

**IMPLEMENTASI METODE LEARNING VECTOR
QUANTIZATION UNTUK KLASIFIKASI PENYAKIT DEMAM**

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:
Nurhidayati Desiani
NIM: 135150201111160



PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2018

PENGESAHAN

IMPLEMENTASI METODE LEARNING VECTOR QUANTIZATION UNTUK KLASIFIKASI
PENYAKIT DEMAM

SKRIPSI

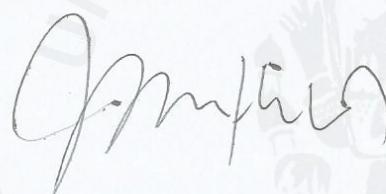
Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :
Nurhidayati Desiani
NIM: 135150201111160

Skripsi ini telah diuji dan dinyatakan lulus pada
12 Oktober 2018

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I



Lailil Muflikhah, S.Kom, M.Sc
NIP: 19741113 200501 2 001

Dosen Pembimbing II



Candra Dewi, S.Kom, M.Sc
NIP: 19771114 200312 2 001

Mengetahui

Ketua Jurusan Teknik Informatika



Tri Astoto Kurniawan, S.T, M.T, Ph.D

NIP: 19710518 200312 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, 19 Juli 2018



Nurhidayati Desiani

NIM: 135150201111160



KATA PENGANTAR

Dengan menyebut nama Allah Yang Maha Pengasih lagi Maha Penyayang. Segala puji bagi Allah yang telah melimpahkan rahmat serta hidayah-Nya sehingga skripsi yang berjudul “Implementasi Metode *Learning Vector Quantization* Untuk Klasifikasi Penyakit Demam” ini dapat terselesaikan.

Melalui kesempatan ini, penulis ingin menyampaikan rasa hormat dan terimakasih yang sebesar-besarnya kepada semua pihak yang telah berkontribusi dan memberikan bantuan lahir batin selama penyelesaian tugas akhir ini. Oleh karena itu, penulis ingin menyampaikan rasa hormat dan terimakasih kepada:

1. Ibu Lailil Muflikhah, S.Kom., M.Sc. dan Ibu Candra Dewi, S.Kom., M.Sc. selaku dosen pembimbing skripsi yang telah sabar dalam membantu dan mengarahkan penulis sehingga skripsi ini bisa penulis selesaikan.
2. Bapak Agus Wahyu Widodo, S.T., M.Cs. selaku Ketua Program Studi Teknik Informatika.
3. Bapak Tri Astoto Kurniawan, S.T., M.T., Ph.D. selaku Ketua Jurusan Teknik Informatika.
4. Bapak Wayan Firdaus Mahmudy, S.Si., M. T., Ph.D. selaku Dekan Fakultas Ilmu Komputer.
5. Bapak Marji, Drs., M. T. selaku dosen Penasehat Akademik yang selalu memberikan nasehat kepada penulis selama menempuh masa studi.
6. Seluruh Dosen yang telah memberikan ilmu kepada penulis dari awal sampai akhir masa studi.
7. Ayahanda dan Ibunda, kakak Nur Oktia N.S., adik Nurul Hikmawati serta keluarga besar atas segala doa, nasehat, perhatian, bantuan lahir maupun batin, serta motivasi sampai skripsi ini dapat terselesaikan.
8. Seluruh Civitas Akademika Informatika Universitas Brawijaya yang telah banyak memberikan dukungan dan bantuan selama masa studi dan penyelesaian skripsi ini.
9. Sahabat-sahabat penulis Shima Fanissa, Vidya Capristyan P., Vinny Riswandha R., Nurul Dyah M., Toni Faqih P., Yuda Agung P., Brilian A.R., Melania Primasta, Keluarga Cemara dan semua pihak yang tidak dapat disebutkan satu per satu oleh penulis yang terlibat secara langsung maupun tidak langsung dalam penyelesaian skripsi ini.



Penulis menyadari bahwa dalam penyusunan skripsi ini masih banyak kekurangan, sehingga kritik dan saran yang membangun sangat penulis harapkan. Semoga skripsi ini bermanfaat bagi pembaca.

Malang, 19 Juli 2018

Penulis

punyaiedha@gmail.com



ABSTRAK

Demam adalah gejala penyakit yang pernah dialami oleh hampir semua orang. Demam adalah gejala awal dari berbagai macam penyakit. Dari berbagai penyakit tersebut, demam tifoid, demam malaria dan demam berdarah merupakan penyakit yang memiliki gejala awal yang mirip sehingga sulit untuk dibedakan. Masih banyak dokter-dokter muda dan tenaga medis sulit dalam mendeteksi lebih awal ketiga penyakit ini karena masih memiliki pengalaman yang kurang. Kemiripan gejala dari masing-masing penyakit sering menimbulkan kesulitan dalam mendapatkan anamnese (diagnosa sementara) sehingga pasien mendapatkan penanganan awal yang kurang tepat dan semakin memperburuk kondisi pasien. Untuk mengatasi hal tersebut dibutuhkan suatu sistem yang dapat mempermudah dalam mengidentifikasi penyakit berdasarkan gejala-gejala yang dirasakan oleh pasien. Pada penelitian ini menggunakan metode *Learning Vector Quantization* yang merupakan metode klasifikasi yang digunakan untuk menentukan kelas dari data uji dengan menghitung jarak terdekat data uji terhadap bobot setiap kelas. Parameter yang digunakan adalah 15 parameter gejala dari penyakit demam. Sistem bekerja dengan tahapan pelatihan dan pengujian yang akan menghasilkan kelas berupa kelas demam tifoid, demam malaria dan demam berdarah. Hasil rata-rata akurasi terbaik yaitu 100% dengan menggunakan perbandingan data uji dan data latih sebesar 10:90, laju pembelajaran 0,1, konstanta pengurang laju pembelajaran 0,1, laju pembelajaran minimum 10^{-5} , dan jumlah iterasi maksimum 10.

Kata kunci: Klasifikasi, Malaria, DBD, Tifoid, Demam, *Learning Vector Quantization*



ABSTRACT

Fever is a symptom of disease that has been experienced by almost people in the world. Fever is the initial indication from many kinds of diseases. From the various of diseases, typhoid fever, malarial fever and dengue fever are the diseases with similar initial indication. that similars make these three of diseases are difficult to distinguish. Many young doctors and medical personnel are difficult to detect early these three of deseases because their less experience. The Similar symptom of each disease often cause difficulty in obtaining anamnese (temporary diagnosis) so that patients get less precise of initial handling and make the condition of the patient is getting worse . To resolve that problem, required a system that can facilitate in identifying diseases based of the symptoms that felt by the patient. This study use Learning Vector Quantization method, that is a classification method used to determine the class of test data by calculating the closest distance of test data on the weight of each class. The parameters used are 15 parameters of symptoms from febrile illness. The system works with the training and testing phases that will result in classes, that are typhoid fever class, malarial fever class and dengue fever class. The result of the best average accuracy is 100% by using comparison of test data and training data at 10:90, the learning rate is 0,1, the learning rate constant is 0,1, the minimum learning rate 10^{-5} and the maximum iteration is amount 10.

Key word: Classification, Malaria, DBD, Tifoid, Fever, *Learning Vector Quantization*



DAFTAR ISI

PENGESAHANii
PERNYATAAN ORISINALITASiii
KATA PENGANTAR.....	.iv
ABSTRAK.....	.vi
ABSTRACT.....	.vii
DAFTAR ISI.....	.viii
DAFTAR TABEL.....	.xi
DAFTAR GAMBAR.....	.xii
BAB 1 PENDAHULUAN.....	1
1.1 Latar belakang.....	1
1.2 Rumusan masalah	2
1.3 Tujuan	2
1.4 Manfaat.....	2
1.4.1 Bagi Peneliti.....	2
1.4.2 Bagi Tim Medis	3
1.5 Batasan masalah dan ruang lingkup penelitian	3
1.6 Sistematika pembahasan	3
BAB 2 LANDASAN KEPUSTAKAAN	5
2.1 Kajian Pustaka	5
2.2 Demam.....	8
2.2.1 Demam Berdarah (Dengue Haemoragic Fever)	8
2.2.2 Demam Tifoid	9
2.2.3 Demam Malaria.....	9
2.3 <i>Learning Vector Quantization (LVQ)</i>	9
2.4 Evaluasi	10
BAB 3 METODOLOGI	11
3.1 Studi literatur	11
3.2 Pengumpulan Data	12
3.3 Analisis Kebutuhan	12
3.4 Perancangan Sistem.....	12



3.5 Implementasi	14
3.6 Pengujian dan Analisis	14
3.7 Kesimpulan.....	14
BAB 4 PERANCANGAN.....	15
4.1 Deskripsi Umum Sistem	15
4.2 Proses Pelatihan dengan LVQ	17
4.3 Proses Pengujian dengan LVQ	19
4.4 Proses Perhitungan Jarak.....	19
4.5 <i>Preprocessing</i> Data	20
4.6 Perhitungan Manual	22
4.6.1 Manualisasi Pelatihan menggunakan LVQ.....	22
4.6.2 Manualisasi Pengujian menggunakan LVQ	35
4.6.3 Perhitungan Akurasi.....	38
4.7 Perancangan Antarmuka	39
4.7.1 Antarmuka Halaman Beranda	39
4.7.2 Antarmuka Halaman Pelatihan	40
4.7.3 Antarmuka Halaman Pengujian.....	40
4.7.4 Antarmuka Halaman Diagnosis	41
BAB 5 IMPLEMENTASI	42
5.1 Lingkungan Implementasi.....	42
5.1.1 Lingkungan Perangkat Keras	42
5.1.2 Lingkungan Perangkat Lunak	42
5.2 Implementasi Program	43
5.2.1 Penentuan Bobot Awal	43
5.2.2 Pelatihan dengan LVQ.....	44
5.2.3 Pengujian dengan LVQ	46
5.3 Implementasi Antarmuka	49
5.3.1 Halaman Beranda.....	49
5.3.2 Halaman Pelatihan.....	50
5.3.3 Halaman Pengujian.....	50
5.3.4 Halaman Diagnosis.....	51
BAB 6 PENGUJIAN DAN ANALISIS.....	52



6.1 Hasil dan Analisis Uji Coba untuk Mengetahui Pengaruh Nilai Laju Pembelajaran terhadap Hasil Akurasi.....	52
6.2 Hasil dan Analisis Uji Coba untuk Mengetahui Pengaruh Nilai Konstanta Pengurang Laju Pembelajaran terhadap Hasil Akurasi	53
6.3 Hasil dan Analisis Uji Coba untuk Mengetahui Pengaruh Nilai Laju Pembelajaran Minimum terhadap Hasil Akurasi.....	55
6.4 Hasil dan Analisis Uji Coba untuk Mengetahui Pengaruh Jumlah Iterasi Maksimum terhadap Hasil Akurasi.....	56
6.5 Hasil dan Analisis Uji Coba untuk Mengetahui Pengaruh Perbandingan Jumlah Data Latih dan Data Uji terhadap Hasil Akurasi	58
6.6 Hasil dan Analisis Pengujian 1 untuk Mengetahui Akurasi Berdasarkan Data Uji yang Dimasukkan Pengguna	59
6.7 Hasil dan Analisis Pengujian 2 untuk Mengetahui Hasil Klasifikasi dari Gejala-Gejala yang dipilih Pengguna	61
BAB 7 PENUTUP	62
7.1 Kesimpulan.....	62
7.2 Saran	62
DAFTAR PUSTAKA.....	63
LAMPIRAN A DATA HASIL PEMERIKSAAN PASIEN	64
LAMPIRAN B DATA GEJALA PENYAKIT	71
LAMPIRAN C DATA NILAI BOBOT GEJALA PENYAKIT DBD, MALARIA DAN TIFOID	72



DAFTAR TABEL

Tabel 2.1 Kajian Pustaka	6
Tabel 4.1 Bobot untuk Gejala Demam	21
Tabel 4.2 Data Latih	22
Tabel 4.3 Bobot Awal	23
Tabel 4.4 Data Uji	35
Tabel 4.5 Hasil Klasifikasi Perhitungan Manual	38
Tabel 5.1 Lingkungan Perangkat Keras	42
Tabel 5.2 Lingkungan Perangkat Lunak.....	42
Tabel 6.1 Hasil Uji Coba Pengaruh Nilai Laju Pembelajaran Terhadap Akurasi....	52
Tabel 6.2 Hasil Uji Coba Pengaruh Nilai Konstanta Pengurang Laju PembelajaranTerhadap Akurasi	54
Tabel 6.3 Hasil Uji Coba Pengaruh Nilai Laju Pembelajaran Minimum Terhadap Akurasi.....	55
Tabel 6.4 Hasil Uji Coba Pengaruh Jumlah Iterasi Maksimum Terhadap Hasil Akurasi.....	57
Tabel 6.5 Hasil Uji Coba Pengaruh Perbandingan Jumlah Data Latih dan Data Uji Terhadap Akurasi	58
Tabel 6.6 Data Pengujian	60
Tabel 6.7 Jarak Pada Pengujian 2	61



DAFTAR GAMBAR

Gambar 3.1 Diagram Blok Metodologi Penelitian	11
Gambar 3.2 Arsitektur Sistem.....	13
Gambar 4.1 Bagan Perancangan	15
Gambar 4.2 Diagram Alir Proses Kerja Sistem Secara Umum	16
Gambar 4.3 Diagram Alir Proses Pelatihan LVQ	18
Gambar 4.4 Diagram Alir Proses Pengujian LVQ.....	19
Gambar 4.5 Diagram Alir Proses Perhitungan Jarak	20
Gambar 4.6 Rancangan Antarmuka Halaman Beranda.....	39
Gambar 4.7 Rancangan Antarmuka Halaman Pelatihan.....	40
Gambar 4.8 Rancangan Antarmuka Halaman Pengujian.....	41
Gambar 4.9 Rancangan Antarmuka Halaman Diagnosis.....	41
Gambar 5.1 Implementasi Antarmuka Beranda.....	49
Gambar 5.2 Implementasi Antarmuka Pelatihan LVQ.....	50
Gambar 5.3 Implementasi Antarmuka Pengujian.....	50
Gambar 5.4 Implementasi Antarmuka Diagnosis.....	51
Gambar 6.1 Grafik Uji Coba Pengaruh Nilai Laju Pembelajaran Terhadap Akurasi	53
Gambar 6.2 Grafik Uji Coba Pengaruh Nilai Konstanta Pengurang Laju Pembelajaran Terhadap Akurasi	54
Gambar 6.3 Grafik Uji Coba Pengaruh Laju Pembelajaran Minimum Terhadap Akurasi.....	56
Gambar 6.4 Grafik Uji Coba Pengaruh Nilai Iterasi Terhadap Akurasi.....	57
Gambar 6.5 Grafik Hasil Uji Coba Pengaruh Perbandingan Jumlah Data Latih dan Data Uji Terhadap Akurasi	59



BAB 1 PENDAHULUAN

1.1 Latar belakang

Demam merupakan salah satu jenis penyakit yang dapat mengganggu produktivitas setiap orang bahkan dapat menyebabkan kematian dan masih menjadi permasalahan kesehatan Indonesia. Demam yang harus diwaspadai Beberapa jenis demam yang perlu diwaspadai antara lain Demam Berdarah Dengue, Demam Malaria dan Demam Tifoid.

Jika mengalami demam, suhu tubuh bisa mengalami peningkatan hingga di atas 36 derajat celcius. Hal ini bisa disebabkan oleh faktor infeksi maupun faktor non infeksi. Demam merupakan respon dari tubuh untuk melawan zat-zat yang berbahaya yang mengganggu. Penyakit-penyakit dengan gejala seperti meningitis, infeksi saluran pernafasan dan infeksi saluran kandung kemih selalu diawali dengan gejala demam. Penyakit demam tersebut merupakan demam tifoid, malaria, dan demam berdarah (Wafiyah, 2017).

Demam tifoid masih menjadi salah satu penyakit diantara sepuluh penyakit yang menjadi masalah kesehatan untuk beberapa negara maju. Berdasarkan Daftar Tabulasi Data (DTD), jumlah data pasien yang mengalami demam tifoid yaitu 55.098 pasien dan memiliki persentase angka kematian 2.06% (Wafiyah, 2017).

Selain demam tifoid, penyakit demam berdarah juga merupakan penyakit yang perlu secara khusus diperhatikan, karena selama 30 tahun terakhir total pasien yang mengalami demam berdarah terus bertambah secara signifikan dengan total kasus sejumlah 1.395 (Depkes RI, 2008). Anak-anak maupun orang dewasa dapat diserang oleh penyakit demam berdarah. Indikator utama dari demam berdarah yaitu demam yang semakin memburuk pada hari kedua dan seterusnya serta rasa nyeri pada sendi dan otot.

Penyakit malaria merupakan jenis penyakit yang sangat berbahaya yang disebabkan oleh parasit. Jumlah kematian pasien mencapai 30.000 setiap tahun dan jumlah pasien sebanyak 15 juta setiap tahun.

Demam tifoid, malaria dan demam berdarah memiliki beberapa indikator untuk diagnosa klinis antara lain, demam, gangguan pencernaan, nyeri otot, kondisi lidah serta pembesaran pada hati dan limpa. Gejala yang mirip dari masing-masing penyakit sering menimbulkan kesulitan dalam mendapatkan anamnese (diagnosa sementara) sehingga penanganan awal yang kurang tepat dapat memperburuk kondisi pasien (Hariman, 2014).

Kemajuan teknologi telah membantu menyelesaikan permasalahan diberbagai bidang, termasuk dalam bidang kesehatan. Gejala-gejala dari demam tifoid, malaria, dan demam berdarah dapat diklasifikasikan berdasarkan jarak terdekat antara bobot dengan data baru. Proses klasifikasi menggunakan algoritma dalam komputer yang melalui proses pelatihan dan pengujian sebagai



acuan untuk menentukan hasil diagnosa pada data baru sehingga dapat mengurangi kesalahan dalam melakuakan diagnosa.

Salah satu penelitian yang membahas klasifikasi penyakit demam tifoid, demam malaria dan demam berdarah dilakukan oleh Fakihatin Wafiyah (2017) menggunakan algoritma modified k-nearest neighbor. Data yang digunakan dalam proses klasifikasi yaitu sejumlah 133 data yang terdiri dari 100 data latih dan 33 data uji. Hasil dari penelitian dapat mengklasifikasikan penyakit demam berdarah, tifoid, dan malaria dengan akurasi sebesar 97.21%.

Penelitian selanjutnya dilakukan oleh Nurul Islamiah (2016) yaitu mengklasifikasi penyakit mulut dan gigi menggunakan metode *Learning Vector Quantization*. Proses klasifikasi dengan input gejala-gejala penyakit gigi dan mulut dan output hasil yaitu klasifikasi penyakit gigi dan mulut. Hasil akurasi tertinggi dari penelitian ini adalah 100%.

Berdasarkan beberapa penelitian sebelumnya, penulis tertatik untuk melakukan penelitian dengan judul “ IMPLEMENTASI METODE LEARNING VECTOR QUANTIZATION UNTUK KLASIFIKASI PENYAKIT DEMAM”. Tujuan dari penelitian ini yaitu untuk memudahkan para ahli medis dalam memperoleh anamnese (diagnosa sementra) dengan cara mengklasifikasikan gejala-gejala yang diderita oleh pasien dan memberikan output salah satu penyakit demam berdarah, tifoid atau malaria.

1.2 Rumusan masalah

1. Bagaimana implementasi metode *Learning Vector Quantization* untuk klasifikasi penyakit demam?
2. Berapa tingkat akurasi dari hasil implementasi metode *Learning Vector Quantization* untuk klasifikasi penyakit demam?

1.3 Tujuan

Tujuan dari penelitian ini adalah sebagai berikut :

- a. Untuk menerapkan metode Learning Vector Quantization dalam klasifikasi demam.
- b. Untuk mengukur nilai kualitas dan akurasi yang dihasilkan metode *Learning Vector Quantization* dalam klasifikasi demam.

1.4 Manfaat

1.4.1 Bagi Peneliti

1. Lebih memahami kelebihan metode *Learning Vector Quantization* dalam pemecahan masalah.
2. Dapat menambah pengetahuan dan pengalaman.



1.4.2 Bagi Tim Medis

1. Membantu dalam melakukan diagnosa awal terhadap penyakit demam yaitu demam tifoid, malaria, dan demam berdarah sehingga pasien bisa mendapatkan penanganan yang tepat.

1.5 Batasan masalah dan ruang lingkup penelitian

Batasan masalah dalam penelitian ini adalah :

- a. Data yang digunakan adalah data sekunder berasal dari penelitian sebelumnya yang dilakukan oleh Wafiyah (2017).
- b. Data input yaitu bobot dari 15 gejala penyakit demam yang sesuai dengan kriteria bobot yang telah diberikan oleh pihak rumah sakit pada penelitian sebelumnya yang dilakukan oleh Wafiyah (2017).
- c. Metode yang digunakan dalam penelitian ini adalah *Learning Vector Quantization*.
- d. Output sistem merupakan hasil klasifikasi demam yaitu demam tifoid, malaria dan demam berdarah.

1.6 Sistematika pembahasan

Sistematika penulisan dalam skripsi ini sebagai berikut :

BAB 1 PENDAHULUAN

Berisi Latar Belakang, Rumusan Masalah, Tujuan Penelitian, Manfaat Penelitian, Batasan Penelitian/ Ruang Lingkup Penelitian, dan Sistematika Pembahasan/ Laporan.

BAB 2 LANDASAN KEPUSTAKAAN

Berisi tentang teori - teori yang mendasari dan mendukung penulisan klasifikasi penyakit demam menggunakan metode *Learning Vector Quantization*.

BAB 3 METODOLOGI

Bab ini menguraikan metodologi yang digunakan dalam menganalisis, merancang, dan mengimplementasikan serta menguji sistem klasifikasi penyakit demam menggunakan metode *Learning Vector Quantization*.

BAB 4 PERANCANGAN

Bab ini membahas tentang perancangan sistem untuk klasifikasi penyakit demam dengan metode *Learning Vector Quantization*.

BAB 5 IMPLEMENTASI

Bab ini membahas tentang implementasi sistem untuk klasifikasi penyakit demam dengan menggunakan metode *Learning Vector Quantization*.

BAB 6 PENGUJIAN DAN ANALISIS

Bab ini membahas proses pengujian terhadap sistem dan melakukan analisa terhadap hasil pengujian.



BAB 7 PENUTUP

Bab ini berisi kesimpulan dari hasil penelitian serta saran untuk pengembangan lebih lanjut.



BAB 2 LANDASAN KEPUSTAKAAN

Bab ini memuat kajian pustaka dan dasar teori. Kajian pustaka dan dasar teori yang digunakan adalah yang berkaitan dengan tema, masalah atau pertanyaan penelitian. Kajian pustaka berisi informasi mengenai penelitian-penelitian sebelumnya yang berhubungan dengan topik. Sedangkan dasar teori membahas berbagai sumber pustaka yang terkait dengan metode dan teori yang digunakan dalam menyusun skripsi ini. Teori yang digunakan yaitu teori tentang Demam Berdarah, Demam Tifoid, Demam Malaria, Jaringan Syaraf Tiruan, serta *Learning Vector Quantization* (LVQ).

2.1 Kajian Pustaka

Nurul Islamiah (2016) melakukan klasifikasi penyakit gigi dan mulut menggunakan metode Learning Vector Quantization pada penelitian sebelumnya. Gejala-gejala penyakit gigi dan mulut merupakan masukan yang akan di proses menggunakan algoritma LVQ. Sebelumnya dilakukan pelatihan terhadap data latih. Data latih merupakan data penyakit yang sudah diklasifikasi ke dalam kelas atau penyakit masing-masing. Hasil dari pelatihan tersebut akan digunakan untuk klasifikasi inputan gejala baru. Hasil yang didapatkan berupa klasifikasi penyakit dari pasien. Rata-rata akurasi tertinggi yang didapatkan dalam penelitian tersebut adalah 100%.

Penelitian menggunakan algoritma Fuzzy K-Nearest Neighbor (FKNN) oleh Dwi Citra Annisa (2016) untuk melakukan klasifikasi penyakit demam berdarah, demam tifoid, dan demam malaria berdasarkan nilai keanggotaan terbesar pada tiap kelas penyakit sebagai penyakit hasil prediksi. Pada penelitian ini mendapatkan akurasi tertinggi sebesar 94% pada pengujian pengaruh komposisi data latih.

Wafiyah (2017) melakukan klasifikasi penyakit demam berdarah, tifoid, dan demam malaria. Pada penelitian ini menggunakan jumlah parameter gejala yang sama dengan penelitian sebelumnya yaitu 15 gejala penyakit yang sudah memiliki bobot masing-masing. Penelitian ini menggunakan Algoritma Modified K-Nearest Neighbor (MKNN) dan menghasilkan rata-rata akurasi tertinggi 97,21%.

Penelitian menggunakan algoritma LVQ oleh Eliyen, Tolle, & Muslim (2017) untuk melakukan penilaian ujian *Objective Structure Clinical Examinations* (OSCE) pada mahasiswa kedokteran yang merupakan simulasi pemeriksaan klinis terhadap pasien *virtual*. Mahasiswa akan diberikan keluhan-keluhan pasien, dan mahasiswa akan melakukan pemeriksaan, mendiagnosa penyakit, dan menentukan perawatan yang akan diberikan kepada pasien. Penilaian dilakukan dengan mengklasifikasi jawaban dari mahasiswa menggunakan algoritma LVQ yang akan disimpan oleh sistem untuk setiap kategori pemeriksaan. Penelitian ini menggunakan data latih yang berisi data diri pasien, rekam medis pasien,



diagnosis penyakit dan rencana perawatan. Hasil akurasi yang diperoleh adalah 98,8%.

Tabel 2.1 Kajian Pustaka

No	Judul	Objek	Metode	Hasil
1.	“Klasifikasi Penyakit Gigi dan Mulut Menggunakan Metode <i>Learning Vector Quantization</i> (LVQ)” (Islamiah, 2016)	Objek : Penyakit Gigi dan Mulut <i>Input</i> : Gejala penyakit gigi dan mulut	Metode : <i>Learning Vector Quantization</i>	Hasil : akurasi tertinggi yang diperoleh pada penelitian ini yaitu 100%.
2.	“Pendeteksi Jenis Attention Deficit Hyperactivity Disorder (ADHD) Pada Anak Usia Dini Menggunakan Metode <i>Learning Vector Quantization</i> (LVQ)” (Arifien, 2016)	Objek : Jenis ADHD pada anak <i>Input</i> : gejala indikasi ADHD	Metode : <i>Learning Vector Quantization</i>	Hasil : jenis ADHD dengan akurasi 70%.
3.	“Sistem Pakar Diagnosa Awal Penyakit DBD, Malaria dan Tifoid menggunakan Metode Fuzzy	15 Gejala Penyakit	Metode : Fuzzy K-Nearest Neighbor(FKNN)	Hasil : Diagnosa Awal Penyakit DBD, Malaria, dan Tifoid mendapatkan akurasi tertinggi 94%

	K-Nearest Neighbor (FKNN)" (Annisa, 2016)			
4.	"Implementasi Algoritma Modified K-Nearest Neighbor (MKNN) untuk Klasifikasi Penyakit Demam" (Wafiyah, 2017)	Objek : Penyakit demam malaria, tifoid, dan demam berdarah <i>Input :</i> Gejala penyakit demam	Metode : Modified K-Nearest Neighbor (MKNN)	Hasil : Diagnosa Awal Penyakit DBD, Malaria, dan Tifoid dengan akurasi tertinggi 97.21%
5.	"Sistem Pakar Diagnosis Penyakit Demam: DBD, Malaria dan Tifoid Menggunakan Metode <i>K-Nearest Neighbor – Certainty Factor</i> " (Shofia, Putri, & Arwan, 2017)	Objek : Penyakit demam malaria, tifoid, dan demam berdarah <i>Input :</i> Gejala penyakit demam	Metode : K-Nearest Neighbor – Certainty Factor	Hasil : Diagnosa Awal Penyakit DBD, Malaria, dan Tifoid dengan akurasi tertinggi 97.22%
6.	"Learning Vector Quantization untuk Klasifikasi Penilaian Pada Virtual Patient Case" (Eliyen, Tolle, &	Objek : Nilai ujian OSCE pada <i>Virtual Patient Case</i> <i>Input :</i> Jawaban mahasiswa, Kemampuan anamnesa,pemeriksaan fisik, komunikasi dan	Metode : Learning Vector Quantization	Hasil : Nilai ujian OSCE mahasiswa kedokteran dengan akurasi 98,8%



	Muslim, 2017)	perilaku professional		
7.	“ Implementasi Metode <i>Learning Vector Quantization</i> (LVQ) untuk Klasifikasi Penyakit Demam” (Usulan)	Objek : Penyakit Demam Malaria, Demam Tifoid, dan Demam Berdarah. <i>Input :</i> Gejala-gejala penyakit demam.	Metode : <i>Learning Vector Quantization</i>	Hasil :

2.2 Demam

Demam adalah gangguan kesehatan yang mengganggu aktivitas setiap orang bahkan dapat menyebabkan kematian. Jenis demam yang perlu diwaspadai yaitu Demam Berdarah Dengue (DBD), *Tifoid*, Malaria, *Viral*, *Chicken Guinea*, Meningitis, Infeksi Saluran Kemih dan *HIV* (Shofia, Putri, & Arwan, 2017).

Untuk mengetahui seseorang terserang penyakit demam jenis apa harus diketahui dari gejala yang dialami oleh orang tersebut. Padahal gejala pada penyakit Demam sangat mirip dan sulit dibedakan sehingga seringkali terjadi kesalahan dalam mendiagnosis karena keterbatasan pengalaman sehingga kecepatan mendiagnosis penyakit sangat terbatas dan terkadang kurang akurat. Masih banyak dokter internship dan tenaga medis yang sulit dalam mendiagnosis ketiga penyakit ini. Dokter internship adalah dokter yang sedang melakukan program magang dan baru menyelesaikan masa pendidikan profesi (Wafiyah, 2017).

Demam merupakan kondisi dimana suhu badan melebihi 37 derajat Celcius. Demam juga merupakan ciri dari sel antibodi manusia sedang melawan virus atau bakteri. Terdapat 8 jenis demam yang perlu di waspadai antara lain Demam Berdarah Dengue, Tifoid, Malaria, Chicken Guinea, Viral, Meningitis, Infeksi saluran kemih dan HIV. Dari 8 jenis demam tersebut 3 diantaranya memiliki gejala yang mirip yaitu Demam Berdarah Dengue, Malaria dan Tifoid (Shofia, Putri, & Arwan, 2017).

2.2.1 Demam Berdarah (Dengue Haemoragic Fever)

Demam Berdarah Dengue (DBD) adalah salah satu jenis penyakit yang bersifat menular dan disebabkan oleh virus Dengue. Virus Dengue tersebut berasal dari gigitan nyamuk Aedes aegypti. Gejala awal dari demam berdarah yaitu peningkatan suhu badan secara signifikan dalam jangka waktu 7 hari. Selain itu, kondisi tubuh yang mulai lemah atau lesuh, gelisah, rasa nyeri pada



ulu hati, dan terjadi pendarahan pada kulit berupa bintik pendarahan (*petechia*) merupakan gejala awal demam berdarah. Penderita penyakit demam berdarah juga dapat mengalami pendarahan dari hidung (mimisan), penurunan kesadaran, berak darah dan muntah darah (Wafiyah, 2017). Demam berdarah disertai minimal dengan dua gejala yaitu:

- Nyeri Kepala
- Nyeri Retro-orbital
- Nyeri Otot
- Nyeri Sendi/Tulang
- Ruam Kulit makulopapular
- Manifestasi Pendarahan
- Tidak ada tanda Perembesan Plasma

2.2.2 Demam Tifoid

Demam Tifoid merupakan salah satu jenis penyakit yang disebabkan oleh *Salmonella typhi* dengan indikator utama yaitu demam dalam jangka waktu yang lama serta mengalami gejala sakit kepala, badan lemah anoreksia, bradikardi relative dan splenomegaly (Gillepsie, 2009). Demam Tifoid dapat dijumpai merata diseluruh dunia. Pada tahap tertentu, bisa menyebabkan pendarahan pada pasien. Indikator lain yang mungkin terjadi adalah demam tanpa disertai keringat, gangguan berfikir, parotitis dan pendengaran menurun.

Gejala klinis sangat bervariasi dari ringan sampai berat, dari yang tidak terdiagnosis sampai gambaran penyakit yang khas dengan komplikasi hingga menimbulkan kematian. Sering ditemukan keluhan dengan gejala yang mirip penyakit infeksi akut pada minggu pertam, yaitu: demam, nyeri kepala, pusing, nyeri otot, anoreksi, mual, muntah, obstipasi atau diare, perasaan tidak enak di perut, batuk dan epitaksis (Hadi Saputro, 1991).

2.2.3 Demam Malaria

Penyakit malaria adalah penyakit yang disebabkan oleh parasit *Plasmodium vivax* dan *Plasmodium falciparum*. Penyakit ini ditularkan melalui gigitan nyamuk *Anophlese*. Faktor-faktor yang menyebabkan penyebaran penyakit ini yaitu, lingkungan tempat tinggal, pola hidup dan lain sebagainya. Penyakit ini mudah menyerang penduduk yang berada di daerah persawahan, perkebunan, kehutanan dan pesisir pantai. Beberapa gejala malaria yaitu, demam, gangguan pencernaan, gangguan kesadaran, pembengkakan pada limpa, anemia, pendarahan di hidung (Wafiyah, 2017). Selain itu juga bisa mengalami gejala berkeringat berlebih, sakit kepala, nyeri pada persendian dan mual (Departemen Kesehatan RI, 2007).



2.3 Learning Vector Quantization (LVQ)

Metode klasifikasi *Learning Vector Quantization* (LVQ) dikembangkan pada tahun 1989 oleh Teuvo Kohonen. LVQ ini merupakan *single layer net* dimana setiap lapisan *input* terhubung secara langsung dengan lapisan *output*. Keduanya dihubungkan dengan suatu bobot/weight. Struktur jaringan pada LVQ terdiri dari x_i adalah *input*, w_{ij} merupakan bobot dan y_i sebagai *output* (Arifianto, 2014).

LVQ merepresentasikan *output* unit sebagai *particular class* atau kategori. Bobot vektor dalam sebuah unit merupakan *reference vector* ke suatu kelas. Secara umum algoritma dari LVQ sebagai berikut (Fauset, 1994) :

Step 0. Inisialisasi *input vector* (x), bobot awal, *learning rate*(α).

Step 1. *While (stop==false)* kerjakan langkah 2-6

Step 2. Untuk setiap vektor *input* data latih (x) kerjakan langkah 3-4

Step 3. Hitung J minimum dengan *Euclidean* $\| x - w_j \|$

Step 4. *Update* w_j dengan ketentuan :

Jika $T = C_j$ maka

$$w_j(\text{new}) = w_j(\text{old}) + \alpha[x - w_j(\text{old})] \quad (2.1)$$

Jika $T \neq C_j$ maka

$$w_j(\text{new}) = w_j(\text{old}) - \alpha[x - w_j(\text{old})] \quad (2.2)$$

Step 5. Kurangi *learning rate*.

Step 6. Hentikan pada iterasi tertentu atau *learning rate* mendekati *learning rate* minimum.

Keterangan dari algoritma yaitu :

x = vektor *input* data latih (x_1, x_2, \dots, x_n)

T = kelas dari *training vector*

w_j = bobot vektor untuk *output* ke- j

C_j = kelas atau kategori yang dihasilkan dari *output* ke- j

$\| x - w_j \|$ = jarak *Euclidean* antara *vector input* dan bobot vektor ke- j

2.4 Evaluasi

Pada tahap evaluasi, peneliti akan menghitung tingkat akurasi dari metode *Learning Vector Quantization* dalam mengidentifikasi penyakit dengan gejala batuk pada anak. Tingkat akurasi ini dihitung berdasarkan perbandingan dari jumlah kasus penyakit dengan gejala batuk yang berhasil diidentifikasi sesuai target dengan jumlah keseluruhan kasus yang diuji dikali 100%. Pengukuran akurasi dapat dituliskan dengan persamaan 2.3.

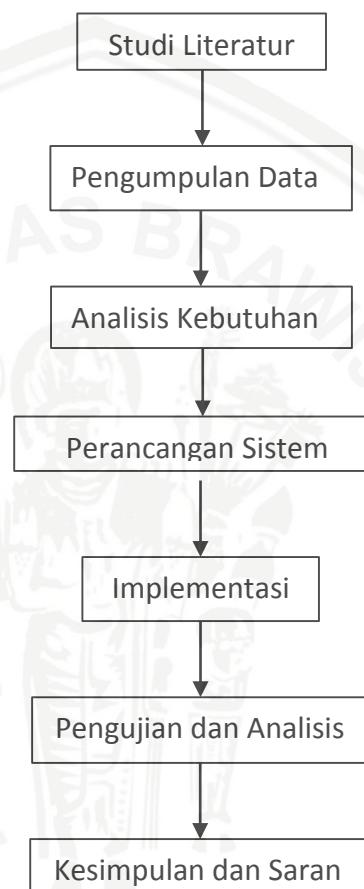


$$\text{Akurasi \%} = \frac{\text{jumlah data uji yang benar}}{\text{jumlah keseluruhan data yang diuji}} \times 100\% \quad (2.3)$$



BAB 3 METODOLOGI

Bab ini membahas tentang metodologi yang akan digunakan dalam penelitian ini. Bab ini juga akan menjelaskan rancangan sistem yang akan dikembangkan. Metodologi penelitian yang dilakukan dalam penelitian ini yaitu studi literatur, pengumpulan data, analisis kebutuhan, perancangan sistem, implementasi, pengujian dan analisis, serta kesimpulan dan saran. Tahapan-tahapan dalam penelitian tersebut dapat ditunjukkan pada diagram blok metodologi penelitian pada Gambar 3.1.



Gambar 3.1 Diagram Blok Metodologi Penelitian

3.1 Studi literatur

Studi literatur bertujuan untuk mempelajari bagaimana literatur dari beberapa bidang ilmu yang berhubungan dengan pembuatan sistem klasifikasi penyakit demam dengan menggunakan metode LVQ. Teori-teori yang dipelajari diantaranya :

- a. Penyakit demam
- b. Penyakit-penyakit dengan gejala demam yang dibahas dalam penelitian ini.



- c. Jaringan Syaraf Tiruan.
- d. *Learning Vector Quantization* meliputi arsitektur dan langkah kerja LVQ.

Sumber atau referensi yang digunakan sebagai penunjang dan pendukung penelitian ini yaitu buku, paper, jurnal, laporan penelitian, dan dari internet.

3.2 Pengumpulan Data

Data yang digunakan adalah data sekunder. Data yang digunakan merupakan data yang telah didapatkan pada penelitian sebelumnya. Data tersebut diperoleh dari Rumah Sakit Umum Daerah Selasih Pangkalan Kerinci Riau. Penelitian ini menggunakan 15 gejala penyakit beserta bobot yang telah diberikan oleh pakar RSUD Selasih dr. Anda Citra Utama, SpPD untuk masing-masing gejala dalam beberapa tingkatan dengan total data sebanyak 133 data pasien penderita penyakit demam berdarah, tifoid, dan malaria.

3.3 Analisis Kebutuhan

Perangkat keras maupun perangkat lunak dibutuhkan untuk lingkungan pengujian. Spesifikasi kebutuhan perangkat keras yang digunakan yaitu:

- a. Processor Intel Core i7-5500U 2.40GHz
- b. RAM 4 GB
- c. HDD 1000 GB
- d. Monitor 14 inch

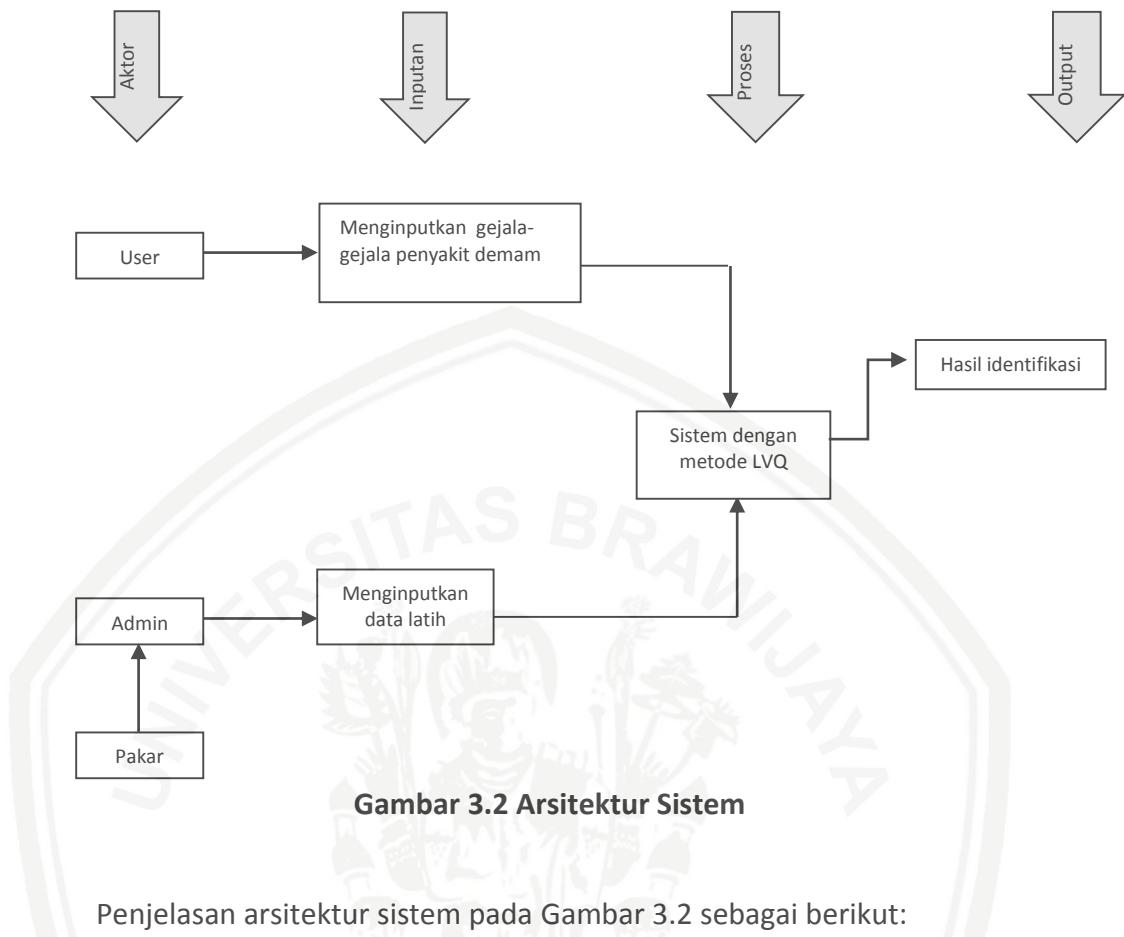
Sedangkan kebutuhan perangkat lunak yang digunakan yaitu :

- a. Sistem Operasi Windows 8 Enterprise 64-bit
- b. Code Editor : NetBeans

3.4 Perancangan Sistem

Sistem untuk klasifikasi penyakit demam menggunakan metode LVQ, dimana pada metode LVQ akan diberikan gejala-gejala penyakit yang dipilih sehingga menghasilkan diagnosa sementara penyakit yang diderita berdasarkan gejala-gejala tersebut. Arsitektur sistem ditunjukkan pada Gambar 3.2.





Penjelasan arsitektur sistem pada Gambar 3.2 sebagai berikut:

1. Admin memasukkan data latih untuk pelatihan dengan diagnosa masing-masing penyakit dan gejala-gejala penyakit.
2. Setelah data tersimpan, maka data yang sudah dimasukkan akan digunakan sebagai data pelatihan yang akan digunakan sebagai acuan dalam klasifikasi penyakit demam menggunakan metode LVQ.
3. Sistem menerima masukan dari user terhadap gejala yang telah diamati, kemudian hasil dari masukan user tersebut akan dilakukan proses perhitungan untuk mencari hasil diagnosa sementara.



3.5 Implementasi

Implementasi sistem ini dilakukan dengan menggunakan bahasa pemrograman java dan beberapa software yang dibutuhkan. Implementasi sistem meliputi :

1. Pembuatan antarmuka pengguna/ user
2. Memasukkan data pelatihan ke dalam database
3. Penerapan Jaringan Syaraf Tiruan dengan menggunakan metode LVQ

Skenario yang dijalankan dalam sistem ini yaitu :

1. User dihadapkan dengan beberapa pertanyaan terkait dengan gejala-gejala penyakit yang berhubungan dengan gejala demam.
2. Jika user sudah selesai mengisi atau menjawab pertanyaan tersebut maka selanjutnya sistem akan mengambil kesimpulan tentang penyakit menggunakan LVQ.
3. User akan mendapatkan hasil diagnosa sementara penyakit.

3.6 Pengujian dan Analisis

Pengujian dapat dilakukan dengan cara memeriksa apakah sistem sudah berjalan dengan baik (sesuai harapan) atau tidak. Membandingkan hasil identifikasi dan perhitungan yang dilakukan secara manual juga dilakukan dalam pengujian sistem serta membandingkan hasil dari sistem dengan hasil dari pakarnya. Output sistem yang akan diuji berupa hasil identifikasi penyakit. Pengujian akan dilakukan sejumlah 10 kali per parameter dan akan dihasilkan data kuantitatif yang kemudian dirata-rata dan hasilnya merupakan tingkat keakuriasan sistem.

Hasil dari analisis didapat dari pengujian yang dilakukan dengan tujuan untuk mengetahui kekurangan dari sistem yang telah dibuat dan apa saja yang harus diperbaiki.

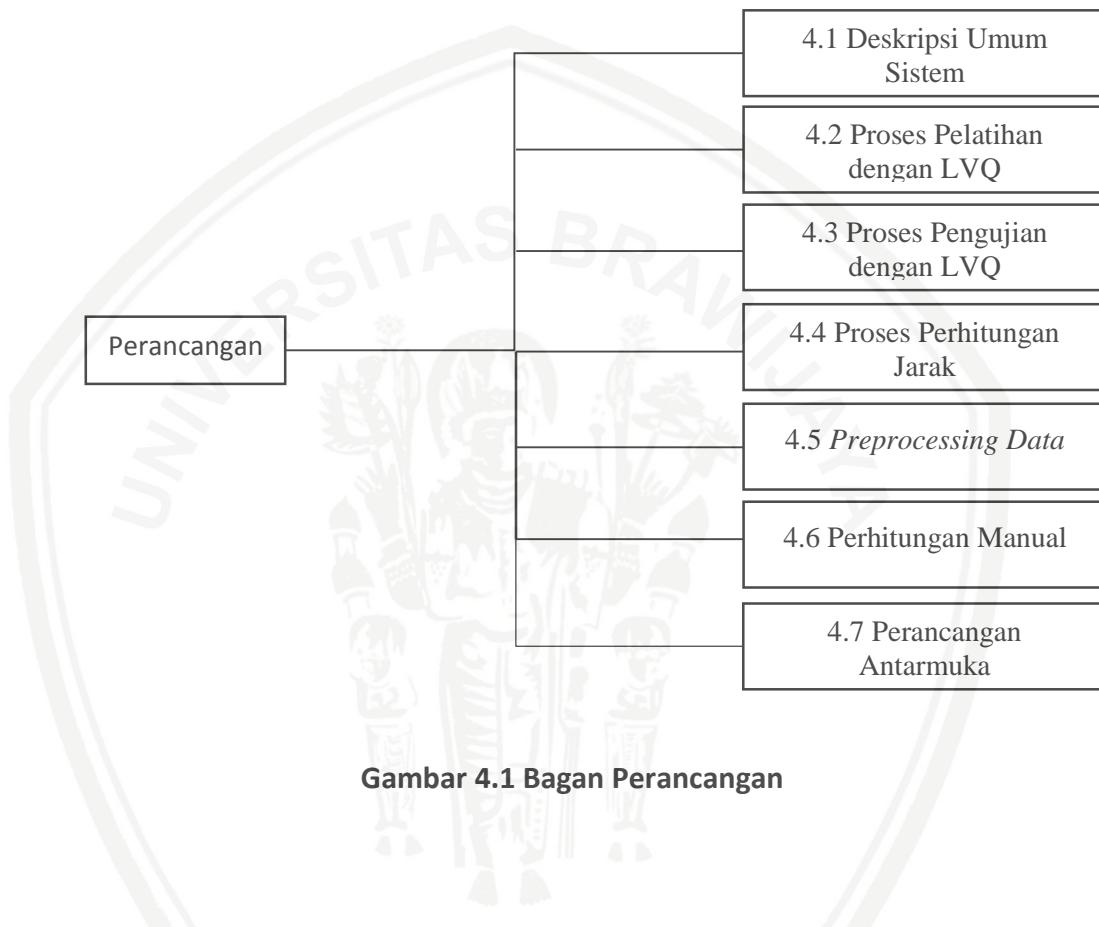
3.7 Kesimpulan

Kesimpulan mengenai semua tahapan yang telah dilalui serta saran yang berkenan dengan hasil yang telah dicapai. Kesimpulan diambil dari tahap perancangan hingga analisa dan pengujian sistem. Saran berfungsi untuk memperbaiki kesalahan yang berguna dalam pengembangan lebih lanjut.



BAB 4 PERANCANGAN

Pada bab ini membahas proses perancangan yang meliputi deskripsi umum sistem, proses manualisasi seperti proses pelatihan dengan LVQ, proses pengujian dengan LVQ, proses perhitungan jarak, dan perancangan antarmuka. Alur perancangan ditunjukkan pada Gambar 4.1.

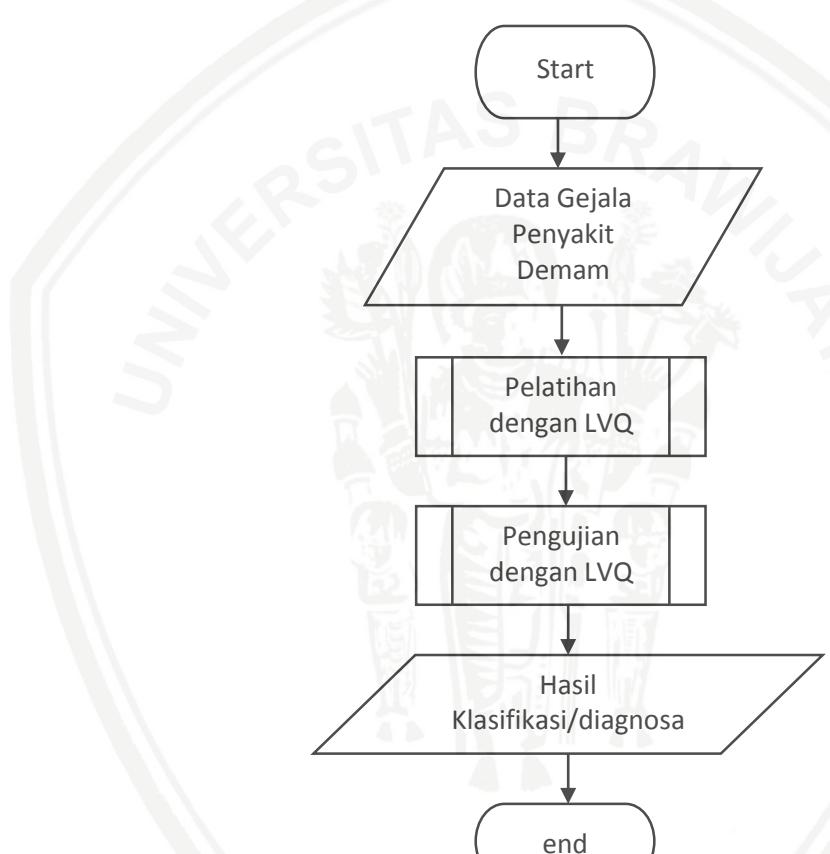


4.1 Deskripsi Umum Sistem

Sistem “Implementasi Metode *Learning Vector Quantization* untuk Klasifikasi Penyakit Demam” yang dibuat dalam skripsi ini adalah untuk mengimplementasikan metode LVQ dan untuk mengetahui hasil setelah diimplementasikan. Inputan yang digunakan dalam sistem ini berupa 15 gejala penyakit dengan opsi dan bobot masing-masing yang selanjutnya akan diproses menggunakan metode LVQ. LVQ menggunakan parameter berupa learning rate, pengali learning rate, jumlah data latih, *epoch* maksimum dan nilai minimum α yang mempengaruhi hasil akurasi.



Beberapa tahapan yang meliputi proses kerja sistem secara umum adalah input gejala penyakit , pelatihan dengan algoritma LVQ, pengujian dengan algoritma LVQ, dan output hasil identifikasi penyakit. Pada proses pelatihan menggunakan LVQ, user perlu memasukkan data latih untuk pembelajaran serta memasukkan parameter-parameter yang dibutuhkan. Sedangkan pada proses pengujian, user memilih gejala-gejala penyakit yang dirasakan, kemudian sistem akan mengubah gejala-gejala tersebut menjadi nilai angka yang nantinya akan diproses untuk mendapatkan hasil identifikasi dari penyakit demam yang dibagi menjadi tiga kelas (demam tifoid, malaria, dan demam berdarah). Proses kerja sistem secara umum ditunjukkan pada Gambar 4.2.



Gambar 4.2 Diagram Alir Proses Kerja Sistem Secara Umum

4.2 Proses Pelatihan dengan LVQ

Proses Pelatihan dengan LVQ dilakukan setelah user memasukkan data latih. Proses pelatihan dimulai dari menginisialisasi vektor input pada data latih hingga perubahan bobot. Proses pelatihan dengan LVQ melalui tahapan-tahapan sebagai berikut :

1. *Input* data latih. Data yang digunakan adalah data gejala penyakit demam malaria, tifoid dan demam berdarah yang sudah diberikan skor masing-masing.
2. Inisialisasi bobot awal dengan nilai acak, inisialisasi laju pembelajaran, konstanta pengurang laju pembelajaran, laju pembelajaran minimum, dan iterasi maksimum. Kemudian melakukan langkah 3 hingga 6 dengan ketentuan iterasi kurang dari iterasi maksimum atau laju pembelajaran lebih dari laju pembelajaran minimum.
3. Tentukan kelas C_j dengan menghitung jarak *Euclidean* minimum dari $|x - w_j|$ untuk setiap vektor pelatihan x .
4. Update w_j yaitu :

Jika $T=C_j$, maka

$$W_j(\text{baru}) = w_j(\text{lama}) + \alpha[x - w_j(\text{lama})]$$

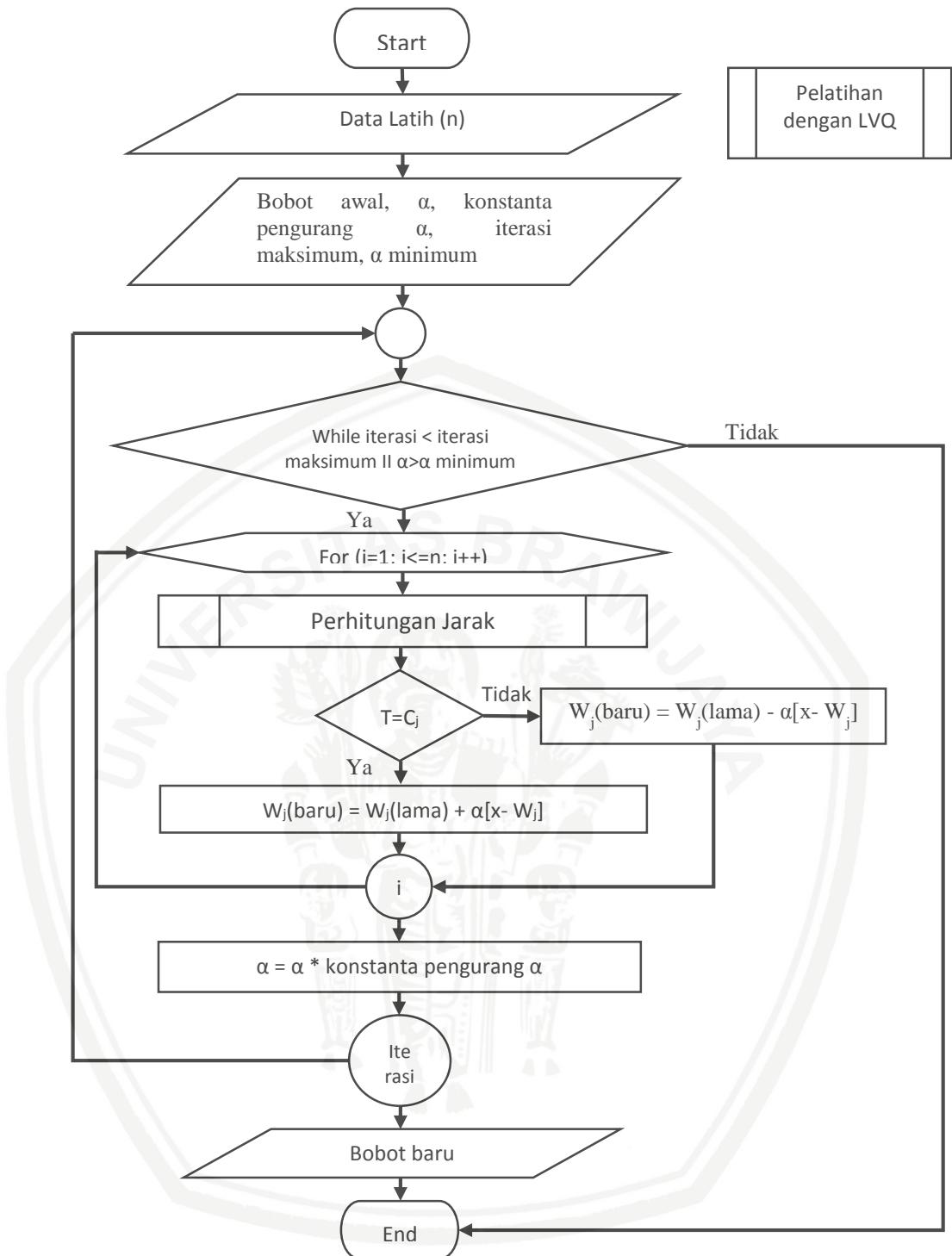
Jika $T \neq C_j$, maka

$$W_j(\text{baru}) = w_j(\text{lama}) - \alpha[x - w_j(\text{lama})]$$

5. Kalikan laju pembelajaran dengan konstanta pengurang laju pembelajaran untuk mengurangi laju pembelajaran.
6. Mengembalikan bobot baru yang diperoleh.

Proses pelatihan dengan LVQ ditunjukkan pada Gambar 4.3.





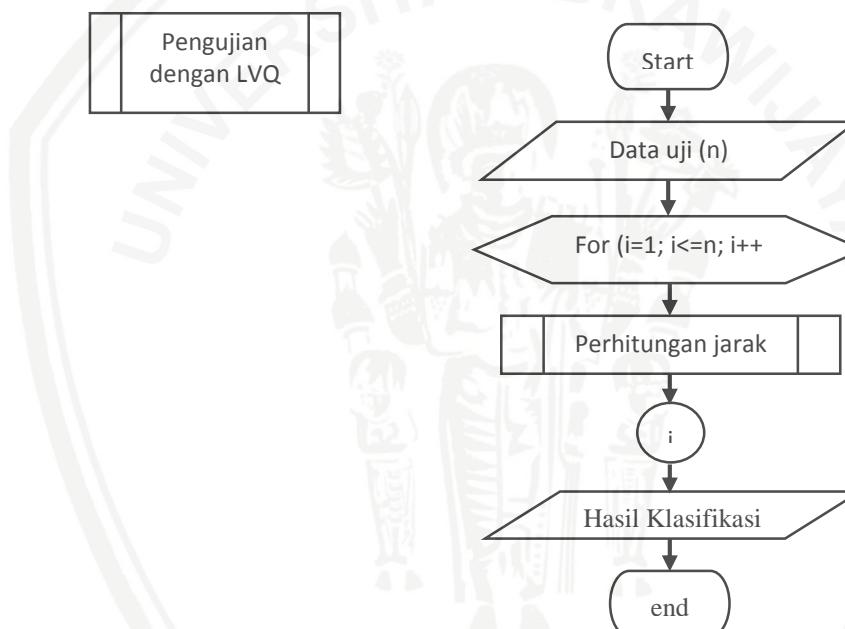
Gambar 4.3 Diagram Alir Proses Pelatihan LVQ

4.3 Proses Pengujian dengan LVQ

Proses pengujian dimulai dengan menginputkan data uji hingga proses penentuan kelas. Langkah-langkah proses pengujian dengan LVQ adalah sebagai berikut:

1. Menginputkan data uji. Data uji berupa data gejala penyakit yang sudah diberikan nilai masing-masing.
2. Menginisialisasi bobot hasil dari pelatihan.
3. Menghitung jarak *Euclidean* minimum dari $||x-w_j||$ bagi setiap vektor pengujian x untuk menentukan kelas C_j .
4. Mengembalikan data hasil klasifikasi yang berupa kelas hasil pengujian menggunakan LVQ.

Diagram alir proses pengujian ditunjukkan pada Gambar 4.4.



Gambar 4.4 Diagram Alir Proses Pengujian LVQ

4.4 Proses Perhitungan Jarak

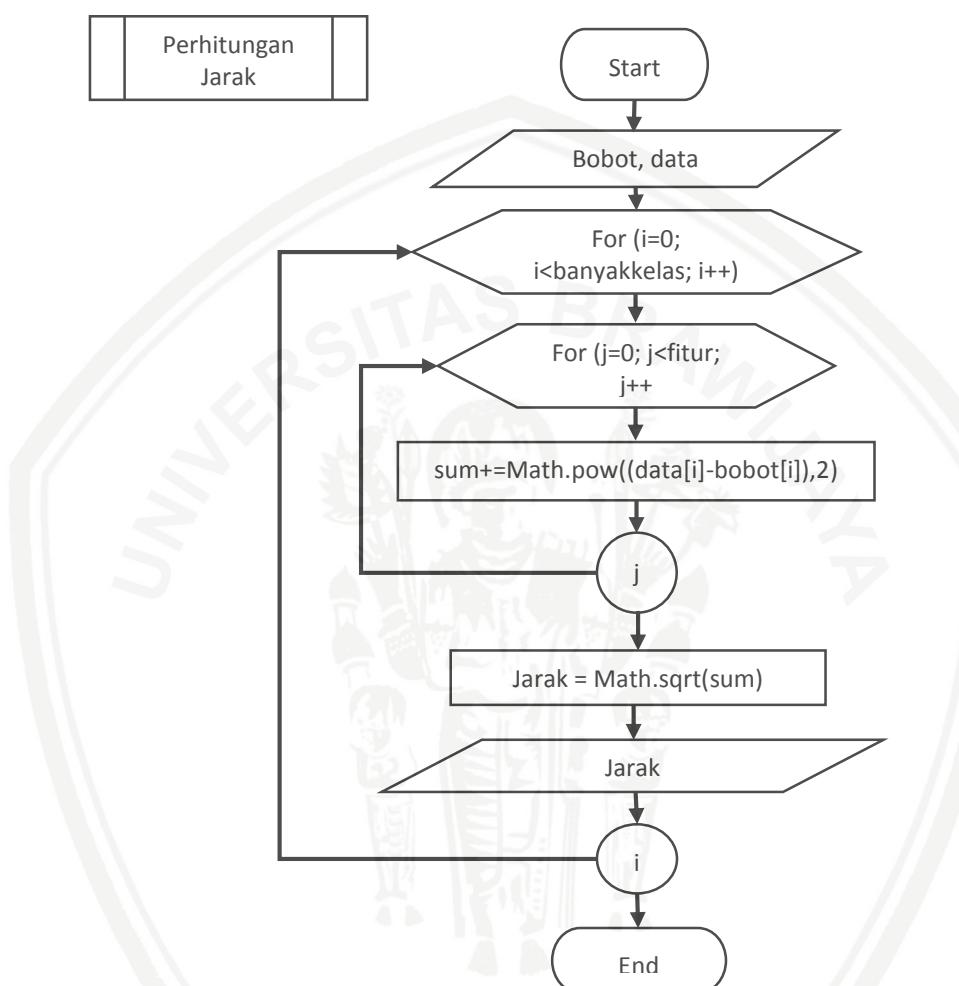
Langkah-langkah perhitungan jarak yaitu dimulai dari inisialisasi vector input data dan bobot hingga menghasilkan nilai jarak. Langkah-langkah mencari jarak pada LVQ adalah sebagai berikut :

1. Inisialisasi variabel yang menyimpan nilai jarak



2. Mencari jumlah seluruh hasil kuadrat dari pengurangan vector data ke-n dengan bobot ke-i menggunakan perulangan sebanyak jumlah fitur.
3. Mengakarkan hasil dari proses penjumlahan sebelumnya.
4. Menentukan jarak minimum untuk menentukan kelas baru.

Diagram alir proses perhitungan jarak ditunjukkan pada Gambar 4.5.



Gambar 4.5 Diagram Alir Proses Perhitungan Jarak

4.5 Preprocessing Data

Tahap preprocessing data adalah tahap yang digunakan untuk melakukan pembobotan terhadap gejala-gejala demam yang diberikan oleh pakar RSUD Selasih yaitu dr. Anda Citra Utama, SpPD. Proses pemberian bobot untuk masing-masing gejala digunakan untuk mempermudah proses klasifikasi karena data bersifat kategorikal. Klasifikasi demam terbatas pada demam tifoid, demam berdarah, dan malaria yang memiliki kemiripan gejala. Pembobotan masing-masing gejala pada penyakit demam ditunjukkan pada Tabel 4.1.



Tabel 4.1 Bobot untuk Gejala Demam

Variabel	Keluhan	Skor
Demam Intermittent (putus-putus) (G1)	Ya	0,8
	Tidak	0
Demam menggigil (G2)	Berat	0,85
	Sedang	0,5
	Tidak ada	0
Demam terutama malam hari (G3)	Ya	0,75
	Tidak	0
Lama demam (G4)	>7 hari	0.75
	4-7 hari	0.5
	1-3 hari	0.2
Sakit kepala (G5)	Berat	0.9
	Sedang	0.7
	Tidak ada	0
Sakit tulang dan sendi (G6)	Berat	0.9
	Sedang	0.75
	Tidak ada	0
Mual dan muntah (G7)	Berat	0.85
	Sedang	0.6
	Tidak ada	0
Mencret atau susah BAB(konstipasi) (G8)	Ada	0.6
	Tidak ada	0.2
Nyeri perut (G9)	Berat	0.8
	Sedang	0.4
	Tidak ada	0
Bintik merah (ptekie) pada kulit (G10)	Berat	1
	Sedang	0,7
	Tidak ada	0
Lidah kotor (coated tongue) (G11)	Berat	0,85
	Sedang	0,4
	Tidak ada	0



Bradikardi relative (G12)	Ya	0,75
	Tidak ada	0
Pembesaran hati (G13)	Ya	0,5
	Tidak ada	0
Pembesaran limpa (G14)	Ya	0,65
	Tidak ada	0
Kulit lembab / Keringat (G15)	Berat	0,85
	Sedang	0,4
	Tidak ada	0

Kelas yang digunakan dalam penelitian ini adalah 3 kelas yang merupakan nama penyakit yaitu :

1. k1 = demam tifoid
2. k2 = demam malaria
3. k3 = demam berdarah

4.6 Perhitungan Manual

Pada sub bab ini akan dijelaskan tentang proses manualisasi dari klasifikasi penyakit demam tifoid, malaria dan demam berdarah. Proses manualisasi dari klasifikasi demam dimulai dari pelatihan dengan LVQ, pengujian dengan LVQ dan perhitungan akurasi. Data yang digunakan untuk proses manualisasi adalah 15 data latih yaitu masing-masing 5 data untuk setiap kelas, dan 6 data uji yaitu 2 data uji untuk masing-masing kelas.

4.6.1 Manualisasi Pelatihan menggunakan LVQ

Untuk pelatihan LVQ diperlukan data latih dan bobot awal. Proses pelatihan menggunakan 15 data latih yaitu 5 data untuk kelas 1 (demam tifoid), 5 data untuk kelas 2 (demam malaria), dan 5 data untuk kelas 3 (demam berdarah). Dalam proses perhitungan manualisasi ini penulis menentukan nilai parameter yang ada pada LVQ yaitu nilai *alpha* sebesar 0,1. Tabel 4.2 adalah data latih yang digunakan yaitu 15 data dari keseluruhan data yang digunakan pada penelitian ini.

Tabel 4.2 Data Latih

Da ta	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	G13	G14	G15	KE LAS
1	0	0,5	0,75	0,75	0,9	0,75	0,85	0,6	0,8	0	0,85	0,75	0,5	0,65	0	1
2	0,8	0,5	0,75	0,5	0	0,75	0,6	0,6	0	0	0	0,75	0,5	0,65	0	1
3	0	0,5	0,75	0,5	0,7	0,75	0,85	0,6	0,4	0	0,85	0	0	0,65	0	1



4	0,8	0,5	0	0,5	0,7	0	0	0,2	0,8	0	0,85	0,75	0,5	0,65	0	1																																																																																	
5	0	0	0,7 5	0,75	0,9	0	0,6	0,6	0,8	0	0	0,75	0,5	0	0	1																																																																																	
6	0,8	0,8 5	0	0,5	0,9	0	0	0,2	0	0	0	0	0,5	0,65	0,85	2																																																																																	
7	0,8	0,8 5	0	0,5	0,7	0	0	0,6	0,4	0,7	0	0	0,5	0,65	0,85	2																																																																																	
8	0,8	0,5	0	0,5	0,9	0,7 5	0	0,2	0	0	0	0	0,5	0,65	0,85	2																																																																																	
9	0,8	0,8 5	0	0,5	0,7	0	0	0,2	0	0	0	0	0,5	0,65	0,85	2																																																																																	
10	0,8	0,8 5	0	0,2	0,9	0,8 5	0,2	0	0	0,4	0	0,5	0,65	0,85	2	11	0	0,5	0,7 5	0,5	0,7	0,7 5	0,6	0,2	0,8	0,7	0,4	0	0,5	0,65	0	3	12	0	0	0	0,2	0,9	0,9 5	0,2	0,4	0,7	0	0	0	0,65	0	3	13	0	0	0	0,2	0,7	0,9 5	0	0,2	0,4	0,7	0	0	0,65	0	3	14	0	0,5	0	0,2	0,9	0,9 5	0,2	0,4	0,7	0,4	0	0,5	0	0	3	15	0	0	0	0,2	0,9	0,9 5	0,2	0	1	0	0	0	0	0	0	3
11	0	0,5	0,7 5	0,5	0,7	0,7 5	0,6	0,2	0,8	0,7	0,4	0	0,5	0,65	0	3																																																																																	
12	0	0	0	0,2	0,9	0,9 5	0,2	0,4	0,7	0	0	0	0,65	0	3	13	0	0	0	0,2	0,7	0,9 5	0	0,2	0,4	0,7	0	0	0,65	0	3	14	0	0,5	0	0,2	0,9	0,9 5	0,2	0,4	0,7	0,4	0	0,5	0	0	3	15	0	0	0	0,2	0,9	0,9 5	0,2	0	1	0	0	0	0	0	0	3																																	
13	0	0	0	0,2	0,7	0,9 5	0	0,2	0,4	0,7	0	0	0,65	0	3																																																																																		
14	0	0,5	0	0,2	0,9	0,9 5	0,2	0,4	0,7	0,4	0	0,5	0	0	3	15	0	0	0	0,2	0,9	0,9 5	0,2	0	1	0	0	0	0	0	0	3																																																																	
15	0	0	0	0,2	0,9	0,9 5	0,2	0	1	0	0	0	0	0	0	3																																																																																	

Tahapan proses pelatihan dilakukan sesuai dengan teori algoritma LVQ, yaitu sebagai berikut :

- Inisialisasi bobot. Bobot yang digunakan untuk pelatihan LVQ menggunakan bobot yang didapatkan dari data acak dari masing-masing data per kelas. Bobot awal yang digunakan dalam manualisasi LVQ ini adalah data yang tidak digunakan dalam data latih pada perhitungan manualisasi. Bobot awal (w) dapat dilihat pada tabel 4.3 berikut.

Tabel 4.3 Bobot Awal

W 1	0	0,5	0,7 5	0,7 5	0, 7	0	0,6	0, 2	0, 8	0, 7	0,8 5	0,7 5	0, 5	0,6 5	0
W 2	0	0,8 5	0	0,2	0, 9	0	0,6	0, 2	0	0, 7	0	0	0	0	0,4
W 3	0	0	0	0,2	0, 9	0, 9	0,5	0, 2	0, 8	1	0	0	0, 5	0	0

- Inisialisasi laju pembelajaran (α) = 0,1 ,konstanta pengurang laju pembelajaran = 0,5.
- Inisialisasi iterasi maksimum = 5
- Ketika iterasi kurang dari iterasi maksimum, maka lakukan langkah selanjutnya.
- Menghitung jarak *Euclidean* antara vector *input* dan bobot vector ke-j.

Untuk vektor pelatihan x , tentukan kelas C_j dengan menghitung jarak *Euclidean* minimum dari $||x - w_j||$.



1. Perhitungan jarak *Euclidean* antara bobot dengan data latih 1 adalah sebagai berikut :

$$D_1W = \sqrt{\left(0 - 0\right)^2 + \left(0,5 - 0,5\right)^2 + \left(0,75 - 0,75\right)^2 + \left(0,75 - 0,75\right)^2 + \dots + \left(0,9 - 0,75\right)^2 + \left(0,9 - 0,9\right)^2 + \left(0,75 - 0\right)^2 + \left(0,85 - 0,6\right)^2 + \left(0,6 - 0,2\right)^2 + \left(0,9 - 0,9\right)^2 + \left(0,75 - 0,9\right)^2 + \left(0,85 - 0,85\right)^2 + \left(0,6 - 0,2\right)^2 + \left(0,8 - 0,8\right)^2 + \left(0,8 - 0\right)^2 + \left(0 - 0,7\right)^2 + \left(0 - 0,7\right)^2 + \left(0,85 - 0,85\right)^2 + \left(0,85 - 0\right)^2 + \left(0,85 - 0\right)^2 + \left(0,75 - 0,75\right)^2 + \left(0,75 - 0\right)^2 + \left(0,5 - 0,5\right)^2 + \left(0,65 - 0,65\right)^2 + \left(0,65 - 0\right)^2 + \left(0 - 0\right)^2 + \left(0 - 0,4\right)^2}$$

$$\text{Maka } D_1W = \begin{pmatrix} 1,146734494 \\ 2,24053565 \\ 2,00124961 \end{pmatrix}$$

Dari hasil perhitungan maka dipilih jarak terdekat yaitu ke w_1 , sehingga $C_j=1$.

- $T=1$ dan $C_j=1$, maka update bobot sebagai berikut:

$$w_1(\text{baru}) = w_1(\text{lama}) + \alpha [x - w_1(\text{lama})]$$

Sehingga diperoleh bobot baru sebagai berikut :

W_1	0	0,5	0,75	0,75	0,72	0,075	0,625	0,24	0,8	0,63	0,85	0,75	0,5	0,65	0
W_2	0	0,85	0	0,2	0,9	0	0,6	0,2	0	0,7	0	0	0	0	0,4
W_3	0	0	0	0,2	0,9	0,9	0,85	0,2	0,8	1	0	0	0,5	0	0

2. Perhitungan jarak *Euclidean* antara bobot dengan data latih 2 adalah sebagai berikut :



$$D_2 W = \sqrt{\left(\begin{matrix} 0,8 - 0 \\ 0,8 - 0 \\ 0,8 - 0 \end{matrix} \right)^2 + \left(\begin{matrix} 0,5 - 0,5 \\ 0,5 - 0,85 \\ 0,5 - 0 \end{matrix} \right)^2 + \left(\begin{matrix} 0,75 - 0,75 \\ 0,75 - 0 \\ 0,75 - 0 \end{matrix} \right)^2 + \left(\begin{matrix} 0,5 - 0,75 \\ 0,5 - 0,2 \\ 0,5 - 0,2 \end{matrix} \right)^2 + \dots \right.} \\ \left. + \left(\begin{matrix} 0 - 0,72 \\ 0 - 0,9 \\ 0 - 0,9 \end{matrix} \right)^2 + \left(\begin{matrix} 0,75 - 0,075 \\ 0,75 - 0 \\ 0,75 - 0,9 \end{matrix} \right)^2 + \left(\begin{matrix} 0,6 - 0,625 \\ 0,6 - 0,6 \\ 0,6 - 0,85 \end{matrix} \right)^2 + \left(\begin{matrix} 0,6 - 0,24 \\ 0,6 - 0,2 \\ 0,6 - 0,2 \end{matrix} \right)^2 \right. \\ \left. + \left(\begin{matrix} 0 - 0,8 \\ 0 - 0 \\ 0 - 0,8 \end{matrix} \right)^2 + \left(\begin{matrix} 0 - 0,63 \\ 0 - 0,7 \\ 0 - 1 \end{matrix} \right)^2 + \left(\begin{matrix} 0 - 0,85 \\ 0 - 0 \\ 0 - 0 \end{matrix} \right)^2 \right. \\ \left. + \left(\begin{matrix} 0,75 - 0,75 \\ 0,75 - 0 \\ 0,75 - 0 \end{matrix} \right)^2 + \left(\begin{matrix} 0,5 - 0,5 \\ 0,5 - 0 \\ 0,5 - 0,5 \end{matrix} \right)^2 + \left(\begin{matrix} 0,65 - 0,65 \\ 0,65 - 0 \\ 0,65 - 0 \end{matrix} \right)^2 \right. \\ \left. + \left(\begin{matrix} 0 - 0 \\ 0 - 0,4 \\ 0 - 0 \end{matrix} \right)^2 \right)$$

$$\text{Maka } D_2 W = \begin{pmatrix} 1,88842527 \\ 2,198294794 \\ 2,285278976 \end{pmatrix}$$

Dari hasil perhitungan maka dipilih jarak terdekat yaitu ke w_1 , sehingga $C_j=1$.

- $T=1$ dan $C_j=1$, maka update bobot sebagai berikut:

$$w_1(\text{baru}) = w_1(\text{lama}) + \alpha [x - w_1(\text{lama})]$$

Sehingga diperoleh bobot baru sebagai berikut :

W_1	0,08	0,5	0,75	0,725	0,648	0,1425	0,6225	0,276	0,72	0,567	0,765	0,75	0,5	0,65	0
W_2	0	0,85	0	0,2	0,9	0	0,6	0,2	0	0,7	0	0	0	0	0,4
W_3	0	0	0	0,2	0,9	0,9	0,85	0,2	0,8	1	0	0	0,5	0	0

3. Perhitungan jarak *Euclidean* antara bobot dengan data latih 3 adalah sebagai berikut :

$$D_3 W = \sqrt{\left(\begin{matrix} 0 - 0,08 \\ 0 - 0 \\ 0 - 0 \end{matrix} \right)^2 + \left(\begin{matrix} 0,5 - 0,5 \\ 0,5 - 0,85 \\ 0,5 - 0 \end{matrix} \right)^2 + \left(\begin{matrix} 0,75 - 0,75 \\ 0,75 - 0 \\ 0,75 - 0 \end{matrix} \right)^2 + \left(\begin{matrix} 0,5 - 0,725 \\ 0,5 - 0,2 \\ 0,5 - 0,2 \end{matrix} \right)^2 + \dots \right.} \\ \left. + \left(\begin{matrix} 0,7 - 0,648 \\ 0,7 - 0,9 \\ 0,7 - 0,9 \end{matrix} \right)^2 + \left(\begin{matrix} 0,75 - 0,1425 \\ 0,75 - 0 \\ 0,75 - 0,9 \end{matrix} \right)^2 + \left(\begin{matrix} 0,85 - 0,6225 \\ 0,85 - 0,6 \\ 0,85 - 0,85 \end{matrix} \right)^2 + \left(\begin{matrix} 0,6 - 0,276 \\ 0,6 - 0,2 \\ 0,6 - 0,2 \end{matrix} \right)^2 \right. \\ \left. + \left(\begin{matrix} 0,4 - 0,72 \\ 0,4 - 0 \\ 0,4 - 0,8 \end{matrix} \right)^2 + \left(\begin{matrix} 0 - 0,567 \\ 0 - 0,7 \\ 0 - 1 \end{matrix} \right)^2 + \left(\begin{matrix} 0,85 - 0,765 \\ 0,85 - 0 \\ 0,85 - 0 \end{matrix} \right)^2 \right. \\ \left. + \left(\begin{matrix} 0 - 0,75 \\ 0 - 0 \\ 0 - 0 \end{matrix} \right)^2 + \left(\begin{matrix} 0 - 0,5 \\ 0 - 0 \\ 0 - 0,5 \end{matrix} \right)^2 + \left(\begin{matrix} 0,65 - 0,65 \\ 0,65 - 0 \\ 0,65 - 0 \end{matrix} \right)^2 \right. \\ \left. + \left(\begin{matrix} 0 - 0 \\ 0 - 0,4 \\ 0 - 0 \end{matrix} \right)^2 \right)}$$



$$\text{Maka } D_3 W = \begin{pmatrix} 1,352453881 \\ 1,885470763 \\ 1,918332609 \end{pmatrix}$$

Dari hasil perhitungan maka dipilih jarak terdekat yaitu ke w_1 , sehingga $C_j=1$.

- $T=1$ dan $C_j=1$, maka update bobot sebagai berikut:

$$w_1(\text{baru}) = w_1(\text{lama}) + \alpha [x - w_1(\text{lama})]$$

Sehingga diperoleh bobot baru sebagai berikut :

w_1	0,07 2	0,5	0,7 5	0,70 25	0,65 32	0,203 25	0,645 25	0,30 84	0,68 8	0,51 03	0,77 35	0,67 5	0,4 5	0,6 5	0
w_2	0	0,8 5	0	0,2	0,9	0	0,6	0,2	0	0,7	0	0	0	0	0, 4
w_3	0	0	0	0,2	0,9	0,9	0,85	0,2	0,8	1	0	0	0,5	0	0

4. Perhitungan jarak *Euclidean* antara bobot dengan data latih 4 adalah sebagai berikut :

$$D_4 W = \sqrt{\left(\frac{(0,8 - 0,072)^2}{0,8 - 0} + \frac{(0,5 - 0,5)^2}{0,5 - 0,85} + \frac{(0 - 0,75)^2}{0 - 0} + \frac{(0,5 - 0,7025)^2}{0,5 - 0,2} + \dots \right) + \left(\frac{(0,7 - 0,6532)^2}{0,7 - 0,9} + \frac{(0 - 0,20325)^2}{0 - 0} + \frac{(0 - 0,64525)^2}{0 - 0,6} + \frac{(0,2 - 0,3084)^2}{0,2 - 0,2} \right) + \left(\frac{(0,8 - 0,688)^2}{0,8 - 0} + \frac{(0 - 0,5103)^2}{0 - 0,7} + \frac{(0,85 - 0,7735)^2}{0,85 - 0} \right) + \left(\frac{(0,75 - 0,675)^2}{0,75 - 0} + \frac{(0,5 - 0,45)^2}{0,5 - 0} + \frac{(0,65 - 0,65)^2}{0,65 - 0} \right) + \left(\frac{(0 - 0)^2}{0 - 0,4} \right)}$$

$$\text{Maka } D_4 W = \begin{pmatrix} 1,37550591 \\ 2,12132034 \\ 2,29346899 \end{pmatrix}$$

Dari hasil perhitungan maka dipilih jarak terdekat yaitu ke w_1 , sehingga $C_j=1$.

- $T=1$ dan $C_j=1$, maka update bobot sebagai berikut:

$$w_1(\text{baru}) = w_1(\text{lama}) + \alpha [x - w_1(\text{lama})]$$

Sehingga diperoleh bobot baru sebagai berikut :

w_1	0,14 48	0,5	0,67 5	0,68 225	0,65 788	0,18 2925	0,5807 25	0,29 756	0,69 92	0,45 927	0,78 115	0,6825	0,455	0,65	0
w_2	0	0,85	0	0,2	0,9	0	0,6	0,2	0	0,7	0	0	0	0	0,4
w_3	0	0	0	0,2	0,9	0,9	0,85	0,2	0,8	1	0	0	0,5	0	0



5. Perhitungan jarak *Euclidean* antara bobot dengan data latih 5 adalah sebagai berikut :

$$D_5W = \sqrt{\left(0 - 0,1448\right)^2 + \left(0 - 0,5\right)^2 + \left(0,75 - 0,675\right)^2 + \left(0,75 - 0,68225\right)^2 + \dots + \left(0,9 - 0,65788\right)^2 + \left(0 - 0,182925\right)^2 + \left(0,6 - 0,580725\right)^2 + \left(0,6 - 0,29756\right)^2 + \left(0,8 - 0,6992\right)^2 + \left(0 - 0,45927\right)^2 + \left(0 - 0,78115\right)^2 + \left(0,75 - 0,6825\right)^2 + \left(0,5 - 0,455\right)^2 + \left(0 - 0,65\right)^2 + \left(0 - 0\right)^2 + \left(0 - 0,4\right)^2}$$

$$\text{Maka } D_5W = \begin{pmatrix} 1,313572753 \\ 1,962141687 \\ 1,860107524 \end{pmatrix}$$

Dari hasil perhitungan maka dipilih jarak terdekat yaitu ke w_1 , sehingga $C_j=1$.

- $T=1$ dan $C_j=1$, maka update bobot sebagai berikut:

$$w_1(\text{baru}) = w_1(\text{lama}) + \alpha [x - w_1(\text{lama})]$$

Sehingga diperoleh bobot baru sebagai berikut :

w_1	0,13 032	0,45	0,68 25	0,68 9025	0,68 2092	0,16 4633	0,5826 53	0,32 7804	0,70 928	0,41 3343	0,70 3035	0,6892 5	0,4595	0,58 5	0
w_2	0	0,85	0	0,2	0,9	0	0,6	0,2	0	0,7	0	0	0	0	0,4
w_3	0	0	0	0,2	0,9	0,9	0,85	0,2	0,8	1	0	0	0,5	0	0

6. Perhitungan jarak *Euclidean* antara bobot dengan data latih 6 adalah sebagai berikut :

$$D_6W = \sqrt{\left(0,8 - 0,13032\right)^2 + \left(0,85 - 0,45\right)^2 + \left(0 - 0,6825\right)^2 + \left(0,5 - 0,689025\right)^2 + \dots + \left(0,9 - 0,682092\right)^2 + \left(0 - 0,164633\right)^2 + \left(0 - 0,582653\right)^2 + \left(0,2 - 0,327804\right)^2 + \left(0 - 0,70928\right)^2 + \left(0 - 0,413343\right)^2 + \left(0 - 0,703035\right)^2 + \left(0 - 0,68925\right)^2 + \left(0,5 - 0,4595\right)^2 + \left(0,65 - 0,585\right)^2 + \left(0,85 - 0\right)^2 + \left(0,85 - 0,4\right)^2}$$



$$\text{Maka } D_6 W = \begin{pmatrix} 1,977886027 \\ 1,566843962 \\ 2,40208243 \end{pmatrix}$$

Dari hasil perhitungan maka dipilih jarak terdekat yaitu ke w_2 , sehingga $C_j=2$.

- $T=2$ dan $C_j=2$, maka update bobot sebagai berikut:

$$W_2(\text{baru}) = w_2(\text{lama}) + \alpha [x - w_2(\text{lama})]$$

Sehingga diperoleh bobot baru sebagai berikut :

W_1	0,13 032	0,4 5	0,68 25	0,68 9025	0,68 2092	0,16 4633	0,5826 53	0,327 804	0,7092 8	0,41 3343	0,70 3035	0,68 925	0,45 95	0,585	0
W_2	0,08 5	0,8 5	0	0,23	0,9	0	0,54	0,2	0	0,63	0	0	0,05	0,065	0,44 5
W_3	0	0	0	0,2	0,9	0,9	0,85	0,2	0,8	1	0	0	0,5	0	0

7. Perhitungan jarak *Euclidean* antara bobot dengan data latih 7 adalah sebagai berikut :

$$D_7 W = \sqrt{\left(\frac{(0,8 - 0,13032)^2}{0,8 - 0,08} + \frac{(0,85 - 0,45)^2}{0,85 - 0,85} + \frac{(0 - 0,6825)^2}{0 - 0} + \frac{(0,5 - 0,689025)^2}{0,5 - 0,23} + \dots \right) + \left(\frac{(0,7 - 0,682092)^2}{0,7 - 0,9} + \frac{(0 - 0,164633)^2}{0 - 0,9} + \frac{(0 - 0,582653)^2}{0 - 0,54} + \frac{(0,6 - 0,327804)^2}{0,6 - 0,2} \right) + \left(\frac{(0,4 - 0,70928)^2}{0,4 - 0,4} + \frac{(0,7 - 0,413343)^2}{0,7 - 0,63} + \frac{(0 - 0,703035)^2}{0 - 0} \right) + \left(\frac{(0 - 0,68925)^2}{0 - 0} + \frac{(0,5 - 0,4595)^2}{0,5 - 0,05} + \frac{(0,65 - 0,585)^2}{0,65 - 0,065} \right) + \left(\frac{(0,85 - 0)^2}{0,85 - 0} + \frac{(0,85 - 0,445)^2}{0,85 - 0} \right)}$$

$$\text{Maka } D_7 W = \begin{pmatrix} 1,851086852 \\ 1,398767314 \\ 2,140093456 \end{pmatrix}$$

Dari hasil perhitungan maka dipilih jarak terdekat yaitu ke w_2 , sehingga $C_j=2$.

- $T=2$ dan $C_j=2$, maka update bobot sebagai berikut:

$$W_2(\text{baru}) = w_2(\text{lama}) + \alpha [x - w_2(\text{lama})]$$

Sehingga diperoleh bobot baru sebagai berikut :

W_1	0,13 032	0,4 5	0,68 25	0,68 9025	0,68 2092	0,16 4633	0,5826 53	0,327 804	0,7092 8	0,41 3343	0,70 3035	0,68 925	0,45 95	0,585	0
W_2	0,15 2	0,8 5	0	0,25 7	0,88	0	0,486	0,24	0,04	0,63 7	0	0	0,09 5	0,1235	0,48 55
W_3	0	0	0	0,2	0,9	0,9	0,85	0,2	0,8	1	0	0	0,5	0	0

8. Perhitungan jarak *Euclidean* antara bobot dengan data latih 8 adalah sebagai berikut :

$$D_8 W = \sqrt{\left(\begin{array}{c} 0,8 - 0,13032 \\ 0,8 - 0,152 \\ 0,8 - 0 \end{array} \right)^2 + \left(\begin{array}{c} 0,5 - 0,45 \\ 0,5 - 0,85 \\ 0,5 - 0 \end{array} \right)^2 + \left(\begin{array}{c} 0 - 0,6825 \\ 0 - 0 \\ 0 - 0 \end{array} \right)^2 + \left(\begin{array}{c} 0,5 - 0,689025 \\ 0,5 - 0,257 \\ 0,5 - 0,2 \end{array} \right)^2 + \dots \right.} \\ \left. + \left(\begin{array}{c} 0,9 - 0,682092 \\ 0,9 - 0,88 \\ 0,9 - 0,9 \end{array} \right)^2 + \left(\begin{array}{c} 0,75 - 0,164633 \\ 0,75 - 0 \\ 0,75 - 0,9 \end{array} \right)^2 + \left(\begin{array}{c} 0 - 0,582653 \\ 0 - 0,486 \\ 0 - 0,85 \end{array} \right)^2 + \left(\begin{array}{c} 0,2 - 0,327804 \\ 0,2 - 0,24 \\ 0,2 - 0,2 \end{array} \right)^2 \right. \\ \left. + \left(\begin{array}{c} 0 - 0,70928 \\ 0 - 0,04 \\ 0 - 0,8 \end{array} \right)^2 + \left(\begin{array}{c} 0 - 0,413343 \\ 0 - 0,637 \\ 0 - 1 \end{array} \right)^2 + \left(\begin{array}{c} 0 - 0,703035 \\ 0 - 0 \\ 0 - 0 \end{array} \right)^2 \right. \\ \left. + \left(\begin{array}{c} 0 - 0,68925 \\ 0 - 0 \\ 0 - 0 \end{array} \right)^2 + \left(\begin{array}{c} 0,5 - 0,4595 \\ 0,5 - 0,095 \\ 0,5 - 0,5 \end{array} \right)^2 + \left(\begin{array}{c} 0,65 - 0,585 \\ 0,65 - 0,1235 \\ 0,65 - 0 \end{array} \right)^2 \right. \\ \left. + \left(\begin{array}{c} 0,85 - 0 \\ 0,85 - 0,4855 \\ 0,85 - 0 \end{array} \right)^2 \right)$$

$$\text{Maka } D_8 W = \begin{pmatrix} 2,017445014 \\ 1,543892969 \\ 2,123676058 \end{pmatrix}$$

Dari hasil perhitungan maka dipilih jarak terdekat yaitu ke w_2 , sehingga $C_j=2$.

- $T=2$ dan $C_j=2$, maka update bobot sebagai berikut:

$$W_2(\text{baru}) = w_2(\text{lama}) + \alpha [x - w_2(\text{lama})]$$

Sehingga diperoleh bobot baru sebagai berikut :

W_1	0,13 032	0,4 5	0,68 25	0,68 9025	0,68 2092	0,16 4633	0,5826 53	0,327 804	0,7092 8	0,41 3343	0,70 3035	0,68 925	0,45 95	0,585 0,585	0
W_2	0,21 68	0,8 15	0 0	0,28 13	0,88 2	0,07 5	0,4374	0,236	0,036 33	0,57 0	0 0	0,13 55	0,1761 5	0,52 195	
W_3	0	0	0	0,2	0,9	0,9	0,85	0,2	0,8	1	0	0	0,5	0	0

9. Perhitungan jarak *Euclidean* antara bobot dengan data latih 9 adalah sebagai berikut :

$$D_9 W = \sqrt{\left(\begin{array}{c} 0,8 - 0,13032 \\ 0,8 - 0,2168 \\ 0,8 - 0 \end{array} \right)^2 + \left(\begin{array}{c} 0,85 - 0,45 \\ 0,85 - 0,815 \\ 0,85 - 0 \end{array} \right)^2 + \left(\begin{array}{c} 0 - 0,6825 \\ 0 - 0 \\ 0 - 0 \end{array} \right)^2 + \left(\begin{array}{c} 0,5 - 0,689025 \\ 0,5 - 0,2813 \\ 0,5 - 0,2 \end{array} \right)^2 + \dots \right.} \\ \left. + \left(\begin{array}{c} 0,7 - 0,682092 \\ 0,7 - 0,882 \\ 0,7 - 0,9 \end{array} \right)^2 + \left(\begin{array}{c} 0 - 0,164633 \\ 0 - 0,075 \\ 0 - 0,9 \end{array} \right)^2 + \left(\begin{array}{c} 0 - 0,582653 \\ 0 - 0,4374 \\ 0 - 0,85 \end{array} \right)^2 + \left(\begin{array}{c} 0,2 - 0,327804 \\ 0,2 - 0,236 \\ 0,2 - 0,2 \end{array} \right)^2 \right. \\ \left. + \left(\begin{array}{c} 0 - 0,70928 \\ 0 - 0,036 \\ 0 - 0,8 \end{array} \right)^2 + \left(\begin{array}{c} 0 - 0,413343 \\ 0 - 0,5733 \\ 0 - 1 \end{array} \right)^2 + \left(\begin{array}{c} 0 - 0,703035 \\ 0 - 0 \\ 0 - 0 \end{array} \right)^2 \right. \\ \left. + \left(\begin{array}{c} 0 - 0,68925 \\ 0 - 0 \\ 0 - 0 \end{array} \right)^2 + \left(\begin{array}{c} 0,5 - 0,4595 \\ 0,5 - 0,1355 \\ 0,5 - 0,5 \end{array} \right)^2 + \left(\begin{array}{c} 0,65 - 0,585 \\ 0,65 - 0,17615 \\ 0,65 - 0 \end{array} \right)^2 \right. \\ \left. + \left(\begin{array}{c} 0,85 - 0 \\ 0,85 - 0,52195 \\ 0,85 - 0 \end{array} \right)^2 \right)$$



$$\text{Maka } D_9W = \begin{pmatrix} 1,965927246 \\ 1,189756469 \\ 2,410394159 \end{pmatrix}$$

Dari hasil perhitungan maka dipilih jarak terdekat yaitu ke w_2 , sehingga $C_j=2$.

- $T=2$ dan $C_j=2$, maka update bobot sebagai berikut:

$$W_2(\text{baru}) = w_2(\text{lama}) + \alpha [x - w_2(\text{lama})]$$

Sehingga diperoleh bobot baru sebagai berikut :

W_1	0,13 032	0,45	0,68 25	0,68 9025	0,68 2092	0,16 4633	0,5826 53	0,32 7804	0,70 928	0,41 3343	0,70 3035	0,68 925	0,4595 0,5	0,58 5	0
W_2	0,27 512	0,81 85	0 0	0,30 317	0,86 38	0,06 75	0,3936 6	0,23 24	0,03 24	0,51 597	0 0	0,1719 5	0,22 3535	0,55 4755	
W_3	0	0	0	0,2	0,9	0,9	0,85	0,2	0,8	1	0	0	0,5	0	0

10. Perhitungan jarak *Euclidean* antara bobot dengan data latih 10 adalah sebagai berikut :

$$D_{10}W = \sqrt{\left(\frac{0,8 - 0,13032}{0,8 - 0} \right)^2 + \left(\frac{0,85 - 0,45}{0,85 - 0} \right)^2 + \left(\frac{0 - 0,6825}{0 - 0} \right)^2 + \left(\frac{0,2 - 0,689025}{0,2 - 0,2} \right)^2 + \dots + \left(\frac{0,9 - 0,682092}{0,9 - 0,9} \right)^2 + \left(\frac{0 - 0,164633}{0 - 0,9} \right)^2 + \left(\frac{0,85 - 0,582653}{0,85 - 0,85} \right)^2 + \left(\frac{0,2 - 0,327804}{0,2 - 0,2} \right)^2 + \left(\frac{0 - 0,70928}{0 - 0,8} \right)^2 + \left(\frac{0 - 0,413343}{0 - 1} \right)^2 + \left(\frac{0,4 - 0,703035}{0,4 - 0} \right)^2 + \left(\frac{0 - 0,68925}{0 - 0} \right)^2 + \left(\frac{0,5 - 0,4595}{0,5 - 0,5} \right)^2 + \left(\frac{0,65 - 0,585}{0,65 - 0} \right)^2 + \left(\frac{0,85 - 0}{0,85 - 0} \right)^2 + \left(\frac{0,85 - 0,554755}{0,85 - 0} \right)^2}$$

$$\text{Maka } D_{10}W = \begin{pmatrix} 1,856074052 \\ 1,142904882 \\ 2,262189205 \end{pmatrix}$$

Dari hasil perhitungan maka dipilih jarak terdekat yaitu ke w_2 , sehingga $C_j=2$.

- $T=2$ dan $C_j=2$, maka update bobot sebagai berikut:

$$W_2(\text{baru}) = w_2(\text{lama}) + \alpha [x - w_2(\text{lama})]$$

Sehingga diperoleh bobot baru sebagai berikut :

W_1	0,13 032	0,45	0,68 25	0,68 9025	0,68 2092	0,16 4633	0,5826 53	0,32 7804	0,70 928	0,41 3343	0,70 3035	0,68 925	0,4595 0,5	0,58 5	0
W_2	0,32 7608	0,82 165	0 0	0,29 2853	0,86 742	0,06 075	0,4392 94	0,22 916	0,02 916	0,46 4373	0,04 0,04	0,2047 55	0,26 6182	0,58 428	
W_3	0	0	0	0,2	0,9	0,9	0,85	0,2	0,8	1	0	0	0,5	0	0

11. Perhitungan jarak *Euclidean* antara bobot dengan data latih 11 adalah sebagai berikut :

$$D_{11}W = \sqrt{\left(0 - 0,13032\right)^2 + \left(0,5 - 0,45\right)^2 + \left(0,75 - 0,6825\right)^2 + \left(0,5 - 0,689025\right)^2 + \dots + \left(0,7 - 0,682092\right)^2 + \left(0,75 - 0,164633\right)^2 + \left(0,6 - 0,582653\right)^2 + \left(0,2 - 0,327804\right)^2 + \left(0,7 - 0,86742\right)^2 + \left(0,75 - 0,06075\right)^2 + \left(0,6 - 0,439294\right)^2 + \left(0,2 - 0,22916\right)^2 + \left(0,8 - 0,70928\right)^2 + \left(0,7 - 0,413343\right)^2 + \left(0,4 - 0,703035\right)^2 + \left(0,8 - 0,02916\right)^2 + \left(0,7 - 0,464373\right)^2 + \left(0,4 - 0,04\right)^2 + \left(0,8 - 0,8\right)^2 + \left(0,7 - 1\right)^2 + \left(0,4 - 0\right)^2 + \left(0 - 0,68925\right)^2 + \left(0,5 - 0,4595\right)^2 + \left(0,65 - 0,585\right)^2 + \left(0,5 - 0,204755\right)^2 + \left(0,65 - 0,266182\right)^2 + \left(0 - 0\right)^2 + \left(0 - 0,58428\right)^2 + \left(0 - 0\right)^2}$$

$$\text{Maka } D_{11}W = \begin{pmatrix} 1,04045374 \\ 1,643517727 \\ 1,303840481 \end{pmatrix}$$

Dari hasil perhitungan maka dipilih jarak terdekat yaitu ke w_1 , sehingga $C_j=1$.

- $T=3$ dan $C_j=1$, maka update bobot sebagai berikut:

$$W_1(\text{baru}) = w_1(\text{lama}) - \alpha [x - w_1(\text{lama})]$$

Sehingga diperoleh bobot baru sebagai berikut :

W_1	0,14 3352	0,44 5	0,67 575	0,70 7928	0,68 0301	0,10 6096	0,5809 18	0,34 0584 4	0,70 0208	0,38 4677	0,73 3339	0,75 8175	0,4554 5	0,57 85	0
W_2	0,32 7608	0,82 165	0	0,29 2853	0,86 742	0,06 075	0,4392 94	0,22 916	0,02 916	0,46 4373	0,04	0	0,2047 55	0,26 6182	0,58 428
W_3	0	0	0	0,2	0,9	0,9	0,85	0,2	0,8	1	0	0	0,5	0	0

12. Perhitungan jarak *Euclidean* antara bobot dengan data latih 12 adalah sebagai berikut :

$$D_{12}W = \sqrt{\left(0 - 0,143352\right)^2 + \left(0 - 0,445\right)^2 + \left(0 - 0,67575\right)^2 + \left(0,2 - 0,707928\right)^2 + \dots + \left(0,9 - 0,680301\right)^2 + \left(0,9 - 0,106096\right)^2 + \left(0,85 - 0,580918\right)^2 + \left(0,2 - 0,3405844\right)^2 + \left(0,9 - 0,86742\right)^2 + \left(0,9 - 0,06075\right)^2 + \left(0,85 - 0,439294\right)^2 + \left(0,2 - 0,22916\right)^2 + \left(0,4 - 0,700208\right)^2 + \left(0,7 - 0,384677\right)^2 + \left(0 - 0,733339\right)^2 + \left(0,4 - 0,02916\right)^2 + \left(0,7 - 0,464373\right)^2 + \left(0 - 0,04\right)^2 + \left(0,4 - 0,8\right)^2 + \left(0,7 - 1\right)^2 + \left(0 - 0\right)^2 + \left(0 - 0,758175\right)^2 + \left(0 - 0,45545\right)^2 + \left(0,65 - 0,5785\right)^2 + \left(0 - 0,204755\right)^2 + \left(0,65 - 0,266182\right)^2 + \left(0 - 0\right)^2 + \left(0 - 0,58428\right)^2 + \left(0 - 0\right)^2}$$



$$\text{Maka } D_{12}W = \begin{pmatrix} 1,794056533 \\ 1,54636843 \\ 0,960468636 \end{pmatrix}$$

Dari hasil perhitungan maka dipilih jarak terdekat yaitu ke w_3 , sehingga $C_j=3$.

- $T=3$ dan $C_j=3$, maka update bobot sebagai berikut:

$$W_3(\text{baru}) = w_3(\text{lama}) + \alpha [x - w_3(\text{lama})]$$

Sehingga diperoleh bobot baru sebagai berikut :

W_1	0,14 3352	0,44 5	0,67 575	0,70 7928	0,68 0301	0,10 6096	0,5809 18	0,34 0584	0,70 0208	0,38 4677	0,73 3339	0,75 8175	0,4554 5	0,57 85	0
W_2	0,32 7608	0,82 165	0	0,29 2853	0,86 742	0,06 075	0,4392 94	0,22 916	0,02 916	0,46 4373	0,04	0	0,2047 55	0,26 6182	0,58 428
W_3	0	0	0	0,2	0,9	0,9	0,85	0,2	0,76	0,97	0	0	0,45	0,06 5	0

13. Perhitungan jarak *Euclidean* antara bobot dengan data latih 13 adalah sebagai berikut :

$$D_{13}W = \sqrt{\left(0 - 0,143352\right)^2 + \left(0 - 0,445\right)^2 + \left(0 - 0,67575\right)^2 + \left(0,2 - 0,707928\right)^2 + \dots + \left(0,7 - 0,680301\right)^2 + \left(0,9 - 0,106096\right)^2 + \left(0 - 0,580918\right)^2 + \left(0,2 - 0,3405844\right)^2 + \left(0,7 - 0,86742\right)^2 + \left(0,9 - 0,06075\right)^2 + \left(0 - 0,439294\right)^2 + \left(0,2 - 0,22916\right)^2 + \left(0,4 - 0,700208\right)^2 + \left(0,7 - 0,384677\right)^2 + \left(0 - 0,733339\right)^2 + \left(0,4 - 0,02916\right)^2 + \left(0,7 - 0,464373\right)^2 + \left(0 - 0,04\right)^2 + \left(0 - 0,758175\right)^2 + \left(0 - 0,45545\right)^2 + \left(0,65 - 0,5785\right)^2 + \left(0 - 0,0\right)^2 + \left(0 - 0,45\right)^2 + \left(0,65 - 0,266182\right)^2 + \left(0 - 0,065\right)^2 + \left(0 - 0,58428\right)^2 + \left(0 - 0\right)^2}$$

$$\text{Maka } D_{13}W = \begin{pmatrix} 1,853596369 \\ 1,562857358 \\ 1,228708672 \end{pmatrix}$$

Dari hasil perhitungan maka dipilih jarak terdekat yaitu ke w_3 , sehingga $C_j=3$.

- $T=3$ dan $C_j=3$, maka update bobot sebagai berikut:

$$W_3(\text{baru}) = w_3(\text{lama}) + \alpha [x - w_3(\text{lama})]$$

Sehingga diperoleh bobot baru sebagai berikut :



W ₁	0,14 3352	0,44 5	0,67 575	0,70 7928	0,68 0301	0,10 6096	0,5809 18	0,34 0584 4	0,70 0208	0,38 4677	0,73 3339	0,75 8175	0,4554 5	0,57 85	0
W ₂	0,32 7608	0,82 165	0	0,29 2853	0,86 742	0,06 075	0,4392 94	0,22 916	0,02 916	0,46 4373	0,04	0	0,2047 55	0,26 6182	0,58 428
W ₃	0	0	0	0,2	0,88	0,9	0,765	0,2	0,72 4	0,94 3	0	0	0,405 35	0,12 35	0

14. Perhitungan jarak *Euclidean* antara bobot dengan data latih 14 adalah sebagai berikut :

$$D_{14}W = \sqrt{\left(0 - 0,143352\right)^2 + \left(0,5 - 0,445\right)^2 + \left(0 - 0,67575\right)^2 + \left(0,2 - 0,707928\right)^2 + \dots + \left(0,9 - 0,680301\right)^2 + \left(0,9 - 0,106096\right)^2 + \left(0,85 - 0,580918\right)^2 + \left(0,2 - 0,3405844\right)^2 + \left(0,9 - 0,86742\right)^2 + \left(0,9 - 0,06075\right)^2 + \left(0,85 - 0,439294\right)^2 + \left(0,2 - 0,22916\right)^2 + \left(0,4 - 0,700208\right)^2 + \left(0,7 - 0,384677\right)^2 + \left(0,4 - 0,733339\right)^2 + \left(0,4 - 0,02916\right)^2 + \left(0,7 - 0,464373\right)^2 + \left(0,4 - 0,04\right)^2 + \left(0 - 0,758175\right)^2 + \left(0,5 - 0,45545\right)^2 + \left(0 - 0,5785\right)^2 + \left(0 - 0,266182\right)^2 + \left(0 - 0,58428\right)^2}$$

$$\text{Maka } D_{14}W = \begin{pmatrix} 1,649566016 \\ 1,38433604 \\ 0,77841329 \end{pmatrix}$$

Dari hasil perhitungan maka dipilih jarak terdekat yaitu ke w_3 , sehingga $C_j=3$.

- $T=3$ dan $C_j=3$, maka update bobot sebagai berikut:

$$W_3(\text{baru}) = w_3(\text{lama}) + \alpha [x - w_3(\text{lama})]$$

Sehingga diperoleh bobot baru sebagai berikut :

W ₁	0,14 3352	0,44 5	0,67 575	0,70 7928	0,68 0301	0,10 6096	0,5809 18	0,34 0584 4	0,70 0208	0,38 4677	0,73 3339	0,75 8175	0,4554 5	0,57 85	0
W ₂	0,32 7608	0,82 165	0	0,29 2853	0,86 742	0,06 075	0,4392 94	0,22 916	0,02 916	0,46 4373	0,04	0	0,2047 55	0,26 6182	0,58 428
W ₃	0	0,05	0	0,2	0,88 2	0,9	0,7735	0,2	0,69 16	0,91 87	0,04	0	0,4145 115	0,11 115	0

15. Perhitungan jarak *Euclidean* antara bobot dengan data latih 15 adalah sebagai berikut :



$$D_{15}W = \begin{pmatrix} (0 - 0,143352)^2 & (0 - 0,445)^2 & (0 - 0,67575)^2 & (0,2 - 0,707928)^2 \\ (0 - 0,327608)^2 & (0 - 0,82165)^2 & (0 - 0)^2 & (0,2 - 0,292853)^2 \\ 0 - 0 & 0 - 0,05 & 0 - 0 & 0,2 - 0,2 \\ & & & \\ + (0,7 - 0,680301)^2 & (0,9 - 0,106096)^2 & (0,85 - 0,580918)^2 & (0,2 - 0,3405844)^2 \\ + (0,7 - 0,86742)^2 & (0,9 - 0,06075)^2 & (0,85 - 0,439294)^2 & (0,2 - 0,22916)^2 \\ 0,7 - 0,882 & 0,9 - 0,9 & 0,85 - 0,7735 & 0,2 - 0,2 \\ & & & \\ + (0,8 - 0,700208)^2 & (1 - 0,384677)^2 & (0 - 0,733339)^2 & \\ + (0,8 - 0,02916)^2 & (1 - 0,464373)^2 & (0 - 0,04)^2 & \\ 0,8 - 0,6916 & 1 - 0,9187 & 0 - 0,04 & \\ & & & \\ + (0 - 0,758175)^2 & (0,5 - 0,45545)^2 & (0 - 0,5785)^2 & \\ 0 - 0 & 0,5 - 0,204755 & 0 - 0,266182 & \\ 0 - 0 & 0,5 - 0,4145 & 0 - 0,11115 & \\ & & & \\ + (0 - 0)^2 & & & \\ 0 - 0,58428 & & & \\ 0 - 0 & & & \end{pmatrix}$$

$$\text{Maka } D_{15}W = \begin{pmatrix} 1,869194089 \\ 1,753582753 \\ 0,2847825 \end{pmatrix}$$

Dari hasil perhitungan maka dipilih jarak terdekat yaitu ke w_3 , sehingga $C_j=3$.

- $T=3$ dan $C_j=3$, maka update bobot sebagai berikut:

$$W_3(\text{baru}) = w_3(\text{lama}) + \alpha [x - w_3(\text{lama})]$$

Sehingga diperoleh bobot baru sebagai berikut :

W_1	0,14 3352	0,44 5	0,67 575	0,70 7928	0,68 0301	0,10 6096	0,5809 18	0,34 0584 4	0,70 0208	0,38 4677	0,73 3339	0,75 8175	0,4554 5	0,57 85	0
W_2	0,32 7608	0,82 165	0	0,29 2853	0,86 742	0,06 075	0,4392 94	0,22 916	0,02 916	0,46 4373	0,04	0	0,2047 55	0,26 6182	0,58 428
W_3	0	0,04 5	0	0,2 38	0,86 0,9	0,7811 5	0,2	0,70 244	0,92 683	0,03 6	0	0,4230 5	0,10 0035	0	

- Kurangi laju pembelajaran dengan cara :

$$\alpha = \alpha * c = 0,1 * 0,5 = 0,05$$

- Selama iterasi < iterasi maksimum , maka lakukan langkah seperti diatas untuk setiap vektor pelatihan x.
- Setelah pelatihan berhenti pada iterasi ke-5 (iterasi maksimum), bobot baru yang dihasilkan adalah sebagai berikut :

W_1	0. 22 24	0.42 09	0.63 90	0.67 85	0.66 51	0.19 50	0.57 63	0.41 95	0.64 06	0.21 22	0.66 38	0.74 40	0.43 07	0.54 71	0
W_2	0. 50 69	0.80 59	0	0.34 71	0.84 94	0.09 45	0.34 16	0.24 74	0.04 74	0.33 94	0.05 73	0	0.31 68	0.41 19	0.68 51
W_3	0	0.07 87	0.01 32	0.20 53	0.84 04	0.89 74	0.73 43	0.20 00	0.64 25	0.87 66	0.06 29	0	0.37 30	0.17 65	0



4.6.2 Manualisasi Pengujian menggunakan LVQ

Pengujian menggunakan algoritma LVQ dengan data uji sebanyak 6 data. 2 data dari kelas 1 (demam tifoid), dua data dari kelas 2 (demam malaria), dan dua data dari kelas 3 (demam berdarah). Klasifikasi dengan metode LVQ menggunakan bobot *update* terakhir pada pelatihan LVQ untuk mengekelompokkan data uji. Data uji untuk manualisasi pengujian menggunakan LVQ ditunjukkan pada tabel 4.4.

Tabel 4.4 Data Uji

No	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	G13	G14	G15	Kelas	
1	0	0	0	0,2	0,9	0,9	0,6	0,2	0,4	0,7	0	0	0,5	0,65	0	3	
2	0,8	0,5	0	0,5	0,9	0	0,6	0,2	0	0	0	0	0,5	0,65	0,85	2	
3	0	0	0	0,2	0,7	0,9	0,5	0,2	0,4	0,7	0	0	0	0	0	3	
4	0	0	0,5	0,7	0,5	0,9	0	0	0,6	0,4	0,7	0,4	0,75	0	0,65	0	1
5	0,8	0,5	0	0,5	0,7	0	0	0,2	0	0	0,4	0,75	0	0	0,4	2	
6	0,8	0	0,5	0,7	0,5	0,9	0	0,2	0,8	0	0,85	0	0,5	0,65	0,45	1	

Pengujian yaitu dengan menghitung jarak antara data uji dengan bobot akhir yang diperoleh pada proses pelatihan. Bobot yang digunakan adalah sebagai berikut :

w_1	0.22 24	0.42 09	0.63 90	0.67 85	0.66 51	0.19 50	0.57 63	0.41 95	0.64 06	0.21 22	0.66 38	0.74 40	0.43 07	0.54 71	0
w_2	0.50 69	0.80 59	0	0.34 71	0.84 94	0.09 45	0.34 16	0.24 74	0.04 74	0.33 94	0.05 73	0	0.31 68	0.41 19	0.68 51
w_3	0	0.07 87	0.01 32	0.20 53	0.84 04	0.89 74	0.73 43	0.64 02	0.87 25	0.06 66	0.06 29	0	0.37 30	0.17 65	0

Langkah-langkah pengujian yaitu :

- Untuk vektor pengujian x pertama, tentukan kelas C_j dengan menghitung jarak *Euclidean* minimum dari $\|x - w_j\|$. Jarak data uji 1 ke w dihitung dengan cara

$$D_{u1}W = \sqrt{\left(0 - 0.2224\right)^2 + \left(0 - 0.4209\right)^2 + \left(0 - 0.6390\right)^2 + \left(0,2 - 0.6785\right)^2 + \dots + \left(0 - 0.5069\right)^2 + \left(0 - 0.8059\right)^2 + \left(0 - 0 - 0\right)^2 + \left(0,2 - 0,3471\right)^2 + \dots + \left(0 - 0 - 0\right)^2 + \left(0 - 0.0787\right)^2 + \left(0,2 - 0,2053\right)^2 + \left(0,9 - 0,6651\right)^2 + \left(0,9 - 0,1950\right)^2 + \left(0,6 - 0,5763\right)^2 + \left(0,2 - 0,4195\right)^2 + \dots + \left(0,9 - 0,8494\right)^2 + \left(0,9 - 0,0945\right)^2 + \left(0,6 - 0,3416\right)^2 + \left(0,2 - 0,2474\right)^2 + \dots + \left(0,9 - 0,8404\right)^2 + \left(0,9 - 0,8974\right)^2 + \left(0,6 - 0,7343\right)^2 + \left(0,2 - 0,2\right)^2 + \left(0,4 - 0,6406\right)^2 + \left(0,7 - 0,2122\right)^2 + \left(0 - 0,6638\right)^2 + \left(0 - 0,7440\right)^2 + \dots + \left(0,4 - 0,0474\right)^2 + \left(0,7 - 0,3394\right)^2 + \left(0 - 0,0573\right)^2 + \left(0 - 0\right)^2 + \left(0,4 - 0,6425\right)^2 + \left(0,7 - 0,8766\right)^2 + \left(0 - 0,0629\right)^2 + \left(0 - 0\right)^2 + \left(0,5 - 0,4307\right)^2 + \left(0,65 - 0,5471\right)^2 + \left(0 - 0\right)^2 + \left(0 - 0\right)^2 + \left(0,5 - 0,3168\right)^2 + \left(0,65 - 0,4119\right)^2 + \left(0 - 0,6851\right)^2 + \left(0 - 0\right)^2 + \left(0,5 - 0,3730\right)^2 + \left(0,65 - 0,1765\right)^2 + \left(0 - 0\right)^2 + \left(0 - 0\right)^2}$$



$$D_{u1}W = \begin{pmatrix} 1.6644 \\ 1.5702 \\ 0.6019 \end{pmatrix}$$

Maka dipilih jarak yang terdekat yaitu ke w_3 , sehingga $C_j = 3$.

- Untuk vektor pengujian x kedua, tentukan kelas C_j dengan menghitung jarak Euclidean minimum dari $\|x - w_j\|$.

$$D_{u2}W = \sqrt{\left(0,8 - 0,2224\right)^2 + \left(0,85 - 0,4209\right)^2 + \left(0 - 0,6390\right)^2 + \left(0,5 - 0,6785\right)^2 + \dots + \left(0,9 - 0,6651\right)^2 + \left(0,9 - 0,8059\right)^2 + \left(0,6 - 0,5763\right)^2 + \left(0,2 - 0,4195\right)^2 + \dots + \left(0,9 - 0,8494\right)^2 + \left(0,9 - 0,0945\right)^2 + \left(0,6 - 0,3416\right)^2 + \left(0,2 - 0,2474\right)^2 + \dots + \left(0,9 - 0,8404\right)^2 + \left(0,9 - 0,8974\right)^2 + \left(0,6 - 0,7343\right)^2 + \left(0,2 - 0,2\right)^2 + \dots + \left(0 - 0,6406\right)^2 + \left(0 - 0,2122\right)^2 + \left(0 - 0,6638\right)^2 + \left(0 - 0,7440\right)^2 + \dots + \left(0 - 0,0474\right)^2 + \left(0 - 0,3394\right)^2 + \left(0 - 0,0573\right)^2 + \left(0 - 0\right)^2 + \dots + \left(0 - 0,6425\right)^2 + \left(0 - 0,8766\right)^2 + \left(0 - 0,0629\right)^2 + \left(0 - 0\right)^2 + \dots + \left(0,5 - 0,4307\right)^2 + \left(0,65 - 0,5471\right)^2 + \left(0,85 - 0\right)^2 + \left(0,85 - 0,6851\right)^2 + \dots + \left(0,5 - 0,3168\right)^2 + \left(0,65 - 0,4119\right)^2 + \left(0,85 - 0,6851\right)^2 + \dots + \left(0,5 - 0,3730\right)^2 + \left(0,65 - 0,1765\right)^2 + \left(0,85 - 0\right)^2 + \dots}$$

$$D_{u2}W = \begin{pmatrix} 1.8130 \\ 0,6556 \\ 2,0728 \end{pmatrix}$$

Maka dipilih jarak yang terdekat yaitu ke w_2 , sehingga $C_j = 2$.

- Untuk vektor pengujian x ketiga, tentukan kelas C_j dengan menghitung jarak Euclidean minimum dari $\|x - w_j\|$.

$$D_{u3}W = \sqrt{\left(0 - 0,2224\right)^2 + \left(0 - 0,4209\right)^2 + \left(0 - 0,6390\right)^2 + \left(0,2 - 0,6785\right)^2 + \dots + \left(0 - 0,5069\right)^2 + \left(0 - 0,8059\right)^2 + \left(0 - 0,0132\right)^2 + \left(0,2 - 0,3471\right)^2 + \dots + \left(0 - 0\right)^2 + \left(0 - 0,0787\right)^2 + \left(0 - 0\right)^2 + \left(0,2 - 0,2053\right)^2 + \dots + \left(0,7 - 0,6651\right)^2 + \left(0,9 - 0,1950\right)^2 + \left(0,85 - 0,5763\right)^2 + \left(0,2 - 0,4195\right)^2 + \dots + \left(0,7 - 0,8494\right)^2 + \left(0,9 - 0,0945\right)^2 + \left(0,85 - 0,3416\right)^2 + \left(0,2 - 0,2474\right)^2 + \dots + \left(0,7 - 0,8404\right)^2 + \left(0,9 - 0,8974\right)^2 + \left(0,85 - 0,7343\right)^2 + \left(0,2 - 0,2\right)^2 + \dots + \left(0,4 - 0,6406\right)^2 + \left(0,7 - 0,2122\right)^2 + \left(0 - 0,6638\right)^2 + \left(0 - 0,7440\right)^2 + \dots + \left(0,4 - 0,0474\right)^2 + \left(0,7 - 0,3394\right)^2 + \left(0 - 0,0573\right)^2 + \left(0 - 0\right)^2 + \dots + \left(0,4 - 0,6425\right)^2 + \left(0,7 - 0,8766\right)^2 + \left(0 - 0,0629\right)^2 + \left(0 - 0\right)^2 + \dots + \left(0 - 0,4307\right)^2 + \left(0 - 0,5471\right)^2 + \left(0 - 0\right)^2 + \left(0 - 0,6851\right)^2 + \dots + \left(0 - 0,3168\right)^2 + \left(0 - 0,4119\right)^2 + \left(0 - 0,6851\right)^2 + \dots + \left(0 - 0,3730\right)^2 + \left(0 - 0,1765\right)^2 + \left(0 - 0\right)^2 + \dots}$$

$$D_{u3}W = \begin{pmatrix} 1.8055 \\ 1,6902 \\ 0,5511 \end{pmatrix}$$

Maka dipilih jarak yang terdekat yaitu ke w_3 , sehingga $C_j = 3$.



- Untuk vektor pengujian x keempat, tentukan kelas C_j dengan menghitung jarak *Euclidean* minimum dari $\|x - w_j\|$.

$$D_{u4}W = \sqrt{\left(0 - 0.2224\right)^2 + \left(0 - 0.4209\right)^2 + \left(0.75 - 0.6390\right)^2 + \left(0.75 - 0.6785\right)^2 + \dots + \left(0.9 - 0.6651\right)^2 + \left(0 - 0.1950\right)^2 + \left(0 - 0.5763\right)^2 + \left(0.6 - 0.4195\right)^2 + \left(0.9 - 0.8494\right)^2 + \left(0 - 0.0945\right)^2 + \left(0 - 0.3416\right)^2 + \left(0.6 - 0.2474\right)^2 + \left(0.9 - 0.8404\right)^2 + \left(0 - 0.8974\right)^2 + \left(0 - 0.7343\right)^2 + \left(0.6 - 0.2\right)^2 + \left(0.4 - 0.6406\right)^2 + \left(0.7 - 0.2122\right)^2 + \left(0.4 - 0.6638\right)^2 + \left(0.75 - 0.7440\right)^2 + \left(0.4 - 0.0474\right)^2 + \left(0.7 - 0.3394\right)^2 + \left(0.4 - 0.0573\right)^2 + \left(0.75 - 0\right)^2 + \left(0.4 - 0.6425\right)^2 + \left(0.7 - 0.8766\right)^2 + \left(0.4 - 0.0629\right)^2 + \left(0.75 - 0\right)^2 + \left(0 - 0.4307\right)^2 + \left(0.65 - 0.5471\right)^2 + \left(0 - 0\right)^2 + \left(0 - 0.3168\right)^2 + \left(0.65 - 0.4119\right)^2 + \left(0 - 0.6851\right)^2 + \left(0 - 0\right)^2 + \left(0 - 0.3730\right)^2 + \left(0.65 - 0.1765\right)^2 + \left(0 - 0\right)^2}$$

$$D_{u4}W = \begin{pmatrix} 1,12406 \\ 1,8559 \\ 1,8663 \end{pmatrix}$$

Maka dipilih jarak yang terdekat yaitu ke w_1 , sehingga $C_j = 1$.

- Untuk vektor pengujian x kelima, tentukan kelas C_j dengan menghitung jarak *Euclidean* minimum dari $\|x - w_j\|$.

$$D_{u5}W = \sqrt{\left(0.8 - 0.2224\right)^2 + \left(0.5 - 0.4209\right)^2 + \left(0 - 0.6390\right)^2 + \left(0.5 - 0.6785\right)^2 + \dots + \left(0.7 - 0.6651\right)^2 + \left(0 - 0.1950\right)^2 + \left(0 - 0.5763\right)^2 + \left(0.2 - 0.4195\right)^2 + \left(0.7 - 0.8494\right)^2 + \left(0 - 0.0945\right)^2 + \left(0 - 0.3416\right)^2 + \left(0.2 - 0.2474\right)^2 + \left(0.7 - 0.8404\right)^2 + \left(0 - 0.8974\right)^2 + \left(0 - 0.7343\right)^2 + \left(0.2 - 0.2\right)^2 + \left(0 - 0.6406\right)^2 + \left(0 - 0.2122\right)^2 + \left(0.4 - 0.6638\right)^2 + \left(0.75 - 0.7440\right)^2 + \left(0 - 0.0474\right)^2 + \left(0 - 0.3394\right)^2 + \left(0.4 - 0.0573\right)^2 + \left(0.75 - 0\right)^2 + \left(0 - 0.6425\right)^2 + \left(0 - 0.8766\right)^2 + \left(0.4 - 0.0629\right)^2 + \left(0.75 - 0\right)^2 + \left(0 - 0.4307\right)^2 + \left(0 - 0.5471\right)^2 + \left(0.4 - 0\right)^2 + \left(0 - 0.3168\right)^2 + \left(0 - 0.4119\right)^2 + \left(0.4 - 0.6851\right)^2 + \left(0.4 - 0\right)^2 + \left(0 - 0.3730\right)^2 + \left(0 - 0.1765\right)^2 + \left(0.4 - 0\right)^2}$$

$$D_{u5}W = \begin{pmatrix} 1,5393 \\ 1,2254 \\ 2,1110 \end{pmatrix}$$

Maka dipilih jarak yang terdekat yaitu ke w_2 , sehingga $C_j = 2$.

- Untuk vektor pengujian x keenam, tentukan kelas C_j dengan menghitung jarak *Euclidean* minimum dari $\|x - w_j\|$.



$$D_{u6}W = \sqrt{\left(\frac{0,8 - 0,2224}{0,8 - 0,5069} \right)^2 + \left(\frac{0 - 0,4209}{0 - 0,8059} \right)^2 + \left(\frac{0,75 - 0,6390}{0,75 - 0,0132} \right)^2 + \left(\frac{0,5 - 0,6785}{0,5 - 0,3471} \right)^2 + \dots + \left(\frac{0,9 - 0,6651}{0,9 - 0,8494} \right)^2 + \left(\frac{0 - 0,1950}{0 - 0,0945} \right)^2 + \left(\frac{0,85 - 0,5763}{0,85 - 0,3416} \right)^2 + \left(\frac{0,2 - 0,4195}{0,2 - 0,2474} \right)^2 + \left(\frac{0,8 - 0,6406}{0,8 - 0,0474} \right)^2 + \left(\frac{0 - 0,2122}{0 - 0,3394} \right)^2 + \left(\frac{0,85 - 0,6638}{0,85 - 0,0573} \right)^2 + \left(\frac{0 - 0,7440}{0 - 0} \right)^2 + \left(\frac{0,5 - 0,4307}{0,5 - 0,3168} \right)^2 + \left(\frac{0,65 - 0,5471}{0,65 - 0,4119} \right)^2 + \left(\frac{0,45 - 0}{0,45 - 0,6851} \right)^2 } \\ D_{u6}W = \begin{pmatrix} 1,2836 \\ 1,7461 \\ 1,9884 \end{pmatrix}$$

Maka dipilih jarak yang terdekat yaitu ke w_1 , sehingga $C_j = 1$.

- Berdasarkan proses pengujian yang telah dilakukan, maka diperoleh data hasil klasifikasi perhitungan manual yang ditunjukkan pada Tabel 4.5.

Tabel 4.5 Hasil Klasifikasi Perhitungan Manual

N O	Gejala														T	K	
	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10	G11	G12	G13	G14	G15		
1	0	0	0	0,2	0,9	0,9	0,6	0,2	0,4	0,7	0	0	0,5	0,65	0	k3	k3
2	0,8	5	0	0,5	0,9	0	0,6	0,2	0	0	0	0	0,5	0,65	0,85	k2	k2
3	0	0	0	0,2	0,7	0,9	0,85	0,2	0,4	0,7	0	0	0	0	0	k3	k3
4	0	0	0,75	0,75	0,9	0	0	0,6	0,4	0,7	0,4	0,75	0	0,65	0	k1	k1
5	0,8	0,5	0	0,5	0,7	0	0	0,2	0	0	0,4	0,75	0	0	0,4	k2	k2
6	0,8	0	0,75	0,5	0,9	0	0,85	0,2	0,8	0	0,85	0	0,5	0,65	0,45	k1	k1

Dari tabel 4.5 dapat diketahui bahwa dari proses manualisasi data uji ke 1 masuk ke kelas 3 (demam berdarah), data uji ke 2 masuk ke kelas 2 (demam malaria), data uji ke 3 masuk ke kelas 3, data uji ke 4 masuk ke kelas 1 (demam tifoid), data uji ke 5 masuk ke kelas 2, dan data uji ke 6 masuk ke kelas 1. Dari keenam pengujian menghasilkan kelas yang sesuai dengan kelas sesungguhnya.

4.6.3 Perhitungan Akurasi

Setelah proses pengujian menggunakan LVQ selesai, maka selanjutnya adalah menghitung akurasi menggunakan persamaan 2.3.



$$\text{Akurasi} = \frac{\text{jumlah data yang relevan}}{\text{jumlah seluruh data}} \times 100 = \frac{6}{6} \times 100 = 100\%$$

Dari hasil perhitungan akurasi yang didapatkan dinyatakan dalam persentase yaitu 100%.

4.7 Perancangan Antarmuka

Pada tahap ini perancangan antarmuka dilakukan untuk implementasi metode LVQ dalam sistem “Implementasi Metode Learning Vector Quantization pada Klasifikasi Penyakit Demam”. Berikut ada 4 macam perancangan :

1. Antarmuka halaman beranda.
2. Antarmuka halaman pelatihan.
3. Antarmuka halaman pengujian.
4. Antarmuka halaman diagnosis.

4.7.1 Antarmuka Halaman Beranda

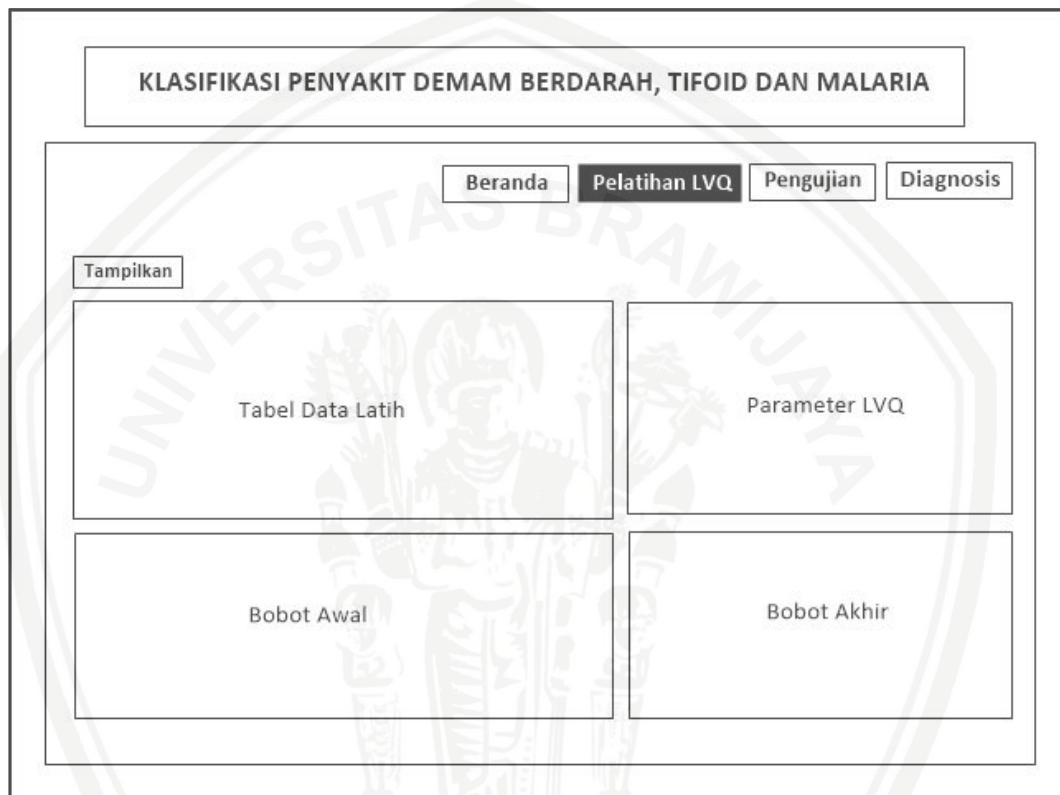
Halaman beranda merupakan halaman depan sistem yang pertama kali muncul saat aplikasi dijalankan. Halaman depan sistem ini terdiri dari *tabbed pane* untuk menampilkan halaman beranda, *tabbed pane* untuk membuka halaman pelatihan LVQ, *tabbed pane* untuk membuka halaman pengujian dan *tabbed pane* untuk membuka halaman diagnosis. Perancangan antarmuka halaman beranda ditunjukkan pada Gambar 4.6.



Gambar 4.6 Rancangan Antarmuka Halaman Beranda

4.7.2 Antarmuka Halaman Pelatihan

Pada halaman pelatihan LVQ akan ditampilkan informasi tentang proses pelatihan LVQ, informasi tersebut berupa data latih yang digunakan dalam proses pelatihan, parameter-parameter LVQ yang digunakan berdasarkan hasil dari uji coba terbaik dan bobot awal yang digunakan dalam proses pelatihan serta bobot akhir yang dihasilkan dari proses pelatihan dan akan digunakan sebagai bobot awal dalam proses pengujian. Perancangan antarmuka halaman pelatihan ditunjukkan pada Gambar 4.7.



Gambar 4.7 Rancangan Antarmuka Halaman Pelatihan

4.7.3 Antarmuka Halaman Pengujian

Pada halaman pengujian akan ditampilkan informasi tentang proses pengujian, informasi tersebut berupa data uji yang digunakan, bobot serta hasil akurasi. Pada halaman pengujian juga terdapat *tabbed pane* untuk membuka halaman beranda, halaman pelatihan LVQ dan halaman diagnosis. Perancangan antarmuka halaman pengujian ditunjukkan pada Gambar 4.8.





Gambar 4.8 Rancangan Antarmuka Halaman Pengujian

4.7.4 Antarmuka Halaman Diagnosis

Pada halaman diagnosis akan ditampilkan proses diagnosis dari penyakit demam berdasarkan gejala-gejala yang dialami oleh pasien. Gejala-gejala tersebut dipilih pada *combo box* yang disediakan. Halaman diagnosis juga menampilkan hasil dari diagnosis serta *tabbed pane* untuk membuka halaman beranda, halaman pelatihan dan halaman pengujian. Perancangan antarmuka halaman diagnosis ditunjukkan pada Gambar 4.9.

Rancangan antarmuka halaman diagnosis yang menampilkan formulir diagnosis dengan daftar gejala pasien dan tombol diagnosis. Formulir ini dilengkapi dengan tabbed pane untuk beranda, pelatihan LVQ, pengujian, dan diagnosis.

Gejala 1	Gejala 9
-Pilih-	-Pilih-
Gejala 2	Gejala 10
-Pilih-	-Pilih-
Gejala 3	Gejala 11
-Pilih-	-Pilih-
Gejala 4	Gejala 12
-Pilih-	-Pilih-
Gejala 5	Gejala 13
-Pilih-	-Pilih-
Gejala 6	Gejala 14
-Pilih-	-Pilih-
Gejala 7	Gejala 15
-Pilih-	-Pilih-
Gejala 8	Diagnosis

Hasil Diagnosis :

Gambar 4.9 Rancangan Antarmuka Halaman Diagnosis



BAB 5 IMPLEMENTASI

Pada bab ini dibahas mengenai lingkungan implementasi, implementasi program, serta implementasi antarmuka untuk klasifikasi penyakit demam menggunakan metode Learning Vector Quantization (LVQ).

5.1 Lingkungan Implementasi

Lingkungan implementasi yang digunakan dalam membangun sistem untuk klasifikasi penyakit demam menggunakan metode *Learning vector Quantization* yaitu lingkungan perangkat keras dan lingkungan perangkat lunak.

5.1.1 Lingkungan Perangkat Keras

Dalam membangun sistem klasifikasi penyakit demam ini menggunakan beberapa spesifikasi perangkat keras yang ditunjukkan pada Tabel 5.1.

Tabel 5.1 Lingkungan Perangkat Keras

Perangkat Keras	Keterangan
<i>Processor Intel Core i7-5500U 2.40GHz</i>	Digunakan untuk memproses algoritma dalam aplikasi
<i>Memory RAM 4 GB</i>	Digunakan untuk menampung beban berjalannya aplikasi

5.1.2 Lingkungan Perangkat Lunak

Dalam membangun sistem klasifikasi penyakit gigi dan mulut ini menggunakan beberapa spesifikasi perangkat lunak yang ditunjukkan pada tabel 5.2.

Tabel 5.2 Lingkungan Perangkat Lunak

Perangkat Lunak	Keterangan
<i>Sistem Operasi Windows 8 Enterprise 64-bit</i>	Sebagai perantara aplikasi dengan hardware
<i>Bahasa Pemrograman Java</i>	Bahasa Pemrograman yang digunakan untuk membangun aplikasi penyakit demam menggunakan metode LVQ
<i>Netbeans IDE 8.0.2</i>	Aplikasi dan compiler yang digunakan untuk membangun aplikasi klasifikasi penyakit demam menggunakan metode LVQ



Tabel 5.2 Lingkungan Perangkat Lunak

Perangkat Lunak	Keterangan
<i>Notepad</i>	Digunakan untuk menyimpan data gejala yang digunakan sebagai data latih dan data uji
<i>Microsoft Word 2013</i>	Digunakan untuk pembuatan dokumen skripsi
<i>Microsoft Excel 2013</i>	Digunakan untuk pembuatan proses perhitungan manual menggunakan metode LVQ

5.2 Implementasi Program

Sub bab ini akan membahas tentang implementasi program yang digunakan dalam membangun sistem klasifikasi penyakit demam menggunakan metode *Learning Vector Quantization*. Implementasi program adalah implementasi dari algoritma LVQ yang terdiri dari penentuan bobot awal, pelatihan dan pengujian LVQ.

5.2.1 Penentuan Bobot Awal

Penentuan bobot awal bertujuan untuk menghasilkan bobot yang akan digunakan sebagai bobot awal dalam proses pelatihan LVQ. Bobot awal diambil secara acak dari data latih. Pengambilan bobot awal masing-masing satu data dari masing-masing kelas. Implementasi dari penentuan bobot awal ditunjukkan pada *Source Code 5.1*.

```

1 public void inisialisasi() {
2     int in1 = (int) (Math.random() * barislatih);
3     int in2 = (int) (Math.random() * barislatih);
4     int in3 = (int) (Math.random() * barislatih);
5     while (dl[in1][15] != 1) {
6         in1 = (int) (Math.random() * barislatih);
7     }
8     while (dl[in2][15] != 2) {
9         in2 = (int) (Math.random() * barislatih);
10    }
11    while (dl[in3][15] != 3) {
12        in3 = (int) (Math.random() * barislatih);
13    }
14    for (int j = 0; j < kolomlatih - 1; j++) {

```



```

15         bobotpertama[0][j] = dl[in1][j];
16         bobotpertama[1][j] = dl[in2][j];
17         bobotpertama[2][j] = dl[in3][j];
18     }
19     for (int i = 0; i < banyakkelas; i++) {
20         System.out.print("Bobot " + (i + 1) + " :");
21         for (int j = 0; j < kolomlatih - 1; j++) {
22             System.out.print(bobotpertama[i][j] + "\t");
23         }
24         System.out.println("");
25     }
26 }
```

Source Code 5.1 Penetuan Bobot Awal

Keterangan *Source Code 5.1* :

- Baris 1 : Sebuah kelas untuk inisialisasi bobot awal.
- Baris 2-4 : inisialisasi variabel dengan nilai acak dari barislatih .
- Baris 5-17 : perulangan untuk menentukan nilai dari bobot awal.
- Baris 19-25 : perulangan untuk menampilkan bobot awal.

5.2.2 Pelatihan dengan LVQ

Implementasi pelatihan ini bertujuan untuk menghasilkan bobot baru yang akan digunakan sebagai bobot awal pada proses pengujian. Implementasi dari pelatihan menggunakan metode LVQ ditunjukkan pada *Source Code 5.2*.

```

1 public void hitungjarakterdekat() {
2     double[] jarak = new double[banyakkelas];
3     double jml;
4     for (int i = 0; i < barislatih; i++) {
5         jml = 0;
6         jarak[0] = 0;
7         jarak[1] = 0;
8         jarak[2] = 0;
9
10        for (int j = 0; j < banyakkelas; j++) {
11            for (int k = 0; k < kolomlatih - 1; k++)
12            {
13                jml += (Math.pow(dl[i][k] -
14 bobotpertama[j][k], 2));
15            }
16            jarak[j] = Math.sqrt(jml);
17            jml = 0;
18        }
19        System.out.println("Hasil hitung jarak
```



```
20 antara bobot dengan data latih " + (i + 1));
21 System.out.println("Jarak " + (1) + " = " +
22 jarak[0]);
23 System.out.println("Jarak " + (2) + " = " +
24 jarak[1]);
25 System.out.println("Jarak " + (3) + " = " +
26 jarak[2]);
27
28
29 double jarakterkecil = jarak[0];
30 int t = (int) dl[i][kolomlatih - 1];
31 int c = 0;
32
33 for (int j = 1; j < banyakkelas; j++) {
34     if (jarakterkecil > jarak[j]) {
35         jarakterkecil = jarak[j];
36         c = j;
37     }
38 }
39 System.out.println("Jarak Terkecil : " +
40 jarakterkecil);
41 updateBobot(i, t, c); // i = data
42 }
43
44 alpha = alpha * decAlpha;
45
46 }
47
48
49 public void updateBobot(int d, int t, int c) {
50     if ((t - 1) == c) {
51         for (int i = 0; i < kolomlatih - 1; i++) {
52             bobotpertama[c][i] = bobotpertama[c][i]
53 + (alpha * (dl[d][i] - bobotpertama[c][i]));
54         }
55     } else {
56         for (int i = 0; i < kolomlatih - 1; i++) {
57             bobotpertama[c][i] = bobotpertama[c][i]
58 - (alpha * (dl[d][i] - bobotpertama[c][i]));
59         }
60     }
61     System.out.println("Bobot yg diupdate = " + (c +
62 1));
63     System.out.println("Hasil update bobot :");
64     for (int i = 0; i < bobotpertama.length; i++) {
65         System.out.printf("W[%d]\t", (i + 1));
66         for (int j = 0; j < bobotpertama[i].length;
67 j++) {
68             System.out.printf("%.4f\t",
69 bobotpertama[i][j]);
70         }
71         System.out.println("");
72     }
73 }
74
75 public void tampilBobot() {
76     System.out.println("BOBOT AWAL : ");
77     for (int i = 0; i < banyakkelas; i++) {
78         System.out.printf("W[%d]\t", (i + 1));
```

```

79             for (int j = 0; j < kolomlatih - 1; j++) {
80                 System.out.printf("%.4f\t",
81                 bobotpertama[i][j]);
82             }
83             System.out.println("");
84         }
}

```

Source Code 5.2 Pelatihan LVQ

Keterangan *Source Code 5.2*:

- Baris 1 : sebuah method untuk menghitung jarak terdekat data terhadap kelas.
- Baris 2-3 : inisialisasi dan deklarasi variabel.
- Baris 4-8 : perulangan untuk inisialisasi nilai variabel jarak.
- Baris 10-16 : perulangan untuk menghitung jarak terdekat.
- Baris 18-25 : perintah untuk menampilkan hasil perhitungan jarak.
- Baris 28-35 : perulangan untuk menentukan jarak terkecil.
- Baris 38-39 : perintah untuk menampilkan jarak terkecil.
- Baris 40 : perintah untuk menjalankan method untuk memperbarui nilai bobot.
- Baris 43 : memperbarui nilai alpha dengan mengalikan alpha dengan pengurang alpha.
- Baris 48 : sebuah method untuk perhitungan update bobot.
- Baris 49 : perintah berkondisi yang bertujuan untuk update bobot.
- Baris 60-70 : perintah untuk menampilkan bobot yang di update serta hasil update bobot dan bobot awal.
- Baris 74-82 : method untuk menampilkan bobot awal.

5.2.3 Pengujian dengan LVQ

Implementasi pengujian dengan LVQ bertujuan untuk mendapatkan hasil klasifikasi berdasarkan gejala penyakit demam yang diuji. Gejala-gejala tersebut akan diklasifikasikan ke dalam 3 kelas yaitu malaria, tifoid dan demam brdarah. Proses pengujian yaitu dengan menghitung jarak terdekat ke setiap kelasnya untuk mendapatkan hasil klasifikasi. Implementasi dari pengujian algoritma LVQ ditunjukkan pada *Source Code 5.3*.

```

1 public void pengujian() {
2     for (int i = 0; i < bobotakhir.length; i++) {
3         for (int j = 0; j < bobotakhir[i].length;
4             j++) {
5             bobotakhir[i][j] = bobotpertama[i][j];
6         }
}

```



```
7             }
8
9             System.out.println("=====");
10            =====");
11             System.out.println("Bobot Akhir : ");
12             for (int i = 0; i < bobotakhir.length; i++) {
13                 for (int j = 0; j < bobotakhir[i].length;
14                     j++) {
15                     System.out.printf("%.4f\t",
16 bobotakhir[i][j]);
17                 }
18                 System.out.println();
19             }
20
21             System.out.println("=====");
22            =====");
23             System.out.println("Pengujian");
24
25             System.out.println("=====");
26            =====");
27             System.out.println("Data Uji : ");
28             for (int i = 0; i < du.length; i++) {
29                 for (int j = 0; j < du[i].length; j++) {
30                     System.out.print(du[i][j] + " ");
31                 }
32                 System.out.println("");
33             }
34             double[] jarak = new double[banyakkelas];
35             double jml;
36             double databenar = 0;
37             for (int i = 0; i < barisuji; i++) //
38             {
39                 jml = 0;
40                 jarak[0] = 0;
41                 jarak[1] = 0;
42                 jarak[2] = 0;
43                 for (int j = 0; j < banyakkelas; j++) {
44                     for (int k = 0; k < kolomuji - 1; k++) {
45                         jml += (Math.pow(du[i][k] -
46 bobotpertama[j][k], 2));
47                     }
48                     jarak[j] = Math.sqrt(jml);
49                     jml = 0;
50                     //System.out.println("Jarak " + (j + 1) +
51                     " = " + jarak[j]);
52                 }
53                 System.out.println("Hasil hitung jarak antara
54 bobot dengan data uji " + (i + 1));
55                 System.out.println("Jarak " + (1) + " = " +
56 jarak[0]);
57                 System.out.println("Jarak " + (2) + " = " +
58 jarak[1]);
59                 System.out.println("Jarak " + (3) + " = " +
60 jarak[2]);
61
62                 double jarakterkecil = jarak[0];
63                 int t = (int) du[i][kolomuji - 1]; // kelas
64                 data 123
65                 int c = 1; // kelas
```

```
66 hasil sistem 012
67
68         for (int j = 1; j < banyakkelas; j++) {
69             if (jarakterkecil > jarak[j]) {
70                 jarakterkecil = jarak[j];
71                 c = j + 1;
72             }
73         }
74         System.out.println("Jarak Terkecil : " +
75 jarakterkecil);
76
77         System.out.println("Kelas hasil perhitungan
78 sistem = " + (c));
79         System.out.println("Kelas sesungguhnya = " +
80 t);
81         if (c == t) {
82             databenar++;
83         }
84     }
85
86     System.out.println("Banyak data benar : " +
87 databenar);
88     System.out.println("Banyak data uji : " +
89 du.length);
90     double akurasi = (databenar / du.length) * 100;
91     System.out.print("Hasil Akurasi : ");
92     System.out.printf("%.2f\t", akurasi);
}
```

Source Code 5.3 Pengujian LVQ

Keterangan *Source Code 5.3*:

- Baris 1-7 : Sebuah method untuk perhitungan pengujian. Dan perulangan untuk inisialisasi nilai bobot akhir.
- Baris 9-18 : perintah untuk menampilkan bobot akhir yang akan digunakan dalam pengujian.
- Baris 20-32 : perintah untuk menampilkan teks dan data uji yang akan digunakan.
- Baris 34-36 : inisialisasi nilai jarak dan deklarasi variabel *jml* serta variabel *databenar*.
- Baris 37-52 : perulangan untuk menghitung jarak pada data uji.
- Baris 53-60 : perintah untuk menampilkan hasil hitung jarak.
- Baris 62-66 : inisialisasi variabel.
- Baris 68-71 : Perulangan berkondisi untuk menentukan jarak terkecil.
- Baris 74-75 : perintah untuk menampilkan jarak terkecil.
- Baris 77-80 : perintah untuk menampilkan kelas hasil perhitungan sistem dan kelas sesungguhnya.



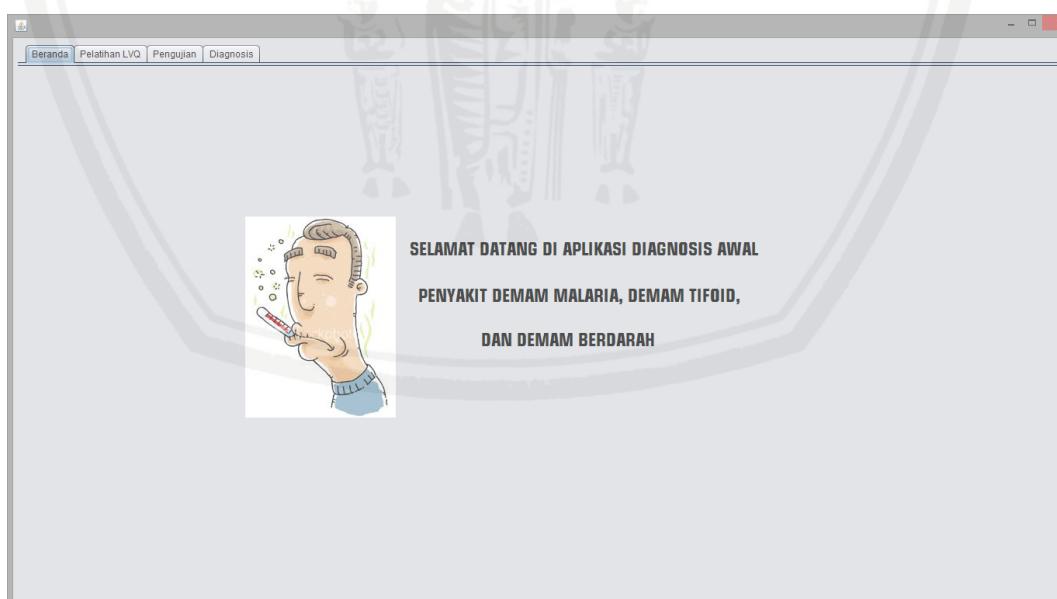
- Baris 81 : jika variabel c dan t atau kelas hasil perhitungan sistem sama dengan kelas sesungguhnya, maka jumlah nilai variabel data benar akan bertambah 1.
- Baris 86-89 : perintah untuk menampilkan banyak data benar dan banyak data uji.
- Baris 90-92 : perhitungan untuk menghitung akurasi serta perintah untuk menampilkan hasil perhitungan akurasi.

5.3 Implementasi Antarmuka

Sub bab ini akan membahas tentang implementasi hasil rancangan antarmuka pada bab sebelumnya untuk klasifikasi penyakit demam. Implementasi antarmuka terdiri dari tampilan halaman beranda, halaman pelatihan LVQ, halaman pengujian, dan halaman diagnosis.

5.3.1 Halaman Beranda

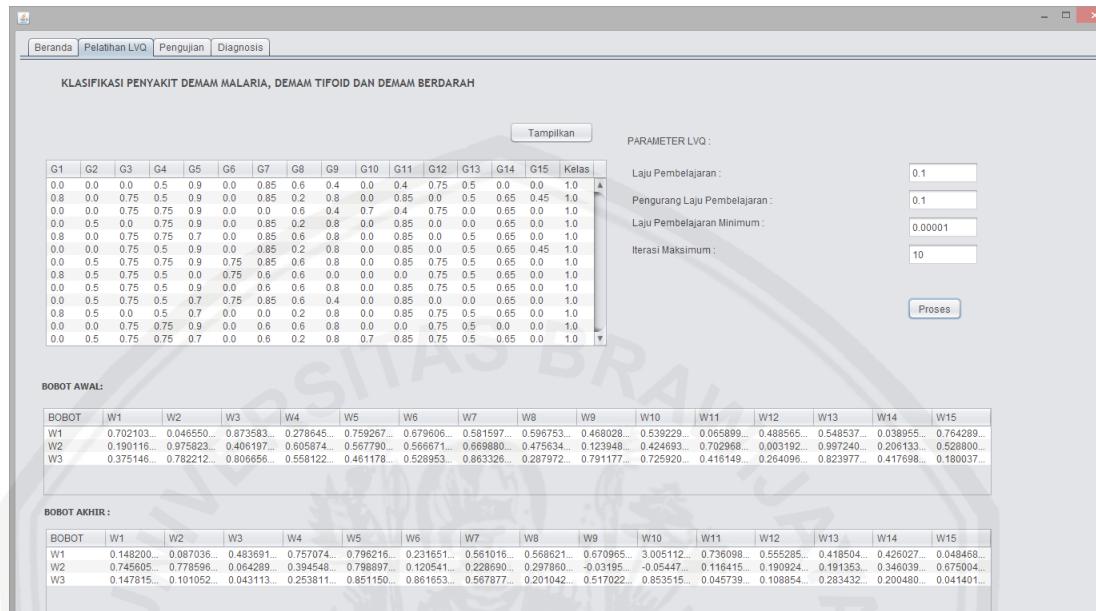
Halaman beranda merupakan halaman yang pertama kali muncul pada saat aplikasi dijalankan. Pada halaman ini pengguna dapat melihat halaman-halaman lain yang dapat diakses. Selain itu pada halaman beranda ini terdapat beberapa informasi mengenai pembuatan sistem yaitu berisi gambar *icon* kartun laki-laki yang sedang demam serta berisi informasi tentang aplikasi klasifikasi penyakit demam menggunakan metode LVQ. Implementasi antarmuka halaman beranda ditunjukkan pada Gambar 5.1.



Gambar 5.1 Implementasi Antarmuka Beranda

5.3.2 Halaman Pelatihan

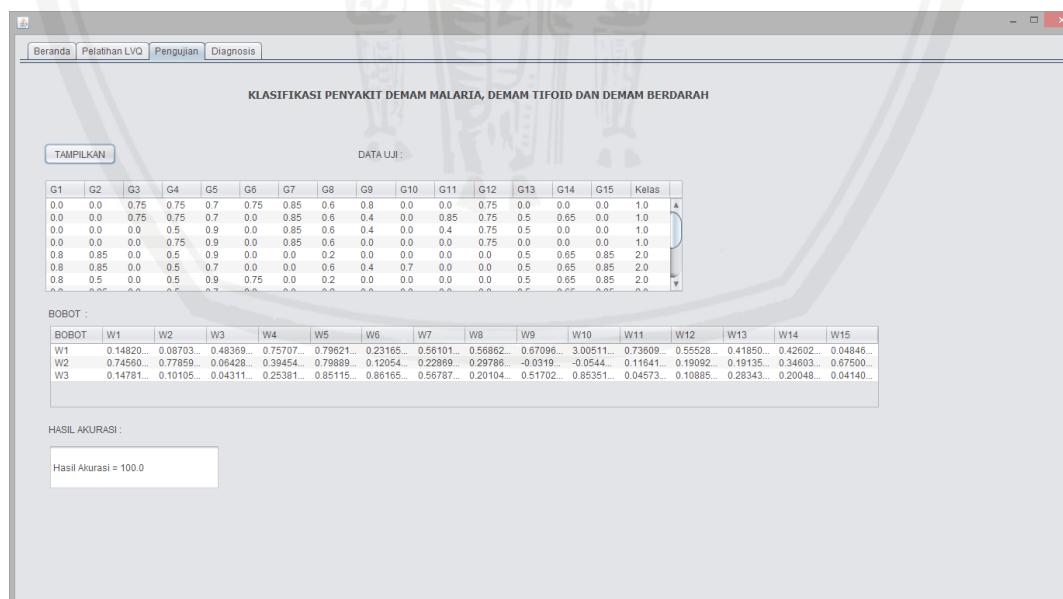
Pada halaman pelatihan LVQ akan ditampilkan hasil proses pelatihan menggunakan LVQ yaitu berupa tampilan tabel bobot awal, bobot baru, data latih dan parameter LVQ yang digunakan untuk proses pelatihan serta button yang digunakan untuk mengambil data latih dari komputer. Implementasi antarmuka halaman pelatihan LVQ ditunjukkan pada Gambar 5.2.



Gambar 5.2 Implementasi Antarmuka Pelatihan LVQ

5.3.3 Halaman Pengujian

Implementasi antarmuka pengujian ditunjukkan pada Gambar 5.3.



Gambar 5.3 Implementasi Antarmuka Pengujian



Halaman pengujian merupakan halaman untuk menampilkan hasil akurasi pengujian terhadap beberapa data uji yang dimasukkan. Pada halaman pengujian terdapat button yang digunakan untuk mengambil data yang digunakan untuk proses pengujian serta menampilkan tabel akurasi hasil dari pengujian yang dilakukan. Proses pengujian dilakukan dengan mencari jarak ke setiap kelas dan menentukan jarak terdekat untuk mengetahui kelas dari data uji tersebut. Proses pengujian ini dilakukan untuk setiap data uji yang tersimpan dalam file txt.

5.3.4 Halaman Diagnosis

Pada halaman diagnosis ditampilkan proses diagnosis yaitu pengguna memilih gejala-gejala yang tersedia dan nantinya akan diperoleh hasil klasifikasi. Pada halaman ini terdapat pilihan-pilihan dari gejala penyakit demam serta hasil dari klasifikasi berdasarkan gejala-gejala yang dipilih pengguna. Implementasi antarmuka diagnosis ditunjukkan pada Gambar 5.4.

The screenshot shows a Windows application window titled "KLASIFIKASI PENYAKIT DEMAM MALARIA, DEMAM TIFOID DAN DEMAM BERDARAH". The window has a menu bar with "Beranda", "Pelatihan LVO", "Pengujian", and "Diagnosis". The main area is a form titled "GEJALA PENYAKIT YANG DISERTAI DENGAN". It lists 15 symptoms (G1-G15) each with a dropdown menu for selection. The symptoms are: 1. Demam Intermittent (putus-putus) (G1), 2. Demam menggigil (G2), 3. Demam terutama malam hari (G3), 4. Lama demam (G4), 5. Sakit kepala (G5), 6. Sakit tulang dan sendi (G6), 7. Mual dan muntah (G7), 8. Mencoret atau susah BAB(konstipasi) (G8), 9. Nyeri perut (G9), 10. Bintik merah (petekie) pada kulit (G10), 11. Lidah kotor (coated tongue) (G11), 12. Bradikardi relatif (G12), 13. Pembesaran hati (G13), 14. Pembesaran limpa (G14), and 15. Kulit lembab / Keringat (G15). A "DIAGNOSIS" button is located at the bottom right of the form. Below the form is a section titled "HASIL DIAGNOSIS" containing a text box with the text "Hasil Diagnosis Adalah Demam Berdarah".

Gambar 5.4 Implementasi Antarmuka Diagnosis



BAB 6 PENGUJIAN DAN ANALISIS

Pada bab ini dibahas mengenai hasil uji coba yang telah dilakukan dalam klasifikasi penyakit demam menggunakan metode *Learning Vector Quantization* (LVQ). Pengujian dilakukan bertujuan untuk mengetahui parameter LVQ yang paling optimal agar dapat menghasilkan akurasi terbaik. Pengujian yang dilakukan yaitu uji coba nilai laju pembelajaran, nilai pengurang laju pembelajaran, laju pembelajaran minimum, iterasi maksimum, perbandingan jumlah data latih dan data uji serta pengujian akurasi.

6.1 Hasil dan Analisis Uji Coba untuk Mengetahui Pengaruh Nilai Laju Pembelajaran terhadap Hasil Akurasi

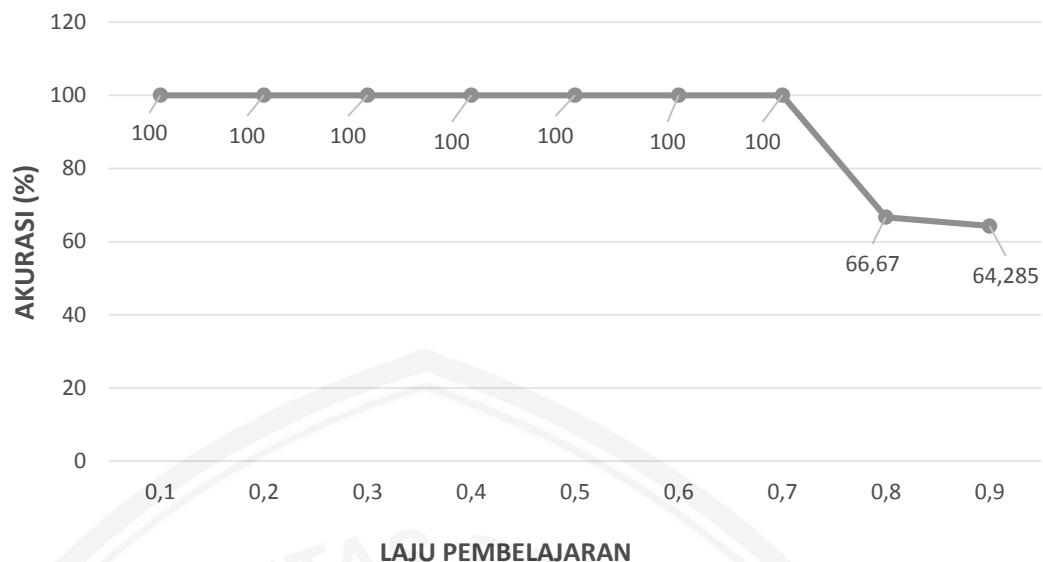
Pada pengujian ini dilakukan uji coba sebanyak 10 kali untuk mendapatkan hasil akurasi terbaik pada masing-masing nilai laju pembelajaran yang digunakan. Nilai laju pembelajaran yang diujikan adalah 0,1, 0,2, 0,3, 0,4, 0,5, 0,6, 0,7, 0,8, dan 0,9. Dalam pengujian ini data latih yang digunakan adalah 112 data latih dan 21 data uji. Terdapat 3 kelas yaitu kelas 1 demam tifoid, kelas 2 demam malaria, dan kelas 3 demam berdarah. Parameter LVQ yang digunakan dalam uji coba ini adalah nilai konstanta pengurang laju pembelajaran = 0,5, laju pembelajaran minimum 10^{-11} dan iterasi maksimum = 20. Sedangkan untuk bobot awal yang digunakan adalah data latih yang diambil acak sesuai kelas masing-masing. Hasil uji coba pengaruh nilai laju pembelajaran ditunjukkan pada Tabel 6.1.

Tabel 6.1 Hasil Uji Coba Pengaruh Nilai Laju Pembelajaran Terhadap Akurasi

Laju Pembela jaran	Percobaan Ke-										Rata- Rata Akuras i (%)
	1	2	3	4	5	6	7	8	9	10	
0,1	100	100	100	100	100	100	100	100	100	100	100
0,2	100	100	100	100	100	100	100	100	100	100	100
0,3	100	100	100	100	100	100	100	100	100	100	100
0,4	100	100	100	100	100	100	100	100	100	100	100
0,5	100	100	100	100	100	100	100	100	100	100	100
0,6	100	100	100	100	100	100	100	100	100	100	100
0,7	100	100	100	100	100	100	100	100	100	100	100
0,8	66,67	66,67	66,67	66,67	66,67	66,67	66,67	66,67	66,67	66,67	66,67
0,9	66,67	61,90	61,90	66,67	61,90	66,67	66,67	61,90	66,67	61,90	64,285

Berdasarkan hasil uji coba pada Tabel 6.1, diperoleh grafik yang ditunjukkan pada Gambar 6.1.





Gambar 6.1 Grafik Uji Coba Pengaruh Nilai Laju Pembelajaran Terhadap Akurasi

Dari grafik hasil uji coba terhadap laju pembelajaran dapat dilihat bahwa laju pembelajaran berpengaruh terhadap akurasi yang dihasilkan. Berdasarkan hasil pengujian, rata-rata tertinggi yang didapatkan adalah 100 % yaitu pada nilai pembelajaran 0,1 sampai 0,7. Sedangkan rata-rata akurasi terendah terdapat pada nilai laju pembelajaran 0,9 yaitu 64,28%. Dapat disimpulkan bahwa semakin besar nilai laju pembelajaran, semakin rendah akurasi yang dihasilkan. Hal ini karena nilai laju pembelajaran berpengaruh terhadap pergeseran nilai bobot pada proses pelatihan LVQ. Sehingga ketika nilai laju pembelajaran semakin besar, maka langkah pembelajaran yang dilakukan juga semakin besar dan mengakibatkan sulitnya mencapai konvergensi pada perubahan bobot yang dihasilkan.

6.2 Hasil dan Analisis Uji Coba untuk Mengetahui Pengaruh Nilai Konstanta Pengurang Laju Pembelajaran terhadap Hasil Akurasi

Pada pengujian ini dilakukan uji coba sebanyak 10 kali untuk mendapatkan hasil akurasi terbaik pada masing-masing nilai konstanta pengurang laju pembelajaran yang digunakan. Nilai konstanta pengurang laju pembelajaran yang diujikan adalah 0,1, 0,2, 0,3, 0,4, 0,5, 0,6, 0,7, 0,8, dan 0,9. Dalam pengujian ini data latih yang digunakan adalah 112 data latih dan 21 data uji. Terdapat 3 kelas yaitu kelas 1 demam tifoid, kelas 2 demam malaria, dan kelas 3 demam berdarah. Parameter LVQ yang digunakan dalam uji coba ini adalah nilai laju pembelajaran = 0,1, laju pembelajaran minimum = 10^{-11} dan iterasi maksimum = 20. Sedangkan untuk bobot awal yang digunakan adalah data latih yang diambil

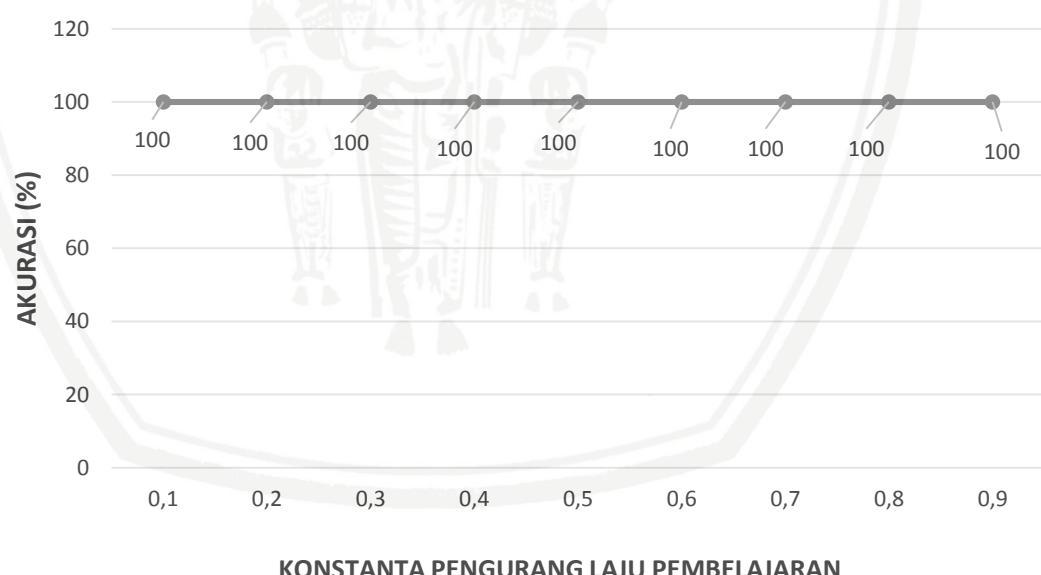


acak sesuai kelas masing-masing. Hasil uji coba pengaruh nilai konstanta pengurang laju pembelajaran ditunjukkan pada Tabel 6.2.

Tabel 6.2 Hasil Uji Coba Pengaruh Nilai konstanta Pengurang Laju Pembelajaran Terhadap Akurasi

Pengurang g Laju Pembelaj aran	Percobaan Ke-										Rata-Rata Akurasi (%)
	1	2	3	4	5	6	7	8	9	10	
0,1	100	100	100	100	100	100	100	100	100	100	100
0,2	100	100	100	100	100	100	100	100	100	100	100
0,3	100	100	100	100	100	100	100	100	100	100	100
0,4	100	100	100	100	100	100	100	100	100	100	100
0,5	100	100	100	100	100	100	100	100	100	100	100
0,6	100	100	100	100	100	100	100	100	100	100	100
0,7	100	100	100	100	100	100	100	100	100	100	100
0,8	100	100	100	100	100	100	100	100	100	100	100
0,9	100	100	100	100	100	100	100	100	100	100	100

Berdasarkan hasil uji coba pada Tabel 6.2, diperoleh grafik yang ditunjukkan pada Gambar 6.2.



Gambar 6.2 Grafik Uji Coba Pengaruh Nilai Konstanta Pengurang Laju Pembelajaran Terhadap Akurasi

Berdasarkan hasil uji coba pengaruh nilai konstanta pengurang laju pembelajaran didapatkan rata-rata akurasi tertinggi sebesar 100% yaitu pada



semua nilai konstanta pengurang laju pembelajaran yang di uji. Nilai konstanta pengurang laju pembelajaran memiliki pengaruh untuk memperkecil laju pembelajaran pada proses pelatihan LVQ sehingga dapat mencapai bobot yang ideal. Berdasarkan hasil uji coba pengaruh konstanta pengurang laju pembelajaran terhadap akurasi bahwa nilai akurasi tetap stabil dan mampu mencapai konvergensi perubahan bobot yang dihasilkan sejak uji coba pertama menggunakan konstanta pengurang laju pembelajaran bernilai 0,1.

6.3 Hasil dan Analisis Uji Coba untuk Mengetahui Pengaruh Nilai Laju Pembelajaran Minimum terhadap Hasil Akurasi

Pada pengujian ini dilakukan uji coba sebanyak 10 kali untuk mendapatkan hasil akurasi terbaik pada masing-masing nilai laju pembelajaran minimum yang digunakan. Nilai laju pembelajaran minimum yang diujikan adalah 10^{-2} , 10^{-3} , 10^{-4} , 10^{-5} , 10^{-6} , 10^{-7} , 10^{-8} , 10^{-9} , 10^{-10} , dan 10^{-11} . Dalam pengujian ini data latih yang digunakan adalah 112 data latih dan 21 data uji. Terdapat 3 kelas yaitu kelas 1 demam tifoid, kelas 2 demam malaria, dan kelas 3 demam berdarah. Parameter LVQ yang digunakan dalam uji coba ini adalah nilai laju pembelajaran = 0,1, konstanta pengurang laju pembelajaran = 0,1, dan iterasi maksimum = 20. Sedangkan untuk bobot awal yang digunakan adalah data latih yang diambil acak sesuai kelas masing-masing. Hasil uji coba pengaruh nilai konstanta pengurang laju pembelajaran ditunjukkan pada Tabel 6.3.

Tabel 6.3 Hasil Uji Coba Pengaruh Nilai Laju Pembelajaran Minimum Terhadap Akurasi

Laju Pembelajaran Minimum	Percobaan Ke-										Rata- Rata Akurasi (%)
	1	2	3	4	5	6	7	8	9	10	
10^{-2}	100	100	100	100	100	100	100	100	100	100	100
10^{-3}	100	100	100	100	100	100	100	100	100	100	100
10^{-4}	100	100	100	100	100	100	100	100	100	100	100
10^{-5}	100	100	100	100	100	100	100	100	100	100	100
10^{-6}	100	100	100	100	100	100	100	100	100	100	100
10^{-7}	100	100	100	100	100	100	100	100	100	100	100
10^{-8}	100	100	100	100	100	100	100	100	100	100	100
10^{-9}	100	100	100	100	100	100	100	100	100	100	100
10^{-10}	100	100	100	100	100	100	100	100	100	100	100
10^{-11}	100	100	100	100	100	100	100	100	100	100	100



Berdasarkan hasil uji coba pada Tabel 6.3, diperoleh grafik yang ditunjukkan pada Gambar 6.3.



Gambar 6.3 Grafik Uji Coba Pengaruh Laju Pembelajaran Minimum Terhadap Akurasi

Berdasarkan hasil uji coba pengaruh laju pembelajaran minimum didapatkan rata-rata akurasi tertinggi sebesar 100% yaitu pada semua nilai laju pembelajaran minimum yang di uji. Dari grafik hasil uji coba pada ambar 6.3 didapatkan hasil bahwa pengujian pengaruh nilai laju pembelajaran minimum menunjukkan rata-rata akurasi yang tetap. Laju pembelajaran minimum adalah salah satu kondisi berhenti dalam proses pelatihan LVQ. Nilai laju pembelajaran minimum ini mempengaruhi proses konvergensi pada perubahan bobot. Semakin kecil nilai laju pembelajaran minimum maka akan memperpanjang proses konvergensi bobot sehingga proses konvergensi bobot semakin lama. Dari pengujian ini dapat diketahui bahwa proses perubahan bobot telah mencapai konvergensi dengan cepat pada saat nilai laju pembelajaran minimum 10^{-2} karena sudah mencapai akurasi tertinggi.

6.4 Hasil dan Analisis Uji Coba untuk Mengetahui Pengaruh Jumlah Iterasi Maksimum terhadap Hasil Akurasi

Pada pengujian ini dilakukan uji coba sebanyak 10 kali untuk mendapatkan hasil akurasi terbaik pada masing-masing nilai iterasi yang digunakan. Nilai iterasi yang diujikan adalah 2, 10, 20, 32, 46, 62, 80, 100, 122, 146. Dalam pengujian ini data latih yang digunakan adalah 112 data latih dan 21 data uji. Terdapat 3 kelas yaitu kelas 1 demam tifoid, kelas 2 demam malaria, dan kelas 3 demam berdarah. Parameter LVQ yang digunakan dalam uji coba ini adalah nilai

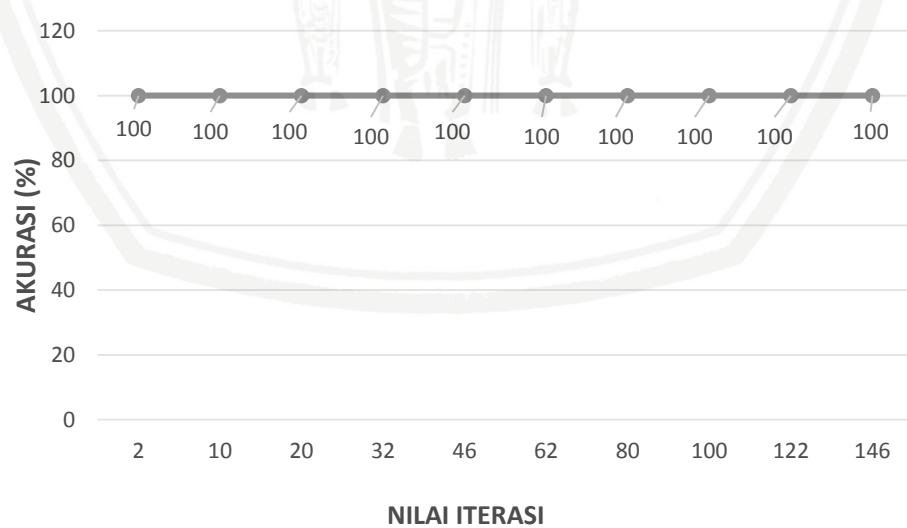


laju pembelajaran = 0.1, dan nilai konstanta pengurang laju pembelajaran = 0,1 serta laju pembelajaran minimum 10^{-5} . Sedangkan untuk bobot awal yang digunakan adalah data latih yang diambil acak sesuai kelas masing-masing. Hasil uji coba pengaruh nilai iterasi ditunjukkan pada Tabel 6.4.

Tabel 6.4 Hasil Uji Coba Pengaruh Nilai Iterasi Terhadap Akurasi

Jumlah Iterasi	Percobaan Ke-										Rata-Rata Akurasi (%)
	1	2	3	4	5	6	7	8	9	10	
2	100	100	100	100	100	100	100	100	100	100	100
10	100	100	100	100	100	100	100	100	100	100	100
20	100	100	100	100	100	100	100	100	100	100	100
32	100	100	100	100	100	100	100	100	100	100	100
46	100	100	100	100	100	100	100	100	100	100	100
62	100	100	100	100	100	100	100	100	100	100	100
80	100	100	100	100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100	100	100	100	100
122	100	100	100	100	100	100	100	100	100	100	100
146	100	100	100	100	100	100	100	100	100	100	100

Berdasarkan hasil uji coba pada Tabel 6.4, diperoleh grafik yang ditunjukkan pada Gambar 6.4.



Gambar 6.4 Grafik Uji Coba Pengaruh Nilai Iterasi Terhadap Akurasi



Pada tahap pengujian, nilai maksimum *epoch* atau maksimum perulangan iterasi berpengaruh karena merupakan salah satu tahap berhentinya suatu proses pelatihan. Dari grafik gambar 6.4 diperoleh hasil akurasi 100% dengan nilai iterasi 2, 10, 20, 32, 46, 62, 80, 100, 122, dan 146. Sehingga dapat disimpulkan bahwa semua iterasi menghasilkan hasil akurasi yang sama dan stabil. Hal ini bisa terjadi dikarenakan pada proses sebelumnya telah diambil nilai laju pembelajaran dan konstanta pengurang laju pembelajaran terbaik yang menyebabkan konvergen sejak iterasi pertama. Berdasarkan hasil tersebut, pada nilai iterasi terkecil yang diinputkan yaitu 2 telah mencapai akurasi maksimum.

6.5 Hasil dan Analisis Uji Coba untuk Mengetahui Pengaruh Perbandingan Jumlah Data Latih dan Data Uji terhadap Hasil Akurasi

Pada pengujian ini dilakukan percobaan sebanyak sepuluh kali untuk mendapatkan hasil akurasi yang terbaik pada masing-masing perbandingan jumlah data latih dan data uji yang digunakan. Uji coba ini menggunakan perbandingan data latih dan data uji dengan rasio 90:10, 80:20, 70:30, 60:40, 50:50, 40:60, 30:70, 20:80 dan 10:90. Parameter LVQ yang digunakan dalam uji coba ini yaitu nilai laju pembelajaran 0,1, laju pembelajaran minimum 10^{-5} , konstanta pengurang laju pembelajaran = 0,1, dan jumlah iterasi maksimum 10. Hasil uji coba untuk mengetahui pengaruh perbandingan jumlah data latih dan data uji terhadap hasil akurasi ditunjukkan pada Tabel 6.5.

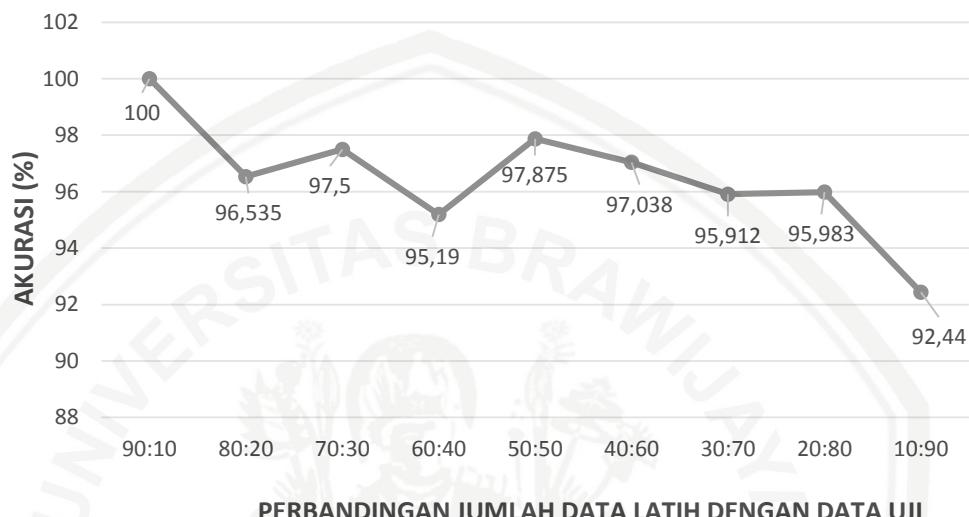
Tabel 6.5 Hasil Uji Coba Pengaruh Perbandingan Jumlah Data Latih dan Data Uji Terhadap Akurasi

Perbandingan Data Latih dan Data Uji	Percobaan Ke-										Rata-Rata Akurasi (%)
	1	2	3	4	5	6	7	8	9	10	
90:10	100	100	100	100	100	100	100	100	100	100	100
80:20	96,15	100	96,15	96,15	96,15	96,15	96,15	96,15	96,15	96,15	96,535
70:30	97,50	97,50	97,50	97,50	97,50	97,50	97,50	97,50	97,50	97,50	97,50
60:40	96,15	94,23	96,15	94,23	94,23	94,23	96,15	96,15	94,23	96,15	95,19
50:50	96,97	98,48	95,45	96,97	98,48	98,48	98,48	98,48	98,48	98,48	97,875
40:60	96,30	97,53	97,53	97,53	97,53	96,30	97,53	96,30	98,77	95,06	97,038
30:70	96,7	96,7	95,7	96,7	95,7	94,6	96,7	94,6	95,7	95,7	95,91



	7	7	0	7	0	2	7	2	0	0	2
20:80	97,2 0	95,3 3	95,3 3	97,2 0	96,2 6	95,3 3	96,2 6	95,3 3	95,3 3	96,2 6	95,98 3
10:90	92,4 4	93,2 8	94,1 2	94,1 2	91,6 0	89,9 2	93,2 8	91,6 0	91,6 0	92,4 4	92,44

Berdasarkan hasil uji coba pada Tabel 6.5, diperoleh grafik yang ditunjukkan pada Gambar 6.5.



Gambar 6.5 Grafik Uji Coba Pengaruh Perbandingan Jumlah Data Latih dengan Data Uji Terhadap Akurasi

Berdasarkan hasil uji coba pengaruh perbandingan jumlah data latih dengan data uji didapatkan rata-rata akurasi tertinggi sebesar 100% yaitu pada nilai perbandingan 90:10. Sedangkan rata-rata terendah pada nilai perbandingan 10:90 yaitu 92,44%. Pada Gambar 6.5 secara keseluruhan menunjukkan penurunan terhadap akurasi yang didapatkan. Dari pengujian ini diketahui bahwa semakin banyak data latih maka akurasi yang didapatkan semakin tinggi, sedangkan semakin banyak data uji dibandingkan data latih maka akurasi yang didapatkan akan semakin rendah. Semakin banyak data latih maka variasi data latih yang digunakan untuk proses pelatihan pada LVQ juga semakin banyak sehingga bisa mendapatkan akurasi tertinggi.

6.6 Hasil dan Analisis Pengujian 1 untuk Mengetahui Akurasi Berdasarkan Data Uji yang Dimasukkan Pengguna

Pengujian ini dilakukan untuk mengetahui akurasi berdasarkan data uji yang dimasukkan oleh pengguna pada file txt. Parameter yang digunakan dalam pengujian ini adalah parameter-parameter hasil uji sebelumnya yang akurasinya terbaik. Parameter tersebut meliputi nilai laju pembelajaran = 0,1, konstanta



pengurang laju pembelajaran = 0,1 dan laju pembelajaran minimum = 10^{-5} serta jumlah iterasi maksimum = 10. Perbandingan data latih dan data uji yang akan digunakan adalah 90:10.

Setelah proses pengujian dilakukan, maka diperoleh bobot awal yaitu :

w1	0.8	0.5	0.0	0.75	0.9	0.0	0.85	0.6	0.0	0.0	0.0	0.75	0.0	0.0	0.0	0.0
w2	0.0	0.85	0.0	0.5	0.7	0.0	0.0	0.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.4
w3	0.0	0.0	0.0	0.2	0.9	0.9	0.85	0.2	0.4	1.0	0.0	0.0	0.0	0.0	0.0	0.0

Sedangkan bobot baru dari hasil pelatihan yang akan digunakan untuk menghitung jarak data uji ke masing-masing kelas yaitu :

w 1	0.1 993	0.1 426	0.4 338	0.7 128	0.8 085	0.2 098	0.5 974	0.5 179	0.5 827	- 0.00 15	0.6 507	0.5 836	0.3 365	0.4 136	0.0 544
w 2	0.6 631	0.7 284	0.0 756	0.3 911	0.7 872	0.1 642	0.1 939	0.2 706	0.0 129	0.03 18	0.0 902	0.1 540	0.1 914	0.3 913	0.6 202
w 3	0.1 109	0.0 898	0.0 390	0.2 557	0.8 544	0.8 646	0.5 829	0.2 000	0.5 162	0.84 21	0.0 549	0.1 102	0.2 579	0.1 929	0.0 379

Berdasarkan bobot awal dan bobot baru yang didapatkan, maka hasil uji dapat ditunjukkan pada Tabel 6.6.

Tabel 6.6 Data Pengujian

Gejala															Kel as Aw al	Kel as Ba ru
G 1	G 2	G 3	G 4	G 5	G 6	G 7	G 8	G 9	G 10	G 11	G 12	G 13	G 14	G 15		
0	0	0,7 5	0,7 5	0,7 5	0,7 5	0,85	0,6	0,8	0	0	0,75	0	0	0	1	1
0	0	0,7 5	0,7 5	0,7 5	0	0,85	0,6	0,4	0	0,8 5	0,75	0,5	0,65	0	1	1
0	0	0	0,5	0,9	0	0,85	0,6	0,4	0	0,4	0,75	0,5	0	0	1	1
0	0	0	0,7 5	0,9	0	0,85	0,6	0	0	0	0,75	0	0	0	1	1
0, 8	0,8 5	0	0,5	0,9	0	0	0,2	0	0	0	0	0,5	0,65	0,85	2	2
0, 8	0,8 5	0	0,5	0,7	0	0	0,6	0,4	0,7	0	0	0,5	0,65	0,85	2	2
0, 8	0,5	0	0,5	0,9	0,7 5	0	0,2	0	0	0	0	0,5	0,65	0,85	2	2
0, 8	0,8 5	0	0,5	0,7	0	0	0,2	0	0	0	0	0,5	0,65	0,85	2	2
0	0	0	0,2	0,7	0,9	0	0,2	0,4	0,7	0	0	0	0	0	3	3
0	0	0	0,5	0,9	0,7 5	0,85	0,2	0,8	1	0	0	0,5	0,65	0	3	3
0	0	0	0,5	0,9	0,9	0,85	0,2	0	1	0	0	0	0,65	0	3	3
0	0	0	0,2	0,9	0,9	0,85	0,2	0	1	0	0	0	0	0	3	3
0	0	0	0,2	0,9	0,9	0,85	0,2	0	1	0	0	0	0	0	3	3

Dari data Tabel 6.6 dapat diketahui bahwa hasil pengujian dari data uji menghasilkan kelas baru yang sama dengan kelas awal. Akurasi yang diperoleh



adalah 100%. Dapat disimpulkan bahwa parameter-parameter LVQ yang digunakan dari hasil uji coba serta jumlah data latih dan data uji merupakan parameter yang terbaik karena menghasilkan akurasi yang tinggi.

6.7 Hasil dan Analisis Pengujian 2 untuk Mengetahui Hasil Klasifikasi dari Gejala-Gejala yang dipilih Pengguna

Pengujian ini dilakukan untuk mengetahui hasil klasifikasi dari gejala-gejala penyakit demam tifoid, demam malaria atau demam berdarah yang dialami pengguna. Pengguna dapat memilih gejala-gejala penyakit yang tersedia. Sistem akan melakukan perhitungan untuk menghasilkan klasifikasi penyakit menggunakan metode LVQ berdasarkan gejala-gejala yang dipilih oleh pengguna. Gejala-gejala tersebut disesuaikan dengan salah satu data yang sudah ada untuk menguji keakuratan sistem. Data yang digunakan yaitu data dari penyakit demam berdarah. Parameter yang digunakan meliputi nilai laju pembelajaran = 0,1, konstanta pengurang laju pembelajaran = 0,1 dan laju pembelajaran minimum = 10^{-5} serta jumlah iterasi maksimum = 10.

Pengujian yang dilakukan adalah sebagai berikut :

- Pengguna memilih gejala penyakit demam. Gejala yang dipilih pengguna pada pengujian ini yaitu demam putus-putus, demam menggil sedang, lama demam 1-3 hari, sakit kepala berat, sakit tulang dan sendi berat, nyeri perut sedang, bintik merah pada kulit berat, dan pembesaran limpa.
- Berdasarkan gejala-gejala yang dipilih oleh pengguna maka dihasilkan jarak seperti pada Tabel 6.7.

Tabel 6.7 Jarak Pada Pengujian 2

Jarak ke kelas 1	Jarak ke kelas 2	Jarak ke kelas 3	Jarak Terdekat
2.2910	1.6490	1.2417	1.2417

- Berdasarkan Tabel 6.7 hasil perhitungan jarak ke masing-masing kelas, jarak yang terdekat adalah jarak terhadap kelas 3 yaitu 1,2417. Jadi dapat diketahui bahwa hasil klasifikasi berdasarkan gejala-gejala yang dipilih oleh pengguna yaitu masuk ke kelas 3 (demam berdarah).



BAB 7 PENUTUP

Pada bab ini dibahas mengenai kesimpulan dari penelitian ini dan saran untuk penelitian selanjutnya.

7.1 Kesimpulan

Berdasarkan hasil analisis pengujian yang telah dilakukan dalam penelitian ini dapat diambil kesimpulan yaitu :

1. Implementasi algoritma *Learning Vector Quantization* (LVQ) untuk klasifikasi penyakit demam tifoid, demam malaria dan demam berdarah yaitu dengan tahapan-tahapan pengambilan data-data gejala dari masing-masing penyakit kemudian melakukan pelatihan dan pengujian menggunakan algoritma LVQ. Pelatihan pada LVQ akan menghasilkan bobot baru yang akan digunakan sebagai bobot pada proses pengujian data uji sehingga menghasilkan klasifikasi penyakit berdasarkan gejala-gejala penyakit pada data uji.
2. Pada proses pengujian didapatkan hasil rata-rata akurasi terbaik yaitu 100% dengan menggunakan parameter laju pembelajaran = 0,1, konstanta pengurang laju pembelajaran = 0,1, laju pembelajaran minimum = 10^{-5} , dan jumlah iterasi maksimum = 10. Perbandingan data latih dan data uji yang digunakan adalah 90:10.

7.2 Saran

Dalam penelitian ini penulis masih merasa banyak kekurangan. Oleh karena itu penulis memberikan saran antara lain :

1. Menambahkan penyakit demam yang di diagnosa. Hal ini karena penyakit yang disertai demam yang dialami oleh manusia sangat banyak. Penambahan penyakit dimaksudkan agar sistem yang dibuat dapat mencakup penyakit demam yang ada.
2. Menambahkan dataset yang digunakan agar lebih variatif sehingga akurasi yang diperoleh dari proses pelatihan menggunakan LVQ lebih akurat.



DAFTAR PUSTAKA

- Annisa, C. D., 2016. *Sistem Pakar diagnosa awal penyakit DBD, Malaria dan Tifoid menggunakan metode Fuzzy K-Nearest Neighbor (FKNN)*. S1. Universitas Brawijaya.
- Arifianto, AJ., Sarosa, M., Setyawati, O., 2014. *Klasifikasi Stroke Berdasarkan Kelainan Patologis dengan Learning Vector Quantization*. Jurnal EECCIS Vol.8 No.2.
- Arifien, Z., 2016. *Pendeteksi jenis Attention Deficit Hyperactivity Disorder (ADHD) pada anak usia dini menggunakan metode Learning Vector Quantization (LVQ)*. S1. Universitas Brawijaya.
- Departemen Kesehatan Republik Indonesia, 2007. *Pedoman Pengobatan Dasar di Puskesmas*. Tersedia di <<http://dokumen.tips/documents/pedoman-pengobatan-dasar-di-puskesmas.html>> (Diakses tanggal 20 Maret 2018)
- Departemen Kesehatan Republik Indonesia, 2008. *Perkembangan Kejadian DBD di Indonesia 2004-2007*. Tersedia di <<http://www.penyakitmenular.info/detil.asp?m=5&s=5&i=217>> (Diakses tanggal 20 Maret 2018)
- Eliyen, K., Tolle, H., Muslim, MA., 2017. *Learning Vector Quantization Untuk Klasifikasi Penilaian Pada Virtual Patient Case*. Jurnal CESS Vol.2 No.2.
- Fauset, L., 1994. *Fundamentals of Neural Network*.
- Gillepsie, CJS., 2009. *Manson's Tropical Disease*. China: Saunders Elsevier.
- Hadisaputro, S., 1991. *Tropical Disease Update*. Laboratorium Ilmu Penyakit Dalam FK Undip. Semarang.
- Hariman, I & Noviar, A., 2014. *Sistem Pakar untuk Mendiagnosa Penyakit DBD (Demam Berdarah Dengue) dengan Metode Forward Chaining Berbasis Web*. Sekolah Tinggi Manajemen Informatika dan Ilmu Komputer LPKIA.
- Islamiah, N., 2016. *Klasifikasi penyakit gigi dan mulut menggunakan metode Learning Vector Quantization (LVQ)*. S1. Universitas Brawijaya.
- Shofia, E.N., Putri, R.R & Arwan, A., 2017. *Sistem Pakar Diagnosis Penyakit Demam: DBD, Malaria dan Tifoid Menggunakan Metode K-Nearest Neighbor – Certainty Factor*. S1. Universitas Brawijaya.
- Wafiyah, F., 2017. *Implementasi Algoritma Modified K-Nearest Neighbor (MKNN) untuk klasifikasi penyakit demam*. S1. Universitas Brawijaya.

