Terhadap Akurasi

Pada Gambar 5.1 untuk pengujian parameter nilai *learning rate* awal didapatkan rata-rata akurasi terbaik pada *learning rate* awal sebesar 0.1 dengan rata-rata akurasi sebesar 48.5 dan mengalami tren penurunan pada *learning rate* awal 0.2, 0.3 dan kemudian kembali mengalami tren kenaikan pada *learning rate* 0.4 sampai dengan 0,9 dengan grafik yang *fluktuatif*.

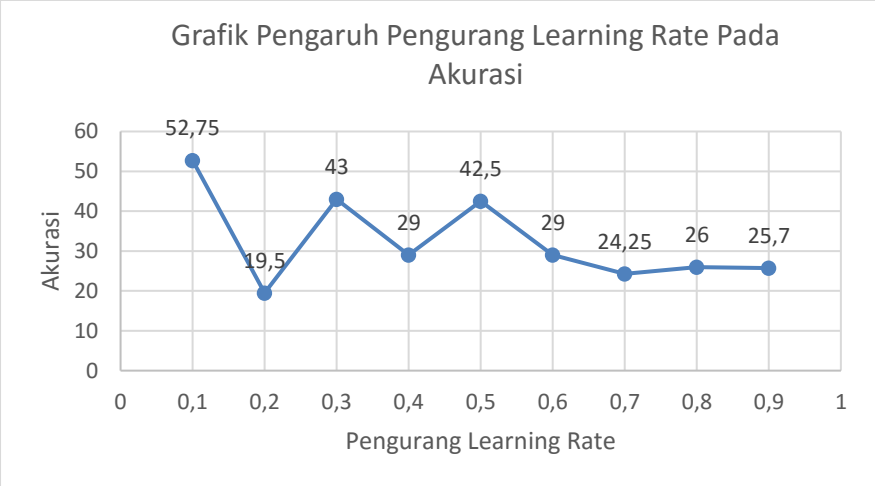
5.2 Hasil dan Analisis Pengujian Tingkat Akurasi Terhadap Parameter Pengurang *Learning rate*.

Pada sub-bab ini membahas tentang hasil dan analisis pengujian tingkat akurasi terhadap parameter pengurang *learning rate* pada algoritma SOM. Tujuan dari pengujian ini adalah untuk mengetahui pengaruh pengurang *learning rate* terhadap tingkat akurasi pada algoritma SOM. Pengujian ini juga bertujuan untuk mengetahui pengaruh pengurang *learning rate* terhadap rata-rata akurasi yang dihasilkan dari 5 kali percobaan. Parameter yang digunakan pada penelitian ini adalah *learning rate* awal sebesar 0.1 yang didapatkan dari hasil rata-rata terbaik pada pengujian sebelumnya, pengurang *learning rate* 0.5 dan *learning rate* minimum sebesar 0.000001 dan maksimum *iterasi* sebesar 100000. Hasil pengujian tingkat akurasi terhadap parameter pengurang *learning rate* ditunjukkan tabel 5.2.

Tabel 5.2 Tingkat Akurasi Terhadap Parameter Pengurang *Learning Rate*

Pengurang Learning rate	Hasil Akurasi Awal Percobaan ke-i					Rata - rata akurasi %
	1	2	3	4	5	
0.1	56.25	58.75	65	62.5	21.25	**52.75**
0.2	25	20	15	20	17.5	19.5
0.3	61.25	58.75	17.5	16.25	61.25	43
0.4	18.75	23.75	52.5	26.25	23.75	29
0.5	55	58.75	57.5	18.75	22.5	42.5
0.6	28.75	22.5	21.25	53.75	18.75	29
0.7	51.25	20	16.25	17.5	16.25	24.25
0.8	18.75	20	22.5	52.5	16.25	26
0.9	51.25	21.25	16.25	22.5	17.25	25.75

Pada pengujian ini didapatkan hasil rata-rata terbaik 52.75 % dengan nilai pengurang *learning rate* sebesar 0.1. Dengan demikian 0.1 dijadikan parameter pengurang *learning rate* untuk proses pengujian berikutnya.



Gambar 5.2 Grafik Pengaruh Pengurang *Learning Rate* Terhadap Akurasi

Pada Gambar 5.2 akurasi terbaik didapatkan ketika pengurang *learning rate* 0.1 dengan nilai akurasi sebesar 52,75. Kemudian terjadi penurunan pada *learning rate* 0.2 dan terjadi tren kenaikan ketika pengurang *learning rate* berada pada *learning rate* 0.3 sampai dengan 0.9. Dalam rentang 0.3 sampai dengan 0.9 tingkat akurasi naik dan turun secara *fluktuatif*.

5.3 Hasil dan Analisis Pengujian Tingkat Akurasi Terhadap Parameter Learning Rate *Minimum.*

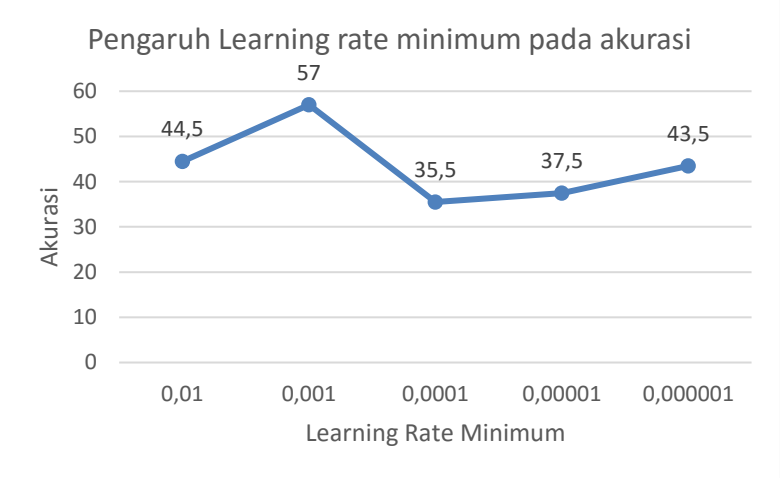
Pada sub-bab ini membahas tentang analisis dan pengujian tingkat akurasi terhadap parameter *learning rate* minimum. Tujuan dari pengujian ini adalah untuk mengetahui pengaruh parameter *learning rate* minimum terhadap rata-rata

akurasi yang dihasilkan dari lima kali percobaan dengan menggunakan 200 data. Pada pengujian ini menggunakan parameter *learning rate* awal sebesar 0.1, pengurang *learning rate* 0.1, *learning rate* minimum sebesar 0.000001 dan maksimum iterasi sebesar 100000.

Tabel 5.3 Tingkat Akurasi Terhadap Parameter *Learning Rate* Minimum

Learning Rate Minimum	Nilai Akurasi (%) Percobaan ke-i					Rata - rata akurasi
	1	2	3	4	5	%
0.01	20	58.75	62.5	20	61.25	44.5
0.001	56.25	53.75	56.25	58.75	60	**57**
0.0001	23.75	60	37.5	42.5	13.75	35.5
0.00001	18.75	58.75	63.75	25	21.25	37.5
0.000001	56.25	58.75	16.25	23.75	61.25	43.25

Pada pengujian ini didapatkan hasil rata-rata akurasi terbaik pada learning rate minimum 0.001 dengan demikian 0.001 digunakan untuk pengujian berikutnya sebagai parameter *learning rate* minimum.



Gambar 5.3 Grafik Pengaruh Learning Rate Minimum Terhadap Akurasi

Pada Gambar 5.3 pada grafik pengaruh *learning rate* minimum dapat dilihat bahwa hasil rata-rata akurasi terbaik terjadi pada *learning rate* minimum 0.001 dengan rata rata akurasi 57. Terjadi tren penurunan pada saat *learning rate* minimum 0.0001 dan rata rata akurasi berangsur naik di kisaran *learning rate* 0,00001 sampai dengan 0,000001.

5.4 Pengujian Tingkat Akurasi Terhadap Jumlah Iterasi *Maksimum*

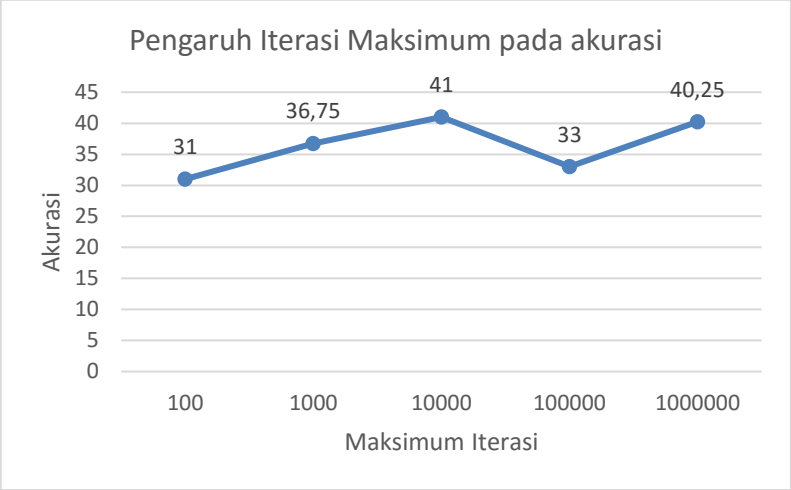
Pada sub-bab ini membahas tentang analisis dan pengujian tingkat akurasi terhadap jumlah *iterasi* maksimum. Tujuan pengujian ini adalah untuk mengetahui besar kecilnya nilai *iterasi* maksimum berpengaruh terhadap pemberhentian pembelajaran pada algoritma SOM terhadap rata-rata akurasi yang dihasilkan dalam lima kali percobaan. Parameter yang digunakan dalam pengujian ini adalah

dengan menggunakan bobot awal *random*,*learning rate* awal = 0.1, pengurang *learning rate* 0.1, *learning rate* minimum 0.001 dengan rasio perbandingan jumlah data latih 60, data uji 40 dan maksimum *iterasi* 100000.

Tabel 5.4 Tingkat Akurasi Terhadap Jumlah *Iterasi Maksimum*

Iterasi Maksimum	Nilai Akurasi (%) Percobaan ke-i					Rata - rata akurasi %
	1	2	3	4	5	
100	31.25	21.25	17.5	62.5	22.5	31
1000	26.25	60	18.75	60	18.75	36.75
10000	50	21.25	15	56.25	62.5	**41**
100000	20	18.75	57.5	15	53.75	33
1000000	55	58.75	37.5	23.75	26.25	40.25

Pada Tabel 5.4 diketahui hasil dan rata-rata akurasi terbaik diperoleh dengan menggunakan maksimum *iterasi* 10000. Berdasarkan hal tersebut nilai maksimum *iterasi* sebesar 10000 digunakan untuk pengujian selanjutnya.



Gambar 5.4 Grafik Pengaruh Maksimum Iterasi Terhadap Akurasi

Grafik pada Gambar 5.4 terjadi tren kenaikan pada maksimum *iterasi* 100,1000 dan 10000 dengan rata-rata akurasi terbaik pada maksimum *iterasi* 10.000 dengan nilai akurasi 41. Terjadi penurunan pada maksimum iterasi 100000 dan naik lagi ketika maksimum iterasi bernilai 1000000.

5.5 Pengujian Tingkat Akurasi Terhadap Perbandingan Jumlah Data Latih dan Data Uji

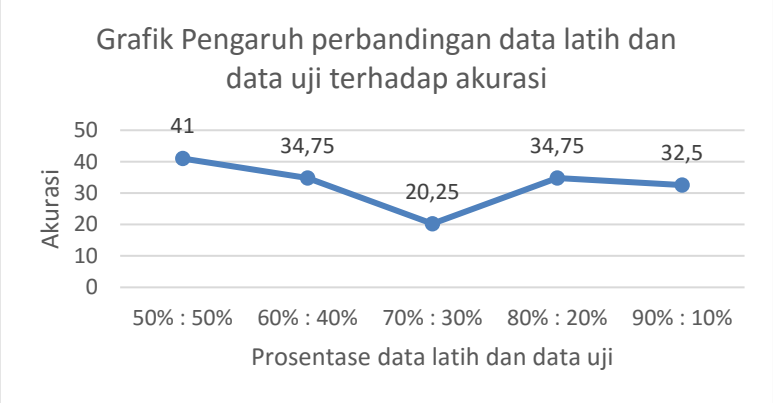
Pada sub-bab ini menjelaskan tentang hasil dan analisis pengujian tingkat akurasi menggunakan perbandingngan data latih dan dat uji.Tujuan dari pengujian ini adalah untuk mengetahui pengaruh jumlah data latih dan data uji yang digunakan terhadap rata-rata akurasi yang dihasilkan dari lima kali *percobaan*

menggunakan,*learning rate* awal =0.1, pengurang *learning rate*=0.1, *learning rate minimum* = 0.001 dan *maksimum iterasi* = 10000 dengan perbandingan data latih 60 dan data uji 40 dari 200 data. Berikut ini merupakan tabel hasil pengujian tingkat akurasi terhadap perbandingan jumlah data latih dan data uji.

Tabel 5.5 Tingkat Akurasi Terhadap Perbandingan Data Latih Dan Data Uji

Rasio	Nilai Akurasi (%)					Rata - rata akurasi %
	Percobaan ke-i					
	1	2	3	4	5	
50 : 50	20	20	41.25	66.25	57.5	**41**
60 : 40	21.25	56.25	13.75	17.5	65	34.75
70 : 30	26.25	22.5	20	17.5	15	20.25
80 : 20	18.75	21.25	60	57.5	16.25	34.75
90 : 10	25	21.25	36.25	63.75	16.25	32.5

Hasil terbaik pada pengujian ini adalah 50:50 dengan rata-rata akurasi sebesar 41% .



Gambar 5.5 Pengaruh perbandingan data latih dan data uji terhadap akurasi

Pada Gambar 5.5 ditunjukkan hasil akurasi perbandingan antara data latih dan data uji. Tingkatan akurasi tertinggi pada perbandingan data latih dan data uji terdapat pada rasio 50:50 dengan rata-rata akurasi sebesar 41%. Pada rasio, 70:30 terjadi tren penurunan. Hal ini dikarenakan data yang dipakai untuk proses pembelajaran untuk masing-masing 5 kali percobaan sangat *variatif* seperti yang ditunjukkan pada tabel 5.5. Hal ini dikarenakan oleh cara kerja Algoritma *Self Organizing Maps* yang bekerja secara *sekuensial* dan sifat pembelajarannya yang un*supervised,* dimana hasil bobot sebelumnya akan digunakan untuk proses selanjutnya. Apabila ditemukan data dengan nilai yang bervariasi dan terlalu jauh maka akan menghasilkan update bobot yang terlalu jauh juga. Hal tersebut juga berpengaruh pada bobot akhir yang dihasilkan pada proses pelatihan.

BAB 6 PENUTUP

Bagian penutup ini memuat kesimpulan dan saran terhadap skripsi. Kesimpulan dan saran disajikan secara terpisah, dengan penjelasan sebagai berikut:

6.1 Kesimpulan

Dari hasil pengujian dan analisis yang dilakukan maka kesimpulan yang dapat diambil dari penelitian ini adalah:

1. Klasifikasi terhadap resiko penyakit stroke dapat di implementasikan menggunakan Algoritma Self Organizing Maps (SOM) dengan melalui berbagai tahapan-tahapan proses menghitung jarak antara bobot dengan data, mencari jarak minimum dan update bobot sehingga menghasilkan klasifikasi yang sesuai dengan algoritma yang digunakan.
2. Pada pengujian sistem ini didapatkan hasil rata-rata terbaik sebesar 41 %, dengan data latih sebanyak 50%, data uji sebanyak 50 % yang ada dengan menggunakan parameter *learning rate* awal = 0.1, pengurang *learning rate*= 0.1, *learning rate* minimum 0.001 dan maksimum iterasi sebesar 10000. Rata rata akurasi yang didapatkan cukup kecil, Hal ini dikarenakan data yang digunakan terlalu banyak yaitu 200 data. Sehingga ada kemungkinan beberapa data yang digunakan pada proses pelatihan masuk ke dalam kelas yang bukan aslinya atau *outlier.*

6.2 Saran

1. Untuk penelitian selanjutnya dalam penggunaan algoritma SOM perlu dilakukan optimalisasi pada tahapan proses *training* data. Karena dalam algoritma SOM terdapat tahapan proses untuk melakukan inisialisasi bobot secara *random* .
2. Data latih yang akan digunakan perlu dilakukan optimalisasi, Hal ini perlu dilakukan karena metode SOM bekerja secara sekuensial.
3. Untuk penelitian selanjutnya apabila menggunakan dataset yang berjumlah banyak, perlu ditambahkan metode lain untuk menghilangkan beberapa data yang *outlier.* Sehingga didapatkan hasil akurasi yang lebih baik pada saat proses klasifikasi data. Beberapa Metode seperti *Hierrarchical Clustering* atau *Metode Clustering Using Representative* dapat diimplementasikan untuk menghapus data *outlier* (Gita., et. All., 2015).

DAFTAR PUSTAKA

A.M.A (American Medical Association) , 2001. *Executive Summary of the Third Report of the National Cholesterol Education Program (NCEP) Expert Panel on Detection, Evaluation, and Treatment of High Blood Cholesterol in Adults (Adult Treatment Panel III*) JAMA (The Journal of the American Medical Association). Vol 289, No. 19, 2486-2497.

Alodokter.com, *Kolesterol Tinggi.,*2015.

Andreas C Widjaja, *Tesis. Uji Diagnostik Pemeriksaan Kadar D-Dimer Plasma Pada Diagnosis Stroke Iskemik*, Patologi Klinik Fakultas Kedokteran Universitas Diponegoro, Semarang 2010.

Ekowati Rahajeng, *Upaya Pengendalian Penyakit tidak Menular Di Indonesia*, Buletin Penyakit Tidak Menular ISSN 2088-270X, Semester II 2012, pp.23-28, Jakarta, 2011

Fitra Riaji, Dhanang., Hariadi, Moch. 2011. *Klasifikasi Kekerapan Kunjungan Lokasi Berbasis Location Based Service (LBS) Menggunakan Self Organizing Maps. Pasca Sarjana Jaringan Cerdas Multimedia* (Game Technology), Teknik Elektro Technology Industri ITS, Surabaya.

Gita Bhagawad, Yoga., Saikhu., Akhmad., 2015. Implementasi Deteksi Outlier Pada Algoritma *Hierarchical Clustering*., Jurusan Teknik Informatika Fakultas Teknologi Informasi Institut Teknologi Sepuluh November (ITS), Surabaya, 60111, Indonesia.

Gregorius Satia Budhi , Rudi Adipranata, Bachtiar Anwar, Bambang Setiahadi, Adrian Hartanto N, Alvin Nathaniel T. 2012. Universitas Kristen Petra, Surabaya., Pusat Pemanfaatan Sains Antariksa Lembaga Penerbangan dan Antartika Nasional (LAPAN).

Harsono. Ed.,*Buku Ajar Neurologi Klinis*. Yogyakarta: Gadjah Mada University Press, 2005.

Irwansyah, Edi, Faisal, Muhammad., 2015. *Advanced Clustering Teori dan Aplikasi*.Edisi.1, Cetakan .1. Deepubulish, Yogyakarta.

Kristanto, Andri. 2004. *Jaringan Syaraf Tiruan (Konsep Dasar Algoritma dan Aplikasi.*Grava Media, Yogyakarta.

Mochammad Bahrudin, *Model Diagnostik Stroke Berdasarkan Gejala Klinis, Jurnal Saintika* Medika Vol.6 No.13 Universitas Muhammadiyah, Malang,2010.

Purwandari, Kartika. 2015. *Implementasi Metode Jaringan Syaraf Tiruan Self Organizing Maps (SOM) Untuk Klasifikasi Kualitas Tanaman Nilam Pada Citra Daun*. Universitas Brawijaya, Malang.

Pusat Data dan Informasi Kementerian Kesehatan ., 2014. RI JL. HR Rasuna Said., Blok X5., Kav 4-9., Lantai 6 ., Blok C ., Jakarta Selatan.

Quantization.Teknik Informatika, Politeknik Negeri Malang., Teknik Elektro Universitas Brawijaya, Malang.

S.M.M Naidu., Uttam R. Baggal., Prem C. Pandey.,Suhas Hardas.,Niranjan D.Khambete, 2015. *Monitoring of stroke volume through Impedance cardiography using an artifical neural network*. Dept. Of Electrical Engineering, Indian Institute of Technology Bombay, Mumbai, India., DIST, International Institute of Information Technology,Pune,India., Dept of Biomedical Engineering, MGM College ofEngineering of Technology, Navi-Mumbai, India., Hardas Heart Care, Shivajinagar, Pune, India., Dept of Clinical Engineering, Deenanath Mangeshkar Hospital and Research Centre, Pune,India.

Seto Arifianto, Aji., Sarosa Moechamad ., Setyawaty, Onny. 2014. *Klasifikasi Stroke Berdasarkan Kelainan Patologis dengan Learning Vector.*

Wahyudi Setiawan.,Nur Maslakhah., Mulaab., 2011. Sistem Pengenalan Retina Menggunakan Self Organizing Map untuk mendeteksi Retinopati Diabetika., Prodi Manajemen Informatika., Prodi Teknik Informatika., Fakultas Teknik,. Universitas Trunojoyo., Kamal Bangkalan, Madura.

Wilson, D.Randall., R.Martinez, Tony., 2001 The Need for Small Learning Rates on Large Problems. Fonix Corporation., 180 West Election Road Draper, Utah, USA., Computer Science Department., Brigham Young University Provo, Utah, USA.

Yastroki, Yayasan Stroke Indonesia.,2009.

LAMPIRAN DATASET

No	Umur	Cholestrol Total	HDL	LDL	Trigliserida	Kelompok
1	44	174	36,5	123,3	71	1
2	60	169	36,9	110,1	110	1
3	23	196	40,3	137,1	93	1
4	41	213	47,4	143,2	112	1
5	40	253	46,5	179,3	136	1
6	48	197	30,5	120,5	79	1
7	49	171	40,1	104,1	134	1
8	40	198	40,5	139,5	90	1
9	49	184	38,4	119,2	133	1
10	59	280	30,5	202,8	128	1
11	60	201	41,6	136,4	115	1
12	29	145	30,5	94,7	99	1
13	46	181	37,7	118,3	125	1
14	66	278	57,2	201,2	98	1
15	48	186	36,2	121,8	140	1
16	66	186	38,1	132,1	79	1
17	61	205	45,2	140,6	96	1
18	28	182	37,7	116,9	137	1
19	34	140	38,8	137,8	128	1
20	75	170	37,4	110,2	112	1
21	59	280	47,6	203,2	146	1
22	45	197	37,4	144,4	76	1
23	49	186	36,1	127,5	112	1
24	37	188	36,4	126,4	126	1
25	39	159	36,7	149,9	142	1
26	53	277	51,7	208,1	86	1
27	72	253	35,1	188,1	149	1
28	54	232	47,9	164,9	96	1
29	71	150	34,5	134,6	138	1
30	32	229	43,5	166,3	96	1
31	54	171	38,1	115,9	85	1
32	52	166	37,8	112,6	33,8	1
33	52	203	38,6	135,6	144	1
34	58	194	40,1	65,9	129	1
35	62	168	35,1	110,1	114	1

36	71	174	35,7	122,1	81	1
37	34	251	47,1	182,9	105	1
38	45	207	46,3	138,9	109	1
39	60	180	42,1	121,1	84	1
40	48	278	30,2	218,8	145	1
41	47	284	52,8	201,4	149	1
42	50	184	33,6	111,8	93	1
43	53	170	39,7	103,5	134	1
44	43	211	35,9	152,3	114	1
45	35	169	37,1	117,1	74	1
46	70	170	47,1	150,7	109	1
47	49	236	42,8	171,4	109	1
48	49	178	42,1	118,9	85	1
49	62	179	40,1	120,9	90	1
50	35	207	40,2	145,4	107	1
51	36	209	44,1	145,3	98	1
52	59	191	36,1	126,2	142	1
53	52	211	42,7	139,1	146	1
54	62	169	35,5	110	104	1
55	59	189	38,2	134,6	81	1
56	46	232	41,6	163,8	133	1
57	32	170	36,5	113,9	98	1
58	61	193	36,7	138,7	88	1
59	57	206	37,6	140	142	1
60	38	300	48,1	225,3	133	1
61	56	178	38,4	110,8	144	1
62	28	276	54,6	202,6	94	1
63	39	276	34,5	293,6	115	1
64	49	182	38,6	122,6	104	1
65	70	216	42,5	144,1	147	1
66	73	200	47,1	133,1	99	1
67	50	200	40	141	95	1
68	68	219	41,6	148,4	145	1
69	76	197	37,6	137,4	110	1
70	56	261	40,5	193,9	133	1
71	30	199	40,2	141,6	86	1
72	23	223	44,6	158,8	98	1
73	63	249	41,5	183,7	119	1

74	57	229	40,8	159,8	142	1
75	66	224	37,2	157,4	147	1
76	30	180	38,4	122,4	96	1
77	73	161	37,8	105,6	88	1
78	62	160	37,2	107	79	1
79	57	240	44,2	171,6	121	1
80	50	184	36,7	125,9	107	1
81	56	148	32,8	102,8	67	1
82	57	179	39,5	120,7	94	1
83	49	258	40,5	192,7	124	1
84	60	182	37,9	125,5	93	1
85	38	246	45,1	141,1	134	1
86	41	238	44,2	168	129	1
87	56	240	45,1	176,3	133	1
88	56	274	39,5	206,1	142	1
89	58	164	39,5	107,1	87	1
90	60	192	37,2	126,8	140	1
91	70	173	36,2	111	132	1
92	52	197	40,6	136,6	94	1
93	60	169	37,8	102,6	143	1
94	53	163	40,1	105,8	83	1
95	54	221	39,6	156,3	126	1
96	57	170	37,6	109,4	115	1
97	43	175	38,6	117	97	1
98	78	199	36,6	145,2	86	1
99	42	207	45,3	133,3	142	1
100	58	225	38,4	159,2	137	1
101	39	179	38,4	116,2	122	1
102	27	165	39,2	106,8	95	1
103	40	211	45,6	167,8	104	1
104	30	176	36,7	121,6	86	1
105	60	281	48,7	206,5	129	1
106	48	210	34,8	213,6	116	1
107	43	206	33,4	107,8	91	1
108	62	147	35,1	94,5	87	1
109	95	192	39,2	123,8	145	1
110	52	241	40,8	177,4	114	1
111	43	181	39,8	124,8	82	1

112	46	255	53,7	110,3	90	1
113	33	227	40,1	164,5	112	1
114	56	162	35,4	112	73	1
115	70	200	36,5	136,1	137	1
116	45	219	41,5	152,1	127	1
117	54	198	38,7	142,1	86	1
118	50	172	38,7	112,5	106	1
119	55	255	37,8	184,6	163	2
120	42	191	40,1	117,1	169	2
121	42	242	42,5	162,5	185	2
122	25	210	40,1	134,1	179	2
123	35	197	39,6	119,6	189	2
124	52	157	30,6	92,6	169	2
125	51	194	40,3	129,5	180	2
126	43	199	35,4	131,4	161	2
127	52	176	40,1	104,5	157	2
128	61	267	41,5	188,1	187	2
129	49	212	40,5	139,7	159	2
130	57	168	40,8	88,8	192	2
131	60	203	45,1	124,9	165	2
132	28	185	42,3	111,5	156	2
133	40	224	45,6	193	186	2
134	52	319	57,2	228	169	2
135	29	213	36,7	145,3	155	2
136	37	239	40	267,9	187	2
137	46	207	38,2	136,8	160	2
138	57	226	42,5	145,5	190	2
139	54	186	36,6	109,8	198	2
140	45	249	44,2	170,6	171	2
141	52	230	36,2	107,6	181	2
142	54	199	42,2	119,2	188	2
143	50	67	42,1	112,1	164	2
144	47	249	42,3	119,7	150	2
145	41	209	37,1	138,3	168	2
146	56	418	45,7	338,1	171	2
147	54	202	38,1	129,5	172	2
148	57	176	39,8	105,6	153	2
149	47	235	45,2	152	189	2

150	55	169	33,9	95,8	198	2
151	65	254	38,6	183,6	159	2
152	49	302	52,4	212	187	2
153	32	237	46,4	159	158	2
154	51	166	34,1	100,5	157	2
155	50	189	35,6	119,2	171	2
156	53	245	45,7	56,5	214	3
157	44	291	46,7	180,5	319	3
158	53	223	39,4	132,2	252	3
159	26	266	40,2	181,2	223	3
160	67	276	51,4	188,2	182	3
161	79	340	31,2	208,9	162	3
162	38	287	51,6	172,2	316	3
163	53	266	45,7	165,5	274	3
164	62	162	36,2	81	224	3
165	45	265	47,6	150,2	241	3
166	57	212	36,9	102,9	361	3
167	44	268	43,6	161,2	316	3
168	85	278	45,3	140,5	261	3
169	69	163	32,5	100	152	3
170	41	173	37,3	113,5	281	3
171	33	259	34,1	166,3	293	3
172	52	287	36,9	193,3	284	3
173	56	189	32,1	91,7	326	3
174	49	324	34	252,2	189	3
175	68	202	37,8	133	156	3
176	55	180	36,7	102,5	204	3
177	65	226	36,1	145,3	223	3
178	42	228	38,6	135,8	268	3
179	49	204	33,6	38,3	461	3
180	51	235	41,6	146,2	236	3
181	56	260	42,2	170,8	235	3
182	50	180	35,1	68,7	381	3
183	29	201	36,1	147,3	226	3
184	58	259	44,2	174,6	201	3
185	53	246	44,9	153,5	238	3
186	56	261	40,6	176,2	221	3
187	50	187	36,1	91,1	295	3

188	64	192	36,1	101,9	260	3
189	53	264	36,9	159,3	319	3
190	51	253	39,3	162,5	256	3
191	48	201	39,5	116,7	224	3
192	50	196	34,5	105,1	282	3
193	41	193	35,7	113,1	221	3
194	91	212	30,4	167,9	155	3
195	69	240	48,3	160,7	155	3
196	80	247	40,5	159,9	233	3
197	81	233	40,7	159,5	164	3
198	46	250	40,8	169	201	3
199	70	175	35,1	99,7	201	3
200	52	514	32,8	318,6	813	3

Daftar Lampiran 1