



# **IMPLEMENTASI *LEARNING VECTOR QUANTIZATION* (LVQ) UNTUK KLASIFIKASI KUALITAS AIR SUNGAI**

**SKRIPSI**

Untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Komputer

Disusun oleh:

Rifwan Hamidi

NIM: 135150200111149



**PROGRAM STUDI TEKNIK INFORMATIKA  
JURUSAN TEKNIK INFORMATIKA  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS BRAWIJAYA**

**MALANG**

**2017**

## PENGESAHAN

IMPLEMENTASI LEARNING VECTOR QUANTIZATION (LVQ) UNTUK KLASIFIKASI  
KUALITAS AIR SUNGAI

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Komputer

Disusun Oleh :  
Rifwan Hamidi  
NIM: 135150200111149

Skripsi ini telah diuji dan dinyatakan lulus pada  
25 Juli 2017

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

M. Tanzil Furdan, S.Kom, M.CompSc  
NIP: 19820930 200801 1 004

Dosen Pembimbing II

Bayu Rahayudi, S.T, M.T  
NIP: 19740712 200604 1 001

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 plagiaris, 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, 31 Juli 2017



Rifwan Hamidi

NIM: 135150200111149



## KATA PENGANTAR

Puji syukur penulis ucapkan kehadirat Allah SWT , karena hanya dengan berkat dan rahmat-Nya, penulis telah menyelesaikan skripsi dengan baik yang berjudul "**Implementasi *Learning Vector Quantization (LVQ)* untuk Klasifikasi Kualitas Air Sungai**".

Skripsi ini disusun untuk memenuhi sebagian persyaratan memperoleh gelar Sarjana Komputer di Program Studi Teknik Informatika, Jurusan Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya. Pada kesempatan ini, penulis ingin menyampaikan rasa hormat dan terima kasih yang sebesar-besarnya kepada semua pihak yang telah memberikan bantuan, pengarahan dan dukungan dalam pengerjaan skripsi ini, diantaranya:

1. Kedua Orang Tua penulis Ayah Budi Setiawan, S.Pd., M.Pd. dan Ibu Rita Lismarni serta keluarga besar, Paman Hendri Ekhman, adik Hakim Mustaqim dan adik Rendi Ramadhan yang selalu memberikan doa dan dukungan baik moril maupun materiil yang begitu besar terhadap kelancaran dalam menyelesaikan skripsi ini.
2. Bapak M. Tanzil Furqon, S.Kom, M.CompSc selaku dosen pembimbing I yang telah memberikan pengarahan dan saran kepada penulis sehingga dapat menyelesaikan skripsi ini.
3. Bapak Bayu Rahayudi , S.T, M.T selaku dosen pembimbing II yang telah memberikan bimbingan dan pengarahan dalam menyelesaikan skripsi ini.
4. Bapak Tri Santoso, Ssi., M.AP., M.IDS, selaku dosen pembimbing dan pakar yang telah memberikan pengarahan dan ilmunya kepada penulis mulai dari pelaksanaan Kuliah Kerja Nyata Praktik sampai menyelesaikan skripsi ini.
5. Seluruh Dosen dan Civitas Akademik Fakultas Ilmu Komputer, Universitas Brawijaya yang telah membagikan ilmu dan memberikan bantuan selama menempuh studi di Informatika dan selama penyelesaian skripsi ini.
6. Teman-teman kontrakan di Jalan Gajayana No.613 A, teman nongkrong AGP dan Pantai, kerajaan akamsi, foto studio padang, rantau pasaman, kelas TIF N, EMIF dan BPMIF informatika 2016, asisten praktikum RPL dan KB, teman-teman minang asal daerah Sumatera Barat serta penyemangat pribadi Sherly Bella yang telah memberikan semangat dan bantuan selama menempuh studi di Universitas Brawijaya.
7. Pihak-pihak lain yang tidak bisa penulis sebutkan satu persatu yang telah membantu kelancaran penulis dalam menyelesaikan skripsi ini.

Dengan segala kerendahan hati, penulis menyadari bahwa skripsi ini masih memiliki banyak kekurangan. Oleh Karena itu penulis mengharapkan kritik dan saran yang bersifat membangun untuk menyempurnakan skripsi ini agar lebih



baik. Penulis berharap semoga skripsi ini dapat bermanfaat bagi semua pihak dan khususnya bagi penulis sendiri. Terima kasih.

Malang, 31 Juli 2017

Penulis

hamidirifwan@gmail.com

## ABSTRAK

Air merupakan sumber daya alam yang sangat penting dan menjadi kebutuhan bagi aktivitas dan kelangsungan makhluk hidup, baik manusia, hewan maupun tumbuh-tumbuhan. Air sungai merupakan salah satu sumber air baku dari berbagai alternatif sumber air yang ada untuk dilakukan proses pengolahan. Namun seiring pertambahan penduduk, pertumbuhan industri, perkembangan ekonomi dan peningkatan standar hidup menyebabkan penurunan mutu atau kualitas air sungai itu sendiri. Pencemaran air sungai terjadi apabila di dalam air sungai terdapat berbagai macam zat atau kondisi yang dapat menurunkan standar kualitas air yang telah ditentukan, sehingga tidak dapat digunakan untuk kebutuhan tertentu. Oleh karena itu perlu adanya upaya untuk menjaga kualitas, kuantitas dan kontinuitas air sungai dengan melakukan pemantauan dan pengukuran kualitas air sungai. Sebelumnya telah dilakukan pengukuran dan penentuan kualitas air sungai menggunakan metode manual seperti Indeks Pencemaran (IP), *Water Quality Index* (WQI) dan *STORET* dengan kendala waktu dan biaya yang cukup tinggi. Sehingga diperlukan metode lain yang untuk mempercepat proses perhitungan secara efektif dan efisien yaitu menggunakan metode *Learning Vector Quantization* (LVQ) yang dapat mengklasifikasikan data menjadi 4 kelas kualitas air sungai berdasarkan 7 parameter masukan. Proses implementasi LVQ untuk klasifikasi air sungai diawali dengan tahapan pembagian dataset, pelatihan data dan pengujian serta klasifikasi data yang akan menghasilkan kelas berupa kelas memenuhi baku mutu, tercemar ringan, sedang dan berat. Hasil akurasi rata-rata terbaik didapatkan sebesar 81.13% dengan menggunakan alfa 0.1, *decrement* alfa 0.4, perbandingan data latih dan data uji 100:35 dari total 135 dataset, maksimal *epoch* 10 dan minimal *epoch* 0.001.

Kata kunci: klasifikasi, air sungai, kualitas air, *Learning Vector Quantization*, LVQ



## ABSTRACT

Water is a one of the natural resources which is very important and necessary for the activity and survival of living things, as humans, animals and plants. River is one of the source from various alternative sources of water available for processing. But nowadays as growth of population grows, industrial growth, economic development and rising standard of living cause degradation of quality of water itself. Pollution of river occurs when in the water there are various substances or conditions that can reduce water quality standards that have been determined, so it can't be used for certain needs. Therefore, there is an effort to maintain the quality, quantity and continuity of river water by monitoring and measuring the quality of water. Previously, river water quality and measurement were measured using manual methods such as Water Pollution Index (IP), *Water Quality Index* (WQI) and *STORET* with high time and cost constraints. So that another method is needed to speed up the calculation process effectively and efficiently using Learning Vector Quantization (LVQ) method which can classify data into 4 water quality class of river based on 7 input parameters. The LVQ implementation process for river water classification begins with the dataset division, data training, data testing and classification that will result in a class of normal, mild, moderate and heavy contaminated classes. The best average accuracy result is 81.13% using alpha 0.1, *decrement* alpha 0.4, comparison of training data and testing data 100: 35 from 135 total dataset, maximum *epoch* 10 and minimum *epoch* 0.001.

Keywords: Classification, river, water quality, Learning Vector Quantization, LVQ

## DAFTAR ISI

HALAMAN JUDUL .....	i
PENGESAHAN .....	ii
PERNYATAAN ORISINALITAS .....	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	vi
ABSTRACT .....	vii
DAFTAR ISI.....	viii
DAFTAR TABEL.....	xii
DAFTAR GAMBAR.....	xiii
DAFTAR PERSAMAAN.....	xv
DAFTAR LAMPIRAN .....	xvi
<b>BAB 1 PENDAHULUAN.....</b>	<b>1</b>
1.1 Latar belakang.....	1
1.2 Rumusan masalah.....	3
1.3 Tujuan .....	3
1.4 Manfaat.....	3
1.5 Batasan masalah.....	4
1.6 Sistematika pembahasan.....	4
<b>BAB 2 LANDASAN KEPUSTAKAAN .....</b>	<b>6</b>
2.1 Penelitian terdahulu .....	6
2.2 Air sungai .....	8
2.2.1 Kualitas air .....	9
2.2.2 Pencemaran air sungai.....	9
2.2.3 Parameter pencemaran air .....	10
2.3 Metode <i>STORET</i> .....	12
2.4 Jaringan Syaraf Tiruan (JST) .....	15
2.4.1 Arsitektur Jaringan Syaraf Tiruan (JST) .....	16
2.4.2 Learning Vector Quantization (LVQ) .....	18
<b>BAB 3 METODOLOGI .....</b>	<b>22</b>





3.1 Tahapan penelitian .....	22
3.2 Penentuan objek penelitian dan algoritma yang akan digunakan .....	23
3.3 Studi literatur .....	23
3.4 Pengumpulan dan analisis data air sungai .....	24
3.5 Analisis dan perancangan sistem klasifikasi kualitas air sungai .....	24
3.6 Implementasi sistem klasifikasi kualitas air sungai .....	25
3.7 Pengujian algoritma klasifikasi kualitas air sungai.....	25
3.8 Evaluasi dan hasil analisis .....	25
<b>BAB 4 PERANCANGAN.....</b>	<b>26</b>
4.1 Proses alur kerja sistem secara umum .....	26
4.2 Proses pelatihan dengan LVQ.....	27
4.3 Proses pengujian dengan LVQ.....	29
4.4 Proses perhitungan jarak .....	30
4.5 Perhitungan manual .....	31
4.5.1 Inisialisasi bobot awal, data latih dan parameter awal .....	32
4.5.2 Pelatihan dengan metode LVQ .....	33
4.5.3 Pengujian dengan metode LVQ .....	41
4.5.4 Perhitungan akurasi .....	43
4.6 Perancangan antarmuka.....	43
4.6.1 Antarmuka halaman beranda .....	43
4.6.2 Antarmuka halaman beranda admin .....	44
4.6.3 Antarmuka Halaman CRUD data admin.....	45
4.6.4 Antarmuka halaman beranda user (pengguna).....	46
4.6.5 Antarmuka Halaman generate dan analisis data.....	47
4.6.6 Antarmuka halaman pelatihan .....	48
4.6.7 Antarmuka halaman pengujian.....	48
4.6.8 Antarmuka halaman manualisasi.....	50
4.7 Skenario pengujian .....	51
4.7.1 Pengujian learning rate (alfa) awal .....	51
4.7.2 Pengujian pengurangan learning rate (decAlfa).....	52
4.7.3 Perbandingan jumlah data Latih dan data Uji .....	52
4.7.4 Pengujian maksimal perulangan ( <i>epoch</i> ).....	53



4.7.5 Pengujian minimal alfa.....	54
4.7.6 Pengujian akurasi dan validasi parameter optimal.....	55
<b>BAB 5 IMPLEMENTASI .....</b>	<b>56</b>
5.1 Lingkup perangkat keras .....	56
5.2 Lingkup perangkat lunak.....	56
5.3 Batasan-batasan implementasi .....	56
5.4 Implementasi kode program .....	57
5.4.1 Implementasi proses input data .....	57
5.4.2 Implementasi proses pelatihan Learning Vector Quantization (LVQ).....	58
5.4.3 Implementasi proses pengujian Learning Vector Quantization (LVQ).....	63
5.5 Implementasi antarmuka.....	66
5.5.1 Antarmuka halaman beranda .....	66
5.5.2 Antarmuka halaman beranda admin .....	66
5.5.3 Antarmuka halaman CRUD data admin .....	67
5.5.4 Antarmuka halaman beranda user (pengguna).....	67
5.5.5 Antarmuka halaman generate dan analisis data .....	67
5.5.6 Antarmuka halaman pelatihan .....	67
5.5.7 Antarmuka halaman pengujian.....	68
5.5.8 Antarmuka halaman manualisasi.....	68
<b>BAB 6 PENGUJIAN .....</b>	<b>72</b>
6.1 Hasil analisis pengujian parameter learning rate (alfa) awal .....	72
6.2 Hasil analisis pengujian parameter pengurang learning rate (decAlfa) .....	74
6.3 Hasil analisis pengujian parameter jumlah data latih dan data uji .....	75
6.4 Hasil analisis pengujian parameter maksimal perulangan ( <i>epoch</i> ) .....	77
6.5 Hasil analisis pengujian parameter minimal alfa .....	79
6.6 Hasil analisis pengujian akurasi dan validasi parameter optimal.....	80
<b>BAB 7 PENUTUP .....</b>	<b>82</b>
7.1 Kesimpulan.....	82
7.2 Saran .....	82
<b>DAFTAR PUSTAKA.....</b>	<b>84</b>





## DAFTAR TABEL

Tabel 2.1 Kriteria Kualitas Air berdasarkan Parameter dan Kelasnya .....	12
Tabel 2.2 Sistem Nilai untuk Menentukan Kualitas Air.....	13
Tabel 2.3 Sistem Nilai untuk Menentukan Kualitas Air.....	13
Tabel 4.1 Inisialisasi Bobot Awal (w).....	32
Tabel 4.2 Data Latih (x) .....	32
Tabel 4.3 Tabel Bobot Akhir (W) .....	41
Tabel 4.4 Data Uji .....	41
Tabel 4.5 Data Hasil klasifikasi perhitungan manual .....	43
Tabel 4.6 Pengujian <i>Learning Rate</i> Awal .....	51
Tabel 4.7 Pengujian Pengurangan <i>Learning Rate</i> .....	52
Tabel 4.8 Pengujian Iterasi Maksimum .....	53
Tabel 4.9 Pengujian Pengaruh Laju Pembelajaran Minimum .....	54
Tabel 4.10 Pengujian Perbandingan Data Latih dan Data Uji .....	53
Source Code 5.1 Proses Input data .....	58
Source Code 5.2 Proses Pelatihan Learning Vector Quantization (LVQ) .....	63
Source Code 5.3 Implementasi Proses Pengujian Learning Vector Quantization (LVQ).....	65
Tabel 6.1 Hasil Pengujian Pengaruh Learning Rate terhadap Hasil Akurasi.....	73
Tabel 6.2 Hasil Pengujian Pengaruh Pengurang Learning Rate terhadap Akurasi...74	
Tabel 6.3 Hasil Pengujian Pengaruh Jumlah Data Latih dan Data Uji terhadap Akurasi.....	76
Tabel 6. 4 Hasil Pengujian Pengaruh Maksimum Epoch terhadap Akurasi .....	77
Tabel 6. 5 Hasil Pengujian Pengaruh Minimum Alfa terhadap Akurasi .....	79
Tabel 6. 6 Hasil Pengujian Akurasi dan Validasi Parameter Optimal.....	81



## DAFTAR GAMBAR

Gambar 2.1 Model Struktur JST .....	16
Gambar 2.2 Struktur JST.....	16
Gambar 2.3 Jaringan Syaraf Tiruan Lapisan Tunggal .....	17
Gambar 2.4 Jaringan Syaraf Tiruan dengan Banyak Lapisan .....	18
Gambar 2.5 aringan Kompetitif .....	18
Gambar 2.6 Struktur Jaringan Learning Vector Quantization.....	19
Gambar 3.1 Diagram Blok Metodologi Penelitian .....	22
Gambar 4.1 Alur Perancangan .....	26
Gambar 4.2 Diagram Proses Alur Kerja Sistem secara Umum.....	27
Gambar 4.3 Diagram Alir Proses Pelatihan LVQ .....	28
Gambar 4.4 Diagram Alir Proses Pengujian LVQ.....	30
Gambar 4.5 Diagram Alir Proses Perhitungan Jarak .....	31
Gambar 4.6 Antarmuka Halaman Beranda .....	44
Gambar 4.7 Antarmuka Halaman <i>Input</i> Data .....	48
Gambar 4.8 Antarmuka halaman pelatihan.....	49
Gambar 4.9 Antarmuka Halaman Pengujian .....	49
Gambar 4.10 Antarmuka Halaman Manualisasi .....	50
Gambar 5.1 Antarmuka Halaman Beranda (Home).....	66
Gambar 5.2 Antarmuka Halaman Beranda Admin .....	67
Gambar 5.3 Antarmuka Halaman CRUD Dataset Kualitas Air Sungai.....	68
Gambar 5.4 Antarmuka Halaman Beranda Pengguna.....	69
Gambar 5.5 Antarmuka Halaman Generate Data bobot awal, Data Pelatihan dan Data Pengujian .....	69
Gambar 5.6 Antarmuka Halaman Pelatihan .....	70
Gambar 5.7 Antarmuka Halaman Pengujian .....	70
Gambar 5.8 Antarmuka Halaman Manualisasi .....	71
Gambar 6.1 Pengujian Pengaruh Learning Rate terhadap Hasil Akurasi.....	73
Gambar 6.2 Pengujian Pengaruh Pengurang Learning Rate terhadap Hasil Akurasi .....	75
Gambar 6.3 Pengujian Pengaruh Jumlah Data Latih dan Data Uji terhadap Hasil Akurasi.....	77



Gambar 6.4 Pengujian Pengaruh Maksimum *Epoch* terhadap Hasil Akurasi ..... 78

Gambar 6.5 Pengujian Pengaruh Minimum Alpha terhadap Hasil Akurasi ..... 80



## DAFTAR PERSAMAAN

Persamaan (2.1) Menghitung Jarak Euclidian.....	20
Persamaan (2.2) Menghitung Perbaikan Bobot $T = C_j$ .....	20
Persamaan (2.3) Menghitung Perbaikan Bobot $T \neq C_j$ .....	20
Persamaan (2.4) Menghitung Pengurangan Nilai <i>Learning Rate</i> .....	20



## DAFTAR LAMPIRAN

Lampiran 1 Form Validasi Dataset .....	86
--	----





## BAB 1 PENDAHULUAN

### 1.1 Latar belakang

Air merupakan sumber daya alam yang sangat penting dan menjadi kebutuhan bagi aktivitas dan kelangsungan makhluk hidup, baik manusia, hewan maupun tumbuh-tumbuhan. Air sungai merupakan salah satu sumber air baku dari berbagai alternatif sumber air yang ada untuk dilakukan proses pengolahan. Sungai menjadi tempat untuk menampung dan mengalirkan sumber daya air melewati daerah-daerah tertentu menuju daerah terendah di permukaan bumi. Sungai berperan sebagai perairan yang menjadi sumber air terdekat bagi beberapa penduduk pedesaan dan perkotaan serta tempat tinggal beberapa ekosistem air. Pemanfaatan terbesar sungai adalah untuk irigasi pertanian, bahan baku air minum, sebagai saluran pembuangan air hujan dan air limbah, bahkan sebenarnya berpotensi sebagai objek wisata. Namun seiring pertumbuhan penduduk, pertumbuhan industri, perkembangan ekonomi dan peningkatan standar hidup menyebabkan penurunan kualitas air sungai itu sendiri. Penurunan kualitas air sungai ditandai dengan kualitas air yang mengalir pada aliran sungai tersebut menjadi tercemar (Hartono, et al., 2009).

Pencemaran air sungai terjadi apabila di dalam air sungai terdapat berbagai macam zat atau kondisi yang dapat menurunkan standar kualitas air yang telah ditentukan, sehingga tidak dapat digunakan untuk kebutuhan tertentu. Berdasarkan laporan yang dikeluarkan oleh Direktorat Jenderal Pengendalian Pencemaran dan Kerusakan Lingkungan Kementerian Lingkungan Hidup dan Kehutanan (KLHK), di tahun 2015 hampir 68 persen atau mayoritas mutu air sungai di 33 provinsi di Indonesia dalam status tercemar berat (Wendyartaka, 2016). Oleh karena itu perlu adanya upaya untuk menjaga kualitas, kuantitas dan kontinuitas air sungai dengan melakukan pemantauan dan pengendalian pencemaran air sungai secara berkala, sebagaimana yang diatur dalam Peraturan Pemerintah Nomor 82 Tahun 2001. Dalam keputusan menteri Negara Lingkungan Hidup Nomor 115 Tahun 2003, terdapat dua metode untuk menentukan status mutu/kualitas air. Kedua metode itu adalah metode *STORET* dan metode Indeks Pencemaran (IP).

Penelitian Ramadhani, et al. (2016) adalah mengenai penerapan metode *Water Quality Index (WQI)* dan perhitungan *STORET* untuk menentukan status mutu air pada ruas Sungai Brantas Hilir. Pengujian kualitas air sungai umumnya menggunakan metode *STORET*, karena memiliki tingkat efektifitas yang tinggi dan mudah dipahami oleh masyarakat awam dibandingkan metode lainnya. Pada penelitian tersebut, parameter masukan dalam penentuan kualitas air berdasarkan metode *STORET* terdiri dari 7 parameter, yaitu TSS, BOD, COD, DO, minyak dan lemak, pH dan Fenol. Hasil dari perhitungan kualitas air dibagi menjadi 4 (empat) kriteria yaitu memenuhi baku mutu (kondisi baik), tercemar ringan, tercemar sedang dan tercemar berat. Namun, dalam penentuan kualitas air sungai dengan metode manual seperti metode *STORET* ini masih dilakukan perhitungan



secara manual dengan menghitung satu-persatu data parameter pengujian. Pada kenyataannya, saat ini proses pengukuran parameter-parameter kualitas air yang dilakukan beberapa instansi seperti PDAM (Perusahaan Daerah Air Minum) dan BLH (Badan Lingkungan Hidup) masih dilakukan secara manual dan penentuan klasifikasi kualitas air sungai menggunakan metode *STORET*. Proses ini membutuhkan waktu sekitar 1 sampai 30 hari tergantung pada parameter apa saja yang ingin diukur dan diteliti. Hasil yang diperoleh dari prosedur manual sangat bergantung pada seberapa terampil dan ahli dalam proses pengukuran air tersebut. Kelemahan metode *STORET* ini adalah memerlukan beberapa seri data yang cukup dalam penentuan kualitas air sungai (Yusrizal, 2015). Jika data yang ada hanya satu maka tidak bisa menentukan maksimum dan minimum baku mutu airnya dan menghasilkan kesimpulan yang berbeda-beda atau tidak pasti pada setiap data yang diuji. Sehingga dengan kelemahan metode tersebut, dalam menentukan dan mendapatkan hasil klasifikasi kualitas air sungai yang sangat akurat, pihak terkait membutuhkan bantuan dari laboratorium. Namun klasifikasi kualitas air sungai melalui laboratorium membutuhkan biaya yang cukup mahal dan waktu yang lama. Penggunaan teknologi informasi berupa penerapan kecerdasan buatan dalam melakukan klasifikasi data, dapat memberikan solusi dalam membantu proses penentuan terhadap klasifikasi kualitas air sungai yang lebih efektif dan efisien.

Penelitian yang memiliki tujuan dan objek yang sama mengenai klasifikasi kualitas air di Bangkok, Thailand yang dilakukan oleh Areerachakul dan Sanguansintukul (2009). Pada penelitian ini, digunakan *neural network* dengan algoritma Levenberg-Marquardt untuk mendapatkan keluaran berupa nilai indeks kualitas air. Terdapat 3 (tiga) kriteria masukan yang digunakan yaitu pH, DO dan BOD serta 5 (lima) kelas hasil yaitu class I *extra clean*, class II *very clean fresh*, class III *medium clean fresh*, class IV *fairly clean fresh* dan class V *the source which are not within class I to class IV*. Hasil dari penelitian tersebut menyimpulkan penerapan *neural network* dengan hanya menggunakan 3 parameter input menghasilkan akurasi yang tinggi yaitu 99,34%, namun memiliki error yang tinggi saat direkomendasikan menggunakan parameter input yang lebih banyak.

Penelitian selanjutnya yaitu Analisis dan klasifikasi tumor otak menggunakan metode *Support Vector Machine (SVM)* dan *Learning Vector Quantization (LVQ)* yang dilakukan oleh Jamadar dan Kakade (2015). Penelitian tersebut membandingkan tingkat akurasi dan tingkat performa antara kedua algoritma tersebut dalam menentukan klasifikasi, diagnosis dan deteksi penyakit tumor otak. Sehingga dengan perbandingan tersebut akan didapatkan metode yang lebih efisien dan lebih akurat secara kuantitatif dalam hal klasifikasi. Kesimpulan yang di dapatkan dari penelitian tersebut adalah algoritma LVQ menghasilkan akurasi dan performa yang lebih baik dengan tingkat akurasi 90% dibanding SVM sebesar 86,67% dalam klasifikasi penyakit tumor otak.

Penelitian perbandingan selanjutnya adalah membandingkan metode pada *neural network*, yaitu membandingkan metode Bp-LVQ *neural network* dalam mengklasifikasikan penyakit hati pada data analog dan digital oleh Rao dan



Murthy (2016). Dalam penelitian ini, dua metode klasifikasi yaitu metode *backpropagation* (Bp) dan *Learning Vector Quantization* (LVQ) digunakan untuk menyelesaikan permasalahan klasifikasi dan diagnosis pada penyakit hati. Dari hasil ujicoba didapatkan bahwa dalam klasifikasi menggunakan *Artificial Neural Network* (ANN), dimana metode LVQ memberikan kinerja lebih baik dibandingkan *backpropagation* dengan tingkat efisiensi *processing time* yang maksimum.

Penelitian selanjutnya memiliki objek dan metode yang sama, yaitu klasifikasi kualitas air sungai menggunakan *Learning Vector Quantization* (LVQ) yang dilakukan oleh Ying dan Mei (2016) dengan menentukan *output* berupa 3 kelas kualitas air berdasarkan 4 parameter *input* berupa pH, DO,  $\text{NH}_3$  dan COD di Sungai Huaihe, China. Pada penelitian ini, Algoritma LVQ terbukti memiliki akurasi yang tinggi dalam melakukan klasifikasi kualitas air sungai dengan tingkat akurasi sebesar 92%.

Berdasarkan pembahasan dari penelitian sebelumnya, metode *Learning Vector Quantization* (LVQ) memiliki peluang yang baik untuk diterapkan dalam klasifikasi status mutu atau kualitas air sungai kedalam 4 (empat) kriteria yaitu memenuhi baku mutu (kondisi baik), tercemar ringan, tercemar sedang dan tercemar berat berdasarkan 7 (tujuh) parameter *input* yang sesuai dan dikembangkan berdasarkan daerah air sungai di Indonesia. Sehingga diperoleh hasil dengan tingkat akurasi yang baik agar dapat membantu pemerintah dan pihak terkait dalam menanggulangi tingkat pencemaran air sungai. Dengan latar belakang demikian, sehingga penulis mengajukan proposal penelitian **Penerapan *Learning Vector Quantization* (LVQ) untuk Klasifikasi Kualitas Air Sungai.**

## 1.2 Rumusan masalah

Dengan latar belakang di atas, diperoleh rumusan masalah sebagai pedoman dalam penelitian ini adalah sebagai berikut:

1. Bagaimana implementasi metode *Learning Vector Quantization* (LVQ) untuk klasifikasi kualitas air sungai ?
2. Bagaimana nilai akurasi dari hasil klasifikasi kualitas air sungai menggunakan metode *Learning Vector Quantization* (LVQ) ?

## 1.3 Tujuan

Penelitian ini memiliki tujuan pencapaian diantaranya:

1. Menerapkan *Learning Vector Quantization* (LVQ) untuk penentuan kualitas air sungai.
2. Mengetahui hasil evaluasi berupa tingkat akurasi ketepatan perhitungan sistem dengan menerapkan *Learning Vector Quantization* (LVQ) untuk klasifikasi kualitas air sungai.

## 1.4 Manfaat

Manfaat dari penelitian ini diharapkan agar dapat melakukan klasifikasi kualitas air sungai dengan hasil yang baik dan dapat mempermudah perhitungan



pada tingkat pencemaran. Sehingga dapat digunakan serta menghasilkan manfaat positif kepada pemerintah dan masyarakat umum untuk menjadikan penelitian ini sebagai acuan untuk melakukan klasifikasi terhadap kualitas air sungai dan mengurangi aktifitas rumah tangga maupun industri yang menyebabkan pencemaran air sungai yang tinggi dan dapat melakukan tindakan penanganan yang tepat terhadap tingkat pencemaran air sungai.

### 1.5 Batasan masalah

Pada bagian ini merupakan batasan-batasan masalah yang diterapkan pada penelitian adalah sebagai berikut:

1. Fitur data yang digunakan pada penelitian ini adalah TSS, BOD, COD, DO, minyak dan lemak, pH dan Fenol.
2. Keluaran yang dihasilkan yaitu klasifikasi kategori kualitas air sungai dengan tingkat memenuhi baku mutu (kondisi baik), tercemar ringan, tercemar sedang dan tercemar berat.
3. Penelitian ini dibatasi pada proses perhitungan kualitas air sungai dengan metode *Learning Vector Quantization* (LVQ).

### 1.6 Sistematika pembahasan

Sistematika penyusunan laporan ditujukan untuk memberikan gambaran dan uraian dari penyusunan skripsi ini secara garis besar yang meliputi beberapa bab, sebagai berikut :

#### BAB 1 PENDAHULUAN

Menguraikan masalah umum terkait dengan penelitian yang bersistematis dan berkesinambungan. Bab ini berisi latar belakang, rumusan masalah, tujuan, manfaat dan batasan masalah yang terkait dalam Penerapan *Learning Vector Quantization* (LVQ) untuk Klasifikasi Kualitas Air Sungai.

#### BAB 2 LANDASAN KEPUSTAKAAN

Mendiskripsikan dasar teori dan penelitian sebelumnya yang digunakan dalam mendukung Penerapan *Learning Vector Quantization* (LVQ) untuk Klasifikasi Kualitas Air Sungai. Dasar teori terdiri dari Air Sungai, Metode *STORET*, Artificial Neural Network (ANN) atau Jaringan Syaraf Tiruan (JST) dan *Learning Vector Quantization* (LVQ).

#### BAB 3 METODOLOGI

Menguraikan langkah-langkah secara umum dan sistematis dalam menerapkan *Learning Vector Quantization* (LVQ) untuk mengklasifikasikan kualitas air sungai. BAB ini akan menjelaskan secara singkat mengenai metode yang menjadi dasar pengerjaan



yang perlu dilakukan sehingga menghasilkan *output* yang diinginkan.

#### **BAB 4 PERANCANGAN**

Membahas tentang perancangan sistem yang akan dibangun dalam implementasi *Learning Vector Quantization* (LVQ) untuk klasifikasi kualitas air sungai, seperti perancangan algoritma, perhitungan manualisasi, tampilan, pengujian dan alur program sistem yang akan diimplementasikan.

#### **BAB 5 IMPLEMENTASI**

Membahas tentang implementasi sistem dari hasil perancangan sistem yang telah dilakukan, seperti mengimplementasikan desain algoritma menggunakan bahasa pemrograman berdasarkan perancangan sistem sebelumnya.

#### **BAB 6 PENGUJIAN**

Menguraikan tentang pengujian fungsionalitas akurasi sistem dari implementasi yang telah diterapkan yaitu pengujian tingkat akurasi klasifikasi kualitas air sungai menggunakan *Learning Vector Quantization* (LVQ).

#### **BAB 7 PENUTUP**

Pada bab terakhir ini menguraikan kesimpulan yang diperoleh terhadap penelitian dan menyertakan saran yang dapat digunakan untuk pengembangan selanjutnya.



## BAB 2 LANDASAN KEPUSTAKAAN

Landasan kepastakaan ini akan membahas teori-teori dasar mengenai objek penelitian yang ada dan objek metode yang akan digunakan menunjang topik-topik pembahasan. Beberapa teori yang berkaitan diantaranya sebagai berikut:

### 2.1 Penelitian terdahulu

Dalam menyusun kajian pustaka ini digunakan penelitian yang memiliki objek, metode dan tujuan yang mirip dengan penelitian ini. Terdapat 6 penelitian sebelumnya yang menjadi referensi dan acuan, karena 6 penelitian tersebut dinilai paling sesuai dengan tujuan penelitian implementasi *Learning Vector Quantization* (LVQ) untuk klasifikasi kualitas air sungai. Penelitian sebelumnya yang menjadi referensi adalah penelitian yang dilakukan menggunakan objek yang sama yaitu klasifikasi terhadap status mutu atau kualitas air sungai tetapi masih menggunakan metode manual yaitu *STORET* maupun metode *artificial intelligence* lainnya dan penelitian yang menggunakan metode *Learning Vector Quantization* (LVQ) dengan objek yang sama dan objek yang berbeda. Beberapa penelitian sebelumnya secara umum telah dibahas pada bab pendahuluan latar belakang. Perbandingan penelitian dilakukan dengan membandingkan objek dan kriteria input yang digunakan, metode yang digunakan serta keluaran dan evaluasi akurasi yang dihasilkan.

Penelitian pertama menggunakan objek yang sama tetapi masih menggunakan metode manual, yaitu penerapan metode *Water Quality Index* (WQI) dan perhitungan *STORET* untuk menentukan status mutu air pada ruas sungai brantas hilir oleh Ramadhani, et al., (2016). Pada penelitian tersebut, peneliti meneliti kualitas air Sungai Brantas Hilir terutama daerah Jombang sampai Mojokerto dalam rentang tahun 2010-2014. Pendekatan yang dilakukan adalah perbandingan terhadap metode *Water Quality Index* (WQI) dan perhitungan *STORET*. Parameter input menggunakan 7 variabel input yang terdiri dari 1 variabel fisika yaitu residu tersuspensi (TSS), 4 variabel kimia organik yaitu Biological Oxygen Demand (BOD), Chemical Oxygen Demand (COD), oksigen terlarut (DO), dan derajat keasaman (pH), 2 variabel kimia organik atau biologi yaitu asam karbolat (Fenol) serta minyak dan Lemak. Adapun objek penelitian meliputi daerah Jembatan Ploso Jombang, Tambangan Cheil Jedang, Tambangan Betro Mojokerto, Jembatan Les Padangan, Jembatan Pulorejo dan Jembatan Tol Mojokerto. Hasil dari penelitian tersebut menjelaskan metode *STORET* memiliki akurasi dan performa yang lebih baik dalam mengklasifikasikan kualitas atau status mutu air sungai dalam 4 kategori kelas yaitu memenuhi baku mutu (kondisi baik), tercemar ringan, tercemar sedang dan tercemar berat serta menjelaskan kelebihan dan kekurangan dari kedua metode tersebut.

Penelitian kedua mengenai klasifikasi yaitu klasifikasi kualitas air sungai menggunakan *neural network* pada studi kasus kanal di Bangkok, Thailand oleh Areerachakul dan Sanguansintukul (2009). Pada penelitian ini, digunakan *artificial neural network* (ANN) atau jaringan syaraf tiruan menggunakan Levenberg-



Marquardt *algorithm* sebagai klasifikasi terhadap kualitas air sungai. Data yang diperoleh merupakan data kualitas air sungai pada 288 canal di Bangkok, Thailand yang didapatkan pada rentang tahun 2003-2007. Kriteria yang digunakan dalam menentukan nilai indeks kualitas air sungai yaitu terdiri dari pH, Dissolved Oxygen (DO), dan Biochemical Oxygen Demand (BOD). Klasifikasi Indeks kualitas berupa status mutu air dibagi menjadi 5 kriteria yaitu yaitu class I *extra clean*, class II *very clean fresh*, class III *medium clean fresh*, class IV *fairly clean fresh* dan class V *the source which are not within class I to class IV*. Tahapan dari penelitian ini dalam menerapkan *artificial neural network* menggunakan Levenberg-Marquardt *algorithm* adalah *preprocessing* data, *experimental data* dan *neural network simulation*. Pengujian sistem memiliki tingkat akurasi sebesar 99,34% sehingga peneliti menyarankan untuk menambah lebih banyak parameter input.

Jamadar dan Kakade (2015) melakukan perbandingan menggunakan metode LVQ dan SVM untuk menentukan klasifikasi, analisis dan deteksi terhadap penyakit tumor otak. Perbandingan metode dimaksudkan untuk mendapatkan metode yang lebih efisien dan lebih akurat secara kuantitatif dalam hal klasifikasi. Peneliti memodelkan metode menjadi dua bagian *training* dan *testing*. Pada bagian *training* dilakukan mulai *preprocessing*, *anisotropic diffusion* dan *feature extraction*, sedangkan pada bagian *testing* dilakukan klasifikasi menggunakan SVM dan LVQ. Hasil klasifikasi dari penelitian menunjukkan bahwa *Learning Vector Quantization* (LVQ) menghasilkan akurasi dan performa yang lebih baik dengan tingkat akurasi 90% dibanding *Support Vector Machine* (SVM) sebesar 86,67%, serta tingkat spesifikasi yang lebih spesifik dalam melakukan klasifikasi yaitu 1 berbanding 0,75. Sehingga berdasarkan penelitian tersebut, didapatkan bahwa metode LVQ memiliki akurasi yang lebih tinggi dibanding SVM untuk diterapkan pada objek klasifikasi.

Penelitian keempat adalah perbandingan metode pada *neural network*, yaitu membandingkan metode Bp-LVQ *neural network* dalam mengklasifikasikan penyakit hati pada data analog dan digital oleh Rao dan Murthy (2016). Dalam penelitian ini, dua metode klasifikasi yaitu metode *Backpropagation* (Bp) dan *Learning Vector Quantization* (LVQ) digunakan untuk menyelesaikan permasalahan klasifikasi dan diagnosis pada penyakit hati. Peneliti melakukan perbandingan performa dalam hal klasifikasi terhadap metode sesama metode yang ada pada *Neural network* tersebut. Peneliti menjelaskan penggunaan dari *artificial neural network* dalam hal klasifikasi memiliki bagian utama berupa arsitektur, inialisasi *weight* dan node serta fungsi aktifasi. Penerapan klasifikasi didasarkan pada dataset penyakit hati dari Cleveland dalam 100 data kasus dengan 14 atribut. Hasil dari penelitian dengan nilai alfa 0,1 di kedua metode, LVQ memiliki maksimum efisiensi yang lebih tinggi berupa nilai efisiensi tertinggi pada *backpropagation* hanya mencapai 79 % sedangkan LVQ mencapai 85%. Selain itu LVQ juga memiliki waktu proses yang lebih cepat secara keseluruhan pengujian. Sehingga, peneliti menyimpulkan bahwa LVQ memiliki performa yang lebih tinggi dibandingkan *Backpropagation*.



Selanjutnya penelitian menentukan klasifikasi penyakit gigi dan mulut (Islamiah, et al., 2015) menggunakan metode yang sama pada penelitian ini yaitu menggunakan metode *Learning Vector Quantization* (LVQ). Penelitian bekerja dengan tahapan pelatihan dan pengujian yang menghasilkan kelas *output* berupa klasifikasi penyakit *pulpitis*, *gingivitis*, *periodontis* dan *nekrosis pulpa*. Hasil pengujian sistem klasifikasi penyakit gigi dan mulut ini memiliki rata-rata akurasi terbaik yaitu 100% dengan menggunakan perbandingan data uji dan data latih sebesar 10:90, laju pembelajaran (alfa) 0,1, konstanta laju pembelajaran (*decAlfa*) 0,2, laju pembelajaran minimum (*minAlfa*) 10-5 dan jumlah iterasi maksimum (*maxEpoch*) adalah 10. Sehingga berdasarkan penelitian tersebut, LVQ memiliki peluang dengan hasil akurasi yang tinggi untuk diterapkan dalam objek klasifikasi.

Penelitian selanjutnya yaitu evaluasi terhadap model klasifikasi air sungai berdasarkan metode *Learning Vector Quantization* yang dilakukan oleh Ying dan Mei (2016). Objek, tujuan dan metode penelitian yang diteliti oleh Ying dan Mei memiliki kesamaan dengan penelitian yang dilakukan oleh penulis yaitu klasifikasi kualitas air sungai menggunakan metode LVQ. Penelitian tersebut bertujuan melakukan evaluasi, perlindungan dan pemeliharaan daerah aliran sungai dengan melakukan klasifikasi pada tiap daerah aliran sungai. Data penelitian didasarkan pada monitoring terhadap tiga daerah utama yaitu sungai Xufi Huaihe, Sungai Zhokou Shenqi dan jembatan Fuyuang Zhang. Data training terdiri dari 200 data (minggu) yang dihimpun dari tahun 2011 sampai 2014, sedangkan data testing terdiri dari 25 data (minggu) yang didapatkan pada tahun 2015. Klasifikasi terhadap objek air sungai didasarkan pada parameter 4 (empat) parameter *input* berupa pH, DO,  $Nh_3n$  dan COD dan parameter *output* berupa 3 (tiga) kelas hasil daerah pencemaran. Pada penelitian tersebut, Algoritma LVQ juga dibandingkan dengan metode *neural network* lainnya, yaitu *Back Propagation* (BP) dan *Radial Basis Function* (RBF). Hasil dari penelitian menjelaskan metode LVQ memiliki performa yang paling tinggi secara umum, tingkat akurasi yang tinggi sebesar 92%. Berdasarkan hasil pengujian tersebut, klasifikasi kualitas air sungai menggunakan metode LVQ dapat dikembangkan dan disesuaikan dengan kondisi di Indonesia berupa penambahan parameter input. Sehingga disimpulkan bahwa penggunaan metode LVQ dalam mengevaluasi klasifikasi kualitas air sungai terbukti sangat efektif dan beralasan.

## 2.2 Air sungai

Air merupakan sumber daya alam yang diperlukan untuk hajat hidup orang banyak, bahkan oleh semua makhluk hidup. Oleh karena itu sumberdaya air harus dilindungi agar tetap dapat dimanfaatkan dengan baik oleh manusia dan makhluk hidup lainnya. Pemanfaatan air untuk berbagai kepentingan harus dilakukan secara bijaksana dengan memperhitungkan kepentingan generasi sekarang dan generasi mendatang (Nugroho, 2008). Salah satu sumber air yang banyak dimanfaatkan untuk memenuhi kebutuhan hidup manusia dan makhluk hidup lainnya yaitu sungai. Sungai merupakan ekosistem yang sangat penting bagi manusia. Sungai juga menyediakan air bagi manusia baik untuk berbagai kegiatan seperti pertanian, industri maupun domestik. Di dalam suatu sistem Daerah Aliran





Sungai, sungai yang berfungsi sebagai wadah pengaliran air selalu berada di posisi paling rendah dalam *landscape* bumi, sehingga kondisi sungai tidak dapat dipisahkan dari kondisi Daerah Aliran Sungai (Peraturan, 2011).

### 2.2.1 Kualitas air

Berdasarkan Peraturan Pemerintah nomor 82 Tahun 2001 tentang Pengelolaan Kualitas Air dan Pengendalian Pencemaran Air, klasifikasi mutu atau kualitas air ditetapkan menjadi 4 (empat) kelas:

- a. Kelas satu, air yang peruntukannya dapat digunakan untuk air baku air minum, dan atau peruntukan lain yang mempersyaratkan mutu air yang sama dengan kegunaan tersebut.
- b. Kelas dua, air yang peruntukannya dapat digunakan untuk prasarana atau sarana rekreasi air, pembudidayaan ikan air tawar, peternakan, air untuk mengairi pertanaman, dan atau peruntukan lain yang mempersyaratkan mutu air yang sama dengan kegunaan tersebut.
- c. Kelas tiga, air yang peruntukannya dapat digunakan untuk budidaya ikan air tawar, peternakan, air untuk mengairi pertanaman, dan atau peruntukan lain yang mempersyaratkan mutu air yang sama dengan kegunaan tersebut.
- d. Kelas empat, air yang peruntukannya dapat digunakan untuk mengairi pertanaman dan atau peruntukan lain yang mempersyaratkan mutu air yang sama dengan kegunaan tersebut.

Kelas air tersebut berdasarkan perhitungan laboratorium dengan menggunakan sistem nilai dari US-EPA (*Unites States Environmental Protection Agency*) dapat diklasifikasikan lebih lanjut menjadi empat kelas, yaitu:

1. Kelas A (Baik Sekali) yang memenuhi baku mutu kelas satu.
2. Kelas B (Baik) dengan kondisi tercemar ringan.
3. Kelas C (Sedang) dengan kondisi tercemar sedang.
4. Kelas D (Buruk) dengan kondisi tercemar berat.

Kualitas air sungai sangat dipengaruhi oleh banyak atau sedikitnya tingkat pencemaran. Semakin tinggi tingkat pencemaran pada air, maka semakin rendah kualitas air tersebut dan tidak bisa dijadikan sumber air minum sehat. Menurut Undang Undang Republik Indonesia Nomor 32 Tahun 2009 tentang Perlindungan dan Pengelolaan Lingkungan Hidup, bahwa yang dimaksud pencemaran lingkungan hidup adalah masuk atau dimasukkannya makhluk hidup, zat, energi, dan/atau komponen lain ke dalam lingkungan hidup oleh kegiatan manusia sehingga melampaui baku mutu lingkungan hidup yang telah ditetapkan.

### 2.2.2 Pencemaran air sungai

Air sungai yang keluar dari mata air biasanya mempunyai kualitas yang sangat baik. Namun dalam proses pengalirannya air tersebut akan menerima berbagai



macam bahan pencemar. Perubahan kondisi kualitas air pada aliran sungai merupakan dampak dari buangan dari penggunaan lahan yang ada. Perubahan pola pemanfaatan lahan menjadi lahan pertanian, tegalan dan permukiman serta meningkatnya aktivitas industri akan memberikan dampak terhadap kondisi suatu Daerah Aliran Sungai (DAS). Selain itu, berbagai aktivitas manusia dalam memenuhi kebutuhan hidupnya yang berasal dari kegiatan industri, rumah tangga, dan pertanian akan menghasilkan limbah yang memberi sumbangan pada penurunan kualitas air sungai (Sofia, et al., 2010).

Peraturan Menteri Negara Lingkungan Hidup No. 01 tahun 2010 mendefinisikan bahwa pencemaran air adalah masuk atau dimasukkannya makhluk hidup, zat, energi, dan/atau komponen lain ke dalam air oleh kegiatan manusia, sehingga kualitas air turun sampai ke tingkat tertentu yang menyebabkan air tidak dapat berfungsi sesuai dengan peruntukannya. Sumber pencemar air berdasarkan karakteristik limbah yang dihasilkan dapat dibedakan menjadi sumber limbah domestik dan sumber limbah non-domestik. Sumber limbah domestik umumnya berasal dari daerah pemukiman penduduk dan sumber limbah non-domestik berasal dari kegiatan seperti industri, pertanian dan peternakan, perikanan, pertambangan, atau kegiatan yang bukan berasal dari wilayah pemukiman.

### 2.2.3 Parameter pencemaran air

Untuk mengetahui tingkat pencemaran suatu perairan perlu dilakukan pengujian parameter kualitas air. Dengan adanya indikator berupa parameter pencemaran air, pencemaran dapat diatasi sedini mungkin atau paling tidak sedikit mungkin. Pada penelitian ini digunakan acuan Keputusan Menteri Negara Lingkungan Hidup nomor : 115 tahun 2003 tentang Pedoman Penentuan Status Mutu air sebagai standar parameter mutu air. Sedangkan, baku mutu air menggunakan acuan Peraturan Pemerintah RI nomor 82 tahun 2001 tentang pengelolaan Kualitas Air dan Pengendalian Pencemaran. Parameter yang umum diuji untuk menentukan tingkat pencemaran air adalah parameter fisika, kimia organik dan kimia anorganik air. Parameter yang digunakan adalah sebagai berikut (Ramadhani, et al., 2016):

#### a. Total Suspended Solid (TSS)

TSS (*Total Suspended Solid*) atau total padatan tersuspensi adalah padatan yang tersuspensi di dalam air yang berupa bahan-bahan organik dan anorganik yang dapat disaring dengan kertas millipore berpori-pori 0,45  $\mu\text{m}$ . TSS terdiri dari lumpur dan pasir halus serta jasad-jasad renik, yang terutama disebabkan oleh kikisan tanah atau erosi tanah yang terbawa ke badan air (Agustira, 2013). Materi yang tersuspensi mempunyai dampak buruk terhadap kualitas air karena mengurangi penetrasi matahari ke dalam badan air, kekeruhan air meningkat yang menyebabkan gangguan pertumbuhan bagi organisme produser.



b. Biochemical Oxygen Demand (BOD)

BOD (*Biochemical Oxygen Demand*) atau kebutuhan oksigen biologi adalah suatu karakteristik yang menunjukkan jumlah oksigen terlarut dalam ppm atau mg/l yang diperlukan oleh organisme untuk mengurai atau mendekomposisi bahan organik dalam kondisi aerobik sehingga limbah tersebut jernih kembali. Semakin besar jumlah bahan-bahan organik yang terkandung didalam air, maka semakin banyak pula oksigen yang diperlukan bakteri untuk menguraikan zat organik, sehingga mengakibatkan menurunnya kadar oksigen pada air. Semakin besar kadar BOD menunjukkan rendahnya tingkat kualitas air. Pengukuran BOD penting untuk menentukan daya cemar air.

c. Chemical Oxygen Demand (COD)

COD (*Chemical Oxygen Demand*) atau kebutuhan oksigen kimiawi adalah jumlah oksigen dalam ppm atau mg/l yang dibutuhkan dalam kondisi tertentu untuk menguraikan benda organik secara kimiawi. Kadar COD pada pencemaran air menunjukkan ukuran zat-zat organik yang secara alamiah dapat dioksidasikan melalui proses mikrobiologis dan mengakibatkan berkurangnya oksigen terlarut pada air.

d. Dissolved Oxygen (DO)

DO (*Dissolved Oxygen*) atau Oksigen terlarut merupakan jumlah oksigen dalam ppm atau mg/l yang terkandung di dalam air. Oksigen yang terlarut dipergunakan sebagai ukuran derajat pengotoran limbah yang ada. Semakin besar nilai oksigen yang terlarut, maka menunjukkan derajat pencemaran air yang relatif kecil.

e. Potential of Hydrogen (pH)

Derajat keasaman pH (*potential of hydrogen*) merupakan ukuran konsentrasi hidrogen yang menyatakan tingkat keasaman atau kebasaan yang dimiliki oleh suatu larutan. Air yang memiliki kualitas yang normal memiliki ukuran pH – 7 yang berarti tidak bersifat basa maupun asam. Air yang memiliki pH kurang dari 6.5 maka air tersebut dapat merusak pipa distribusi.

f. Fenol

Fenol merupakan penyebab rasa yang ada di dalam air. Fenol biasanya dihasilkan dari limbah buangan industri. Fenol yang terkandung pada perairan memiliki dampak negatif berupa menyimpangnya reproduksi pada zooplankton dan invertebrata yang menjadi sumber makanan ikan yang dapat menyebabkan kerusakan genetik dan gangguan sistem kekebalan pada ikan. Selain itu, fenol juga membahayakan masyarakat sekitar sungai yang mengkonsumsi air tersebut. Senyawa fenol dan turunannya dapat menimbulkan gangguan kesehatan seperti gangguan perkembangan janin, cacat lahir dan kanker.

g. Minyak dan Lemak

Minyak dan lemak merupakan komponen utama bahan makanan yang juga banyak terkandung di dalam air limbah. Apabila lemak tidak dihilangkan

sebelum dibuang ke saluran air limbah, maka dapat mempengaruhi kehidupan yang ada di permukaan air dan menimbulkan lapisan tipis dipermukaan sehingga membentuk selaput.

Kriteria kualitas atau status mutu air berdasarkan parameter dan kelasnya ditunjukkan pada Tabel 2.1.

**Tabel 2.1 Kriteria Kualitas Air berdasarkan Parameter dan Kelasnya**

NO	Parameter	Satuan	Kelas Air			
			I	II	III	IV
	FISIKA					
1	TSS	Mg/L	50	50	400	400
	KIMIA					
	a. Anorganik					
2	BOD	Mg/L	2	3	6	12
3	COD	Mg/L	10	25	50	100
4	DO	Mg/L	6	4	3	0
5	Ph	Mg/L	6-9	6-9	6-9	5-9
	b. Organik/Biologi					
6	Fenol	Mg/L	0,001	0,001	0,001	-
7	Minyak dan Lemak	Mg/L	1	1	1	-

Sumber : (Peraturan, 2011)

### 2.3 Metode STORET

Metode *STORET* merupakan salah satu metode manual untuk menentukan status mutu air yang umum digunakan. Dengan metode *STORET* ini dapat diketahui parameter-parameter yang telah memenuhi atau melampaui baku mutu air. Secara prinsip metode *STORET* adalah membandingkan antara data kualitas air dengan baku mutu air yang disesuaikan dengan peruntukannya guna menentukan status mutu air dan “US-ESP (*Environmental Protection Agency*)” dengan mengklasifikasikan kualitas air dalam empat kelas (Keputusan, 2003). Berdasarkan Keputusan Menteri Lingkungan Hidup No. 115 tahun 2003, penentuan status mutu air dengan menggunakan metode *STORET* dilakukan dengan langkah-langkah sebagai berikut:

1. Mengumpulkan data kualitas air dan debit air secara periodik untuk mendapatkan data dari waktu ke waktu (*time series data*) minimal 2 seri data.
2. Mencari nilai maksimum, minimum dan rata-rata dari data mutu air tersebut.



3. Membandingkan data hasil pengukuran/pengujian dari masing-masing parameter air dengan nilai baku mutu sesuai dengan kelas air.
4. Jika hasil pengukuran/pengujian memenuhi nilai baku mutu air (hasil pengukuran/pengujian < baku mutu) maka diberi skor 0. Jika hasil pengukuran/pengujian tidak memenuhi nilai baku mutu air (hasil pengukuran /pengujian melebihi baku mutu), maka diberi skor sesuai dengan Tabel 2.2.

**Tabel 2.2 Sistem Nilai untuk Menentukan Kualitas Air**

Jumlah Parameter*	Nilai	Parameter		
		Fisika	Kimia Anorganik	Kimia Organik/Biologi
< 10	Maksimum	-1	-2	-3
	Minimum	-1	-2	-3
	Rata-rata	-2	-6	-9
≥ 10	Maksimum	-2	-4	-6
	Minimum	-2	-4	-6
	Rata-rata	-6	-12	-18

Sumber : (Keputusan, 2003)

Catatan\* : jumlah parameter yang digunakan untuk penentuan status mutu atau kualitas air

5. Menghitung jumlah negatif dari seluruh parameter dan menentukan status mutu airnya dari jumlah skor yang didapat dengan menggunakan sistem nilai dari US-EPA (*Unites States Environmental Protection Agency*) dengan mengklasifikasikan mutu air dalam empat kelas, yaitu berdasarkan Tabel 2.3.

**Tabel 2.3 Sistem Nilai untuk Menentukan Kualitas Air**

No	Kelas	Kategori	Skor	Keterangan
1	Kelas A	Baik Sekali	0	Memenuhi baku mutu
2	Kelas B	Baik	-1 s/d -10	Tercemar ringan
3	Kelas C	Sedang	-11 s/d -30	Tercemar sedang
4	Kelas D	Buruk	≥-30	Tercemar berat

Sumber : (Keputusan, 2003)

Contoh Perhitungan *STORET*:

Diberikan data parameter fisika, kimia dan biologi sebagai berikut.



No	Parameter	Satuan	Baku Mutu	Max	Min	Rata-rata	Skor
1	Fisika						
a.	TSS	mg/L	50	14	14	14	0
2.	Kimia						
a.	Anorganik						
3.	BOD	mg/L	3	12	4,1	8,05	-10
4.	COD	mg/L	25	20,2	10,3	15,25	0
5.	DO	mg/L	>4	15	7,4	11,2	0
6.	Sianida (CN)	mg/L	0,02	0,002	0,0055	0,00375	0
7.	pH	mg/L	6-9	7,3	6,8	7,05	0
b.	Organik						
8.	Fenol	mg/L	0,001	0,001	0,001	0,001	0
9.	Minyak dan Lemak	mg/L	1	1	0	0,5	0
total							-10
Hasil klasifikasi kualitas air sungai adalah kelas B (tercemar ringan)							

Cara pemberian skor tiap parameter adalah sebagai berikut (Contoh untuk BOD):

- a. Jumlah Parameter yang digunakan pada contoh adalah 9 parameter, berarti gunakan aturan <10 parameter pada Tabel 2.2.
- b. BOD merupakan parameter kimia anorganik, maka skor yang digunakan untuk nilai maksimal, minimum dan rata-rata adalah parameter kimia pada Tabel 2.2.
- c. Baku mutu air untuk BOD adalah 3 mg/L.
- d. Kadar BOD maksimum hasil pengukuran pada contoh adalah 12 mg/L, berarti melebihi baku mutunya. Maka skor untuk nilai maksimum BOD adalah -2.
- e. Kadar BOD minimum hasil pengukuran pada contoh adalah 4,1 mg/L, berarti melebihi baku mutunya. Maka skor untuk nilai minimum BOD adalah -2.
- f. Kadar BOD rata-rata hasil pengukuran pada contoh adalah 8,05 mg/L, berarti melebihi baku mutunya. Maka skor untuk nilai rata-rata BOD adalah -6.
- g. Jumlahkan seluruh skor untuk nilai maksimum, minimum dan rata-rata. Skor BOD pada contoh ini adalah  $(-2) + (-2) + (-6) = -10$



- h. Lakukan langkah yang sama pada tiap parameter, apabila ada baku mutu pada suatu parameter yang tidak didefinisikan, maka tidak perlu dilakukan perhitungan.
- i. Kemudian, jumlahkan seluruh skor parameter.
- j. Pada contoh ini, total skor keseluruhan adalah -10. Berdasarkan Tabel 2.3, hal ini berarti air sungai tersebut tergolong tercemar ringan (kelas B).

## 2.4 Jaringan Syaraf Tiruan (JST)

*Artificial Neural Network* (ANN) atau Jaringan syaraf tiruan (JST) adalah representasi buatan dari otak manusia yang mencoba untuk mensimulasikan proses pembelajaran pada otak manusia. Istilah buatan digunakan karena jaringan syaraf ini diimplementasikan menggunakan program komputer yang mampu menyelesaikan sejumlah proses perhitungan selama proses pembelajaran berjalan. JST menyerupai cara kerja otak manusia dalam dua hal, yaitu pengetahuan diperoleh dari proses belajar dan kekuatan hubungan antar sel syaraf (neuron) yang dikenal sebagai bobot-bobot yang digunakan untuk menyimpan pengetahuan. Fitur unggulan yang ada pada JST adalah kemampuannya melakukan proses generalisasi dari data latih ke dalam data yang baru (data lain). Kemampuan belajar tersebut dapat dianalogikan sama dengan proses manusia belajar mengenali sesuatu (Wuryandari, 2012).

Jaringan syaraf tiruan diciptakan sebagai suatu generalisasi model matematis dari pemahaman manusia (*human cognition*) yang didasarkan atas asumsi (Wuryandari, 2012) sebagai berikut:

1. Informasi diperoleh dari elemen yang sederhana melalui neuron.
2. Terdapat aliran sinyal antara neuron menggunakan penghubung.
3. Setiap sambungan penghubung memiliki nilai bobot yang sesuai. Dimana, bobot berfungsi untuk perkalian sinyal yang dikirim.
4. Keseluruhan sel syaraf akan diterapkan fungsi aktivasi sinyal hasil yang memproses bobot masuk pada setiap sel sehingga menentukan sinyal keluaran.

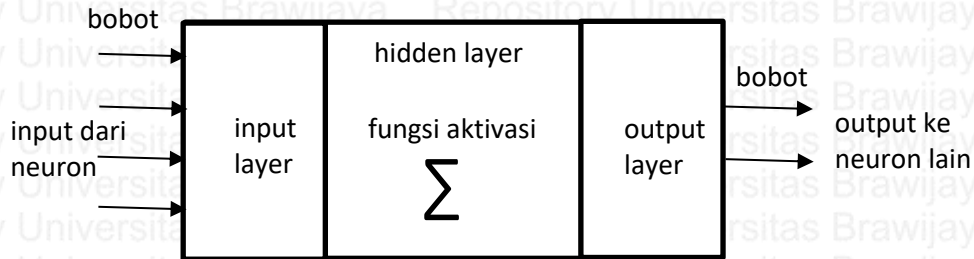
Secara umum, terdapat tiga bagian pada JST yaitu:

- Lapis masukan (*input layer*) merupakan bagian neuron yang menerima data masukan dari suatu variabel. Semua neuron pada lapis ini dapat terhubung ke neuron pada lapisan tersembunyi atau langsung ke lapisan luar jika jaringan tidak menggunakan lapisan tersembunyi.
- Lapisan tersembunyi (*hidden layer*) terdiri dari neuron yang menerima dan memproses data dari lapisan masukan melalui suatu fungsi perambatan.
- Lapisan keluaran (*output layer*) terdiri dari neuron yang menerima data dari lapisan tersembunyi atau langsung dari lapisan masukan yang nilai nilainya



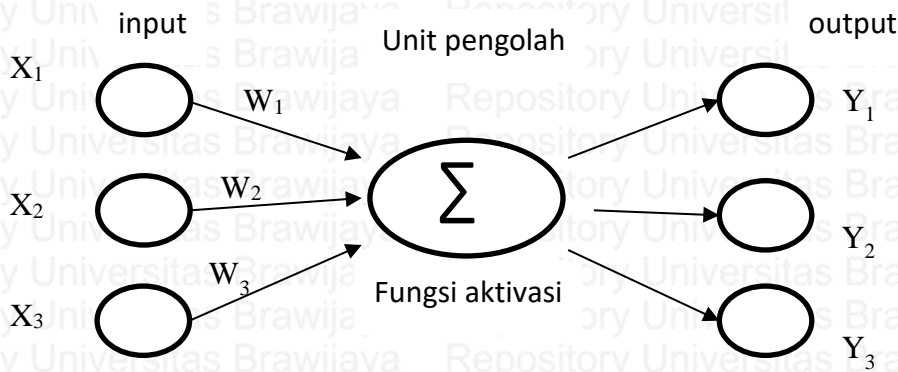
merupakan solusi hasil perhitungan dan *output* yang diharapkan dan dikirimkan ke neuron lainnya.

Pemodelan struktur dari neuron jaringan syaraf tiruan ditunjukkan pada Gambar 2.1 dan 2.2.



**Gambar 2.1 Model Struktur JST**

(Sumber: Wuryandari, 2012)



**Gambar 2.2 Struktur JST**

(Sumber: Wuryandari, 2012)

### 2.4.1 Arsitektur Jaringan Syaraf Tiruan (JST)

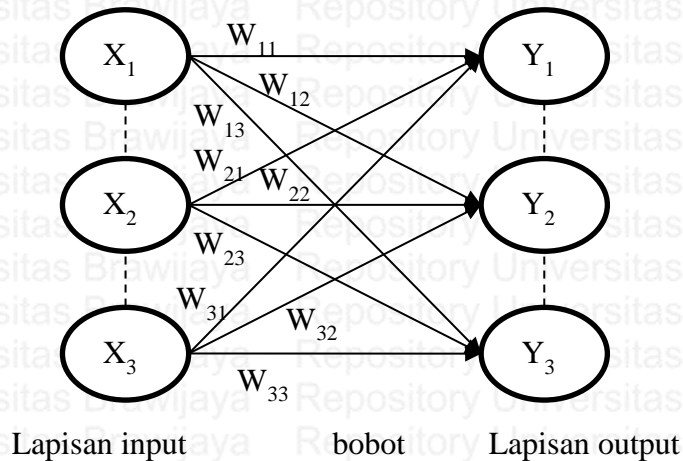
Jaringan Syaraf Tiruan (JST) dapat belajar dari pengalaman yang biasanya data *input* berhubungan dengan data angka (numerik), sehingga data yang tidak numerik harus dibuat numerik. JST disebut dengan metode yang free- estimator yang artinya tidak memiliki rumus yang tetap. JST disebut juga black box atau tidak transparan karena tidak mampu menjelaskan bagaimana suatu hasil bisa didapatkan. JST memiliki beberapa arsitektur, diantaranya sebagai berikut (Puspitaningrum, 2006):





### 1. Jaringan dengan lapisan tunggal (*Single layer net*)

Jaringan yang memiliki arsitektur ini hanya memiliki satu lapisan bobot koneksi. jaringan lapisan tunggal terdiri dari unit-unit input yang menerima sinyal dari luar dan unit output yang memberikan respon tanpa melalui lapisan tersembunyi. Pemodelan jaringan syaraf tiruan lapisan tunggal ditunjukkan pada Gambar 2.3.



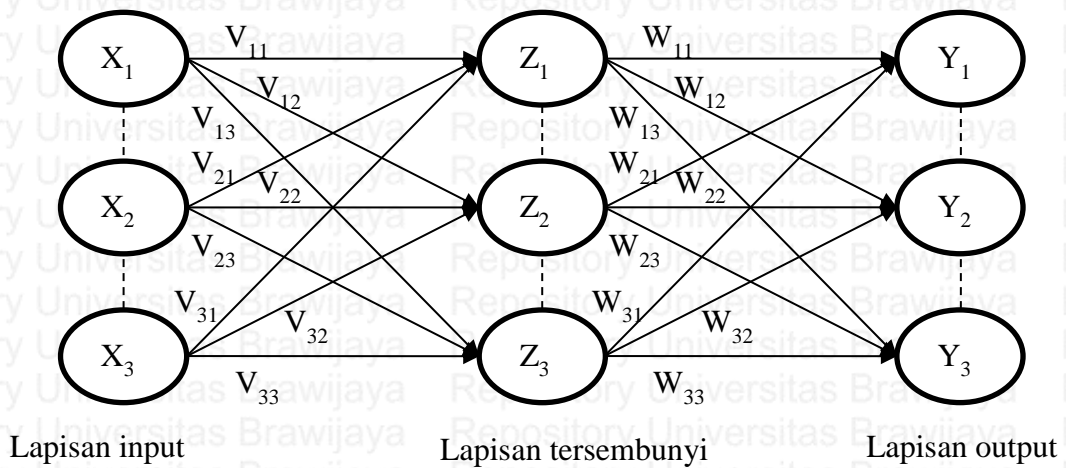
**Gambar 2.3 Jaringan Syaraf Tiruan Lapisan Tunggal**  
(Sumber: Puspitaningrum, 2006)

### 2. Jaringan dengan banyak lapisan (*Multi layer net*)

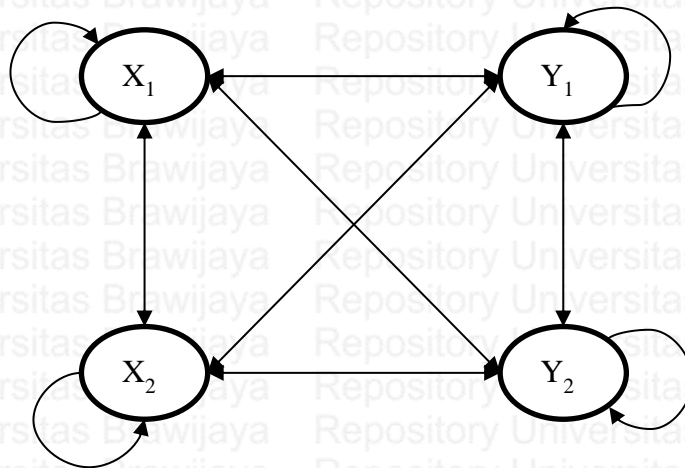
Jaringan multi layer memiliki satu atau lebih lapisan input, lapisan tersembunyi dan lapisan output. Sehingga memiliki kemampuan lebih dalam memecahkan masalah tetapi memiliki pelatihan yang juga lebih rumit dan fungsi pembelajaran yang lebih kompleks. Pemodelan jaringan syaraf tiruan lapisan banyak ditunjukkan pada Gambar 2.4.

### 3. Jaringan Kompetitif

Pada jaringan ini hubungan antar neuron tidak diperlihatkan secara langsung pada arsitektur. Hubungan antar neuron digambarkan sebagai jaringan yang rumit, dimana setiap neuron bersaing untuk menjadi aktif. Pemodelan jaringan syaraf tiruan lapisan kompetitif ditunjukkan pada Gambar 2.5.



**Gambar 2.4 Jaringan Syaraf Tiruan dengan Banyak Lapisan**  
 (Sumber: Puspitaningrum, 2006)



**Gambar 2.5 Jaringan Kompetitif**  
 (Sumber: Puspitaningrum, 2006)

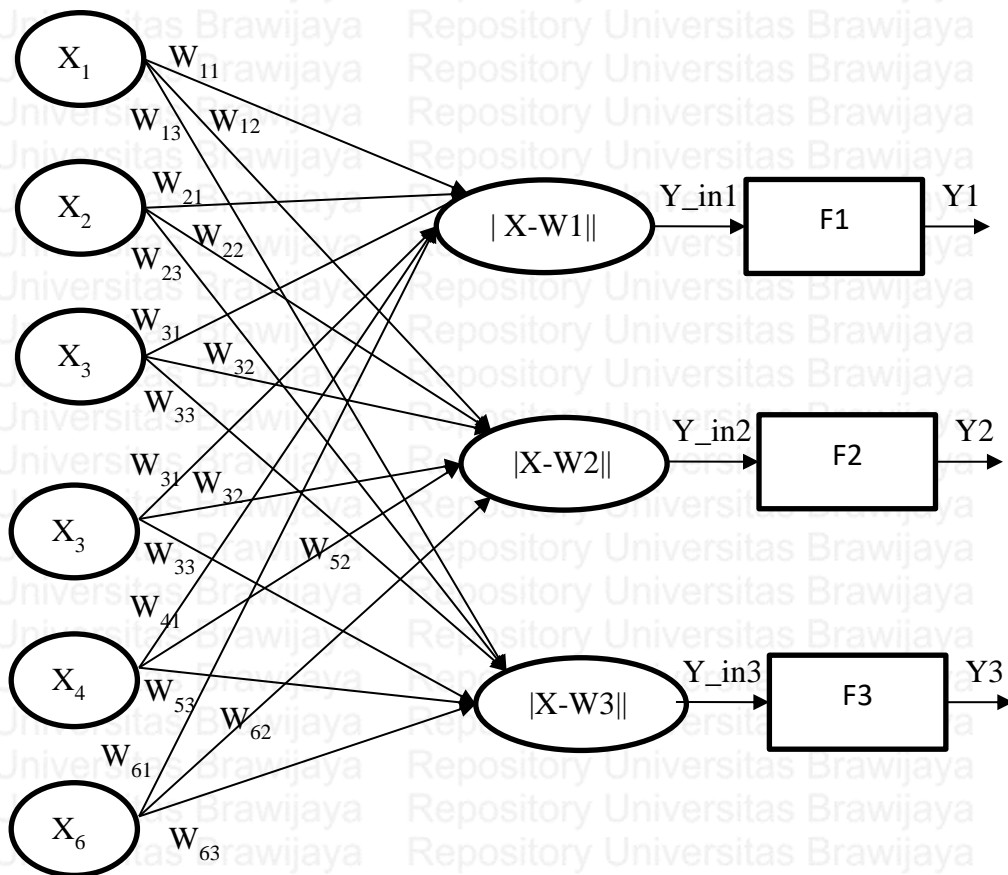
### 2.4.2 Learning Vector Quantization (LVQ)

*Learning Vector Quantization* (LVQ) adalah suatu metode pelatihan untuk melakukan pembelajaran pada lapisan kompetitif yang terawasi (*supervised learning*) yang arsitektur jaringannya berlapis tunggal (*single layer*). Suatu lapisan kompetitif akan secara otomatis belajar untuk mengklasifikasikan vektor-vektor input. Kelas-kelas yang didapatkan sebagai hasil dari lapisan kompetitif ini hanya tergantung pada jarak antara vektor-vektor input. Jika dua vektor input mendekati sama, maka lapisan kompetitif akan meletakkan kedua vektor input tersebut ke



dalam kelas yang sama. LVQ merupakan metode klasifikasi pola masing-masing unit keluaran mewakili kategori atau kelas tertentu (beberapa unit keluaran seharusnya digunakan untuk masing-masing kelas). Keunggulan dari metode LVQ adalah kemampuannya untuk memberikan pelatihan terhadap lapisan-lapisan kompetitif sehingga secara otomatis dapat mengklasifikasikan vektor input yang diberikan (Nugroho, 2011).

Selama training, bobot di-adjust secara supervised. Vektor bobot untuk suatu unit keluaran sering dinyatakan sebagai sebuah *vector referens*. Diasumsikan bahwa serangkaian pola pelatihan dengan klasifikasi yang tersedia bersama dengan distribusi awal *vector referens*. Setelah pelatihan, jaringan LVQ melakukan klasifikasi vektor masukan dengan menugaskan ke kelas yang sama sebagai unit keluaran, sedangkan yang mempunyai *vector referens* diklasifikasikan sebagai vektor masukan. Gambar 2.6 menjelaskan aritektur jaringan LVQ dengan  $X_1$ - $X_6$  bertindak sebagai dendrit atau data masukan,  $W$  sebagai sinapsi atau bobot, soma (badan sel) adalah perhitungan  $X-W$  dan  $Y$  sebagai data keluaran (Wuryandari, 2012).



**Gambar 2.6 Struktur Jaringan *Learning Vector Quantization***

(Sumber: Wuryandari, 2012)



Langkah-langkah algoritma pelatihan LVQ terdiri atas:

1. Inisialisasi:
  - a. Bobot awal variabel input ke- $j$  menuju ke kelas ke- $i$ :  $W_{ij}$ , dengan  $i=1,2,\dots,K$ ; dan  $j=1,2,\dots,m$ .
  - b. Maksimum *epoch*:  $MaxEpoch$ .
  - c. Parameter *learning rate* atau laju pembelajaran ( $\alpha$ ):  $\alpha$ .
  - d. Pengurangan *learning rate* ( $Deca$ ):  $Deca$ .
  - e. Minimal *learning rate* yang diperbolehkan ( $MinAlfa$ ):  $Min\alpha$ .
2. Masukkan:
  - a. Data input:  $X_{ij}$ ; dengan  $i=1,2,\dots,n$ ; dan  $j=1,2,\dots,m$ .
  - b. Target berupa kelas:  $T_k$  dengan  $k=1,2,\dots,n$ .
3. Tetapkan kondisi awal:  $epoch=0$ .
4. Kerjakan jika: ( $epoch < MaxEpoch$ ) dan ( $\alpha \geq Min\alpha$ ).
  - a.  $epoch = epoch+1$ .
  - b. Kerjakan untuk  $i=1$  sampai  $n$ .

i. Tentukan  $J$  sedemikian hingga  $\|X_i - W_j\|$  minimum dengan  $j=1,2,\dots,K$ . Perhitungan  $j$  menggunakan rumus jarak Euclidian.

$$D(j) = \sum (W_{ij} - x_i)^2 \quad (2.1)$$

ii. Perbaiki  $W_j$  dengan ketentuan:

Jika  $T = C_j$  maka

$$W_j(t+1) = w_j(t) + \alpha(t)[x(t) - w_j(t)] \quad (2.2)$$

Jika  $T \neq C_j$  maka

$$W_j(t+1) = w_j(t) - \alpha(t)[x(t) - w_j(t)] \quad (2.3)$$

c. Kurangi nilai  $\alpha$ , pengurangan  $\alpha$  bisa dilakukan dengan:

$$\alpha = \alpha - \alpha * Deca \quad (2.4)$$

5. Tes kondisi berhenti dengan,

$X$ , vektor-vektor pelatihan ( $X_1, \dots, X_i, \dots, X_n$ ).

$T$ , kategori atau kelas yg benar untuk vektor pelatihan  $W_j$ , vektor bobot pada unit keluaran ke- $j$  ( $W_{1j}, \dots, W_{ij}, \dots, W_{nj}$ ).

$C_j$ , kategori atau kelas yang merepresentasikan oleh unit keluaran ke- $j$ .

$\|x - w_j\|$ , jarak Euclidean antara vektor masukan dan vektor bobot untuk unit keluaran ke- $j$ .



Adapun Kelebihan dari LVQ adalah (Wuryandari, 2012):

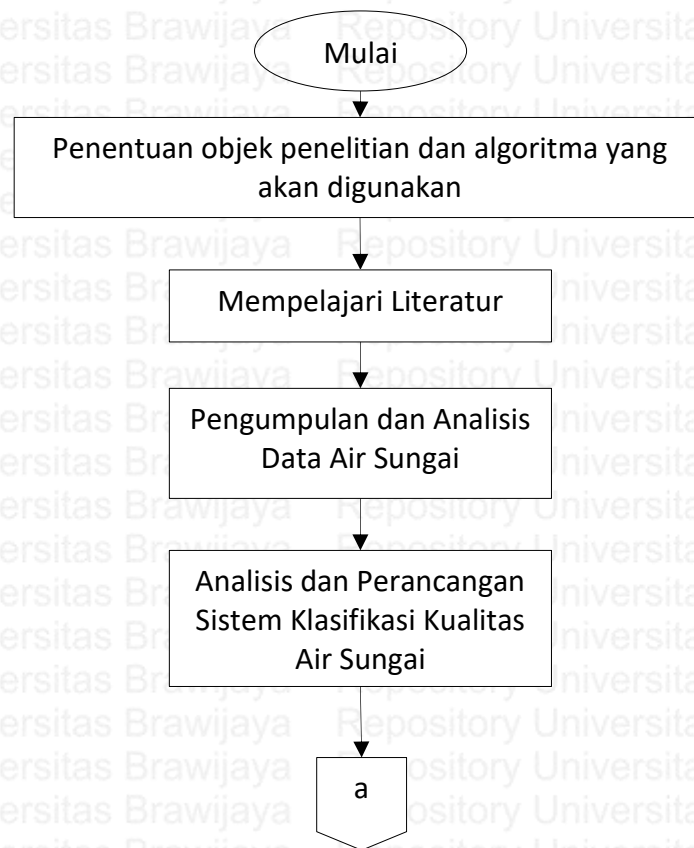
1. Nilai error yang lebih kecil dibandingkan dengan jaringan syaraf tiruan seperti *backpropagation*.
2. Dapat meringkas data set yang besar menjadi vektor *codebook* berukuran kecil untuk klasifikasi.
3. Dimensi dalam *codebook* tidak dibatasi seperti dalam teknik *nearest neighbour*.
4. Model yang dihasilkan dapat diperbaharui secara bertahap.

## BAB 3 METODOLOGI

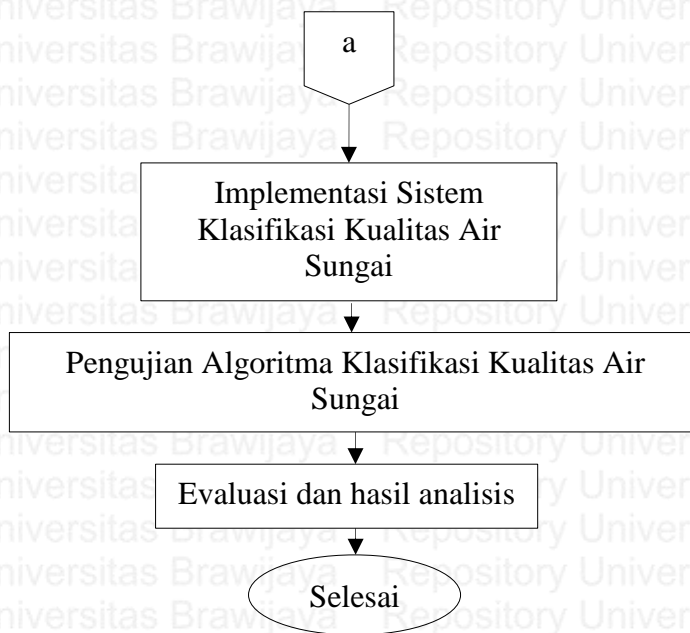
Pada bab ini dijelaskan mengenai tahapan dalam implementasi *Learning Vector Quantization* untuk klasifikasi kualitas air sungai. Penelitian ini termasuk dalam tipe penelitian perancangan (*design*) yang akan menghasilkan sebuah *prototype* atau *simulasi* berupa perangkat lunak untuk melakukan klasifikasi kualitas air sungai.

### 3.1 Tahapan penelitian

Dalam melaksanakan penelitian terdapat beberapa langkah-langkah yang harus dilakukan dan langkah-langkah tersebut digunakan sebagai panduan alur pengerjaan dalam penelitian yang ditujukan agar penelitian berjalan dengan sesuai. Pada Gambar 3.1 ditunjukkan diagram alir metodologi penelitian.



Gambar 3.1 Diagram Blok Metodologi Penelitian



**Gambar 3.1 Diagram Blok Metodologi Penelitian Lanjutan**

### 3.2 Penentuan objek penelitian dan algoritma yang akan digunakan

Penelitian ini menggunakan objek status mutu atau kualitas air sungai yang diklasifikasi menggunakan algoritma *Learning Vector Quantization (LVQ)* seperti yang dijelaskan pada Bab 2 tentang kajian pustaka penelitian terkait sebelumnya yang telah terbukti efektif digunakan pada masalah pengklasifikasian dan pengkategorian objek.

### 3.3 Studi literatur

Pada bagian ini akan difokuskan pada pencarian referensi yang relevan yang berkaitan dengan permasalahan yang sedang diteliti oleh penulis. Hal ini dilakukan agar peneliti mampu meningkatkan pemahaman dan pengetahuan teori tentang permasalahan yang sedang diteliti. Referensi berupa teori yang didapatkan berkaitan dengan implementasi *Learning Vector Quantization (LVQ)* untuk klasifikasi kualitas air sungai. Dasar teori disusun berdasarkan referensi yang diperoleh dari artikel, buku, jurnal, e-book, konferensi, peraturan perundang-undangan, website lembaga, media massa serta penelitian-penelitian terkait baik nasional maupun internasional dan bimbingan secara langsung dari pembimbing dalam proses penelitian. Adapun literatur yang dipelajari tentang :

- a. Penelitian Terdahulu.
- b. Objek Kualitas Air Sungai.
- c. Teori *Learning Vector Quantization (LVQ)*.



### 3.4 Pengumpulan dan analisis data air sungai

Kebutuhan data diperlukan untuk mendukung objek penelitian dalam klasifikasi kualitas air sungai. Data yang digunakan dalam penelitian ini merupakan data parameter pengukuran kualitas air sungai. Nilai dari parameter tersebut dihitung menggunakan hasil laboratorium sehingga diperoleh hasil diagnosis kualitas air sungai. Data kualitas air sungai merupakan dataset atau data sekunder berdasarkan penentuan kualitas air sungai yang dilakukan oleh pakar yaitu Kasubbid Pengendalian Lingkungan dan Pengelolaan Limbah Badan Lingkungan Hidup (BLH) Kota Malang.

### 3.5 Analisis dan perancangan sistem klasifikasi kualitas air sungai

Tahap analisis merupakan langkah awal dalam penelitian yang harus tercukupi. Tahap analisis bertujuan untuk mengetahui dan menganalisis permasalahan sehingga didapatkan sebuah pemecahan atau solusi yang dapat dimodelkan sedemikian rupa dalam klasifikasi kualitas air sungai. Karena penelitian ini bertipe perancangan (*design*), sehingga analisis diperlukan untuk dapat mendukung seluruh tahap-tahap penelitian dalam perancangan, implementasi dan pengujian objek dengan metode *Learning Vector Quantization* (LVQ).

Tahap perancangan bertujuan untuk mengetahui alur kerja dari proses penyelesaian permasalahan. Penggunaan variabel dan penggunaan data kualitas air sungai pada proses perhitungan dan penerapannya terhadap algoritma *Learning Vector Quantization* (LVQ) menjadi salah satu alur dan penyelesaian permasalahan. Desain aplikasi harus dapat menampilkan hasil dan solusi permasalahan. Tahap perancangan sistem bermaksud agar mempermudah dalam tahapan selanjutnya yaitu implementasi sistem. Perancangan yang dihasilkan dapat digunakan sebagai acuan dalam tahap implementasi dan pengujian. Berikut merupakan penjelasan tahap yang dilakukan dalam perancangan sistem dan akan diperjelas lebih rinci pada bab 4 perancangan.

1. Perhitungan *manualisasi* dalam penyelesaian permasalahan dilakukan dengan membuat studi kasus perhitungan sederhana menggunakan dataset kualitas air sungai dan metode *Learning Vector Quantization* (LVQ). Sehingga didapatkan solusi berupa langkah-langkah penyelesaian objek berupa algoritma dan diagram alir.
2. Perancangan antarmuka bertujuan untuk memberikan gambaran umum tentang tampilan sistem.
3. Perancangan pengujian terdiri dari pengujian metode dan data yang digunakan, dengan cara melakukan pengujian tingkat akurasi algoritma dan beberapa skenario pengujian. Secara umum pengujian metode berupa uji coba perulangan atau iterasi terhadap dataset hasil klasifikasi laboratorium dan pakar yang akurat dengan hasil klasifikasi dengan metode *Learning Vector Quantization* (LVQ).





### 3.6 Implementasi sistem klasifikasi kualitas air sungai

Pada tahap ini akan menjelaskan tentang implementasi berdasarkan perancangan *Learning Vector Quantization* (LVQ) untuk klasifikasi kualitas air sungai yang telah dibuat dan akan dilakukan dan diperjelas lebih rinci pada bab 4 implementasi. Implementasi sistem yaitu mengimplementasikan algoritma *Learning Vector Quantization* (LVQ) ke dalam bahasa pemrograman untuk melakukan perhitungan klasifikasi kualitas air sungai dimana bahasa pemrograman yang digunakan adalah bahasa pemrograman PHP.

### 3.7 Pengujian algoritma klasifikasi kualitas air sungai

Pada tahap ini akan dilakukan pengujian yang bertujuan untuk menunjukkan bahwa sistem dapat bekerja sesuai dengan yang diharapkan. Hal yang dilakukan pada tahapan pengujian adalah dengan cara melakukan pengujian tingkat akurasi klasifikasi kualitas air sungai menggunakan metode *Learning Vector Quantization* (LVQ). Parameter yang digunakan dapat berpengaruh terhadap nilai akurasi. Dalam penelitian ini pengujian menggunakan 135 data dari dataset BLH. Skenario pengujian yang akan dilakukan adalah sebagai berikut:

1. Pengujian parameter *Learning Rate* laju pembelajaran (alfa)
2. Pengujian parameter pengurangan *Learning Rate* atau *decAlfa*
3. Pengujian perbandingan jumlah data latih dan data uji
4. Pengujian iterasi maksimum
5. Pengujian pengaruh laju pembelajaran atau *minAlfa*.
6. Pengujian Akurasi dan Validasi Parameter Optimal

### 3.8 Evaluasi dan hasil analisis

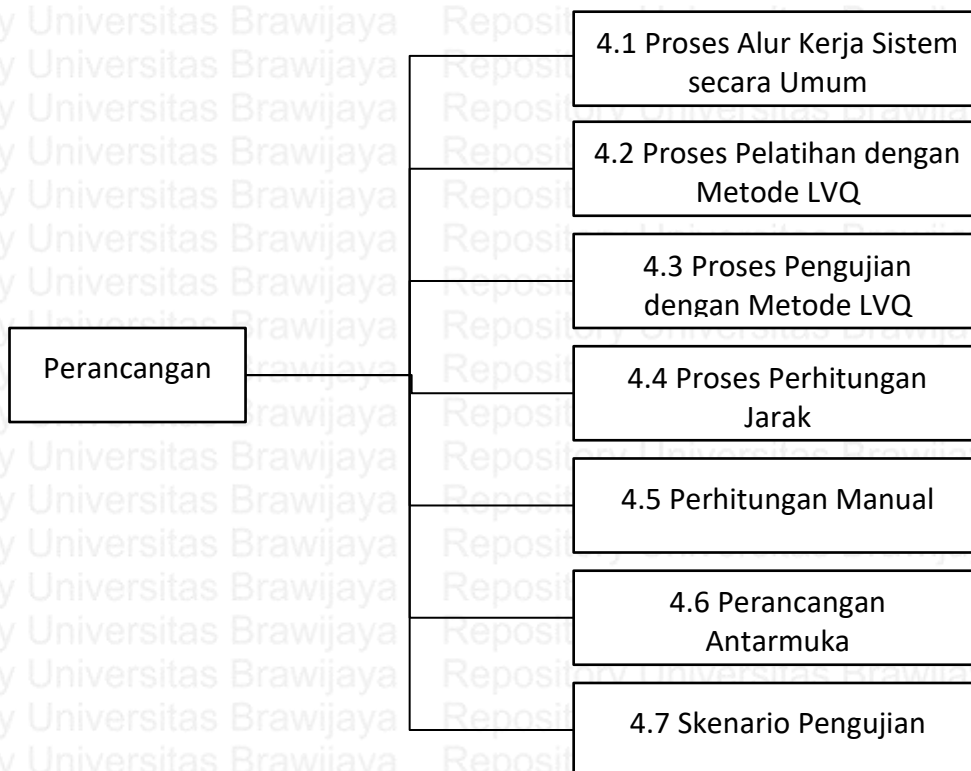
Setelah tahapan penelitian dimulai dari studi literatur sampai pengujian sistem telah dilakukan, selanjutnya dapat dilakukan evaluasi dan hasil analisis berupa kesimpulan berdasarkan hasil penelitian prototype atau simulasi sistem klasifikasi kualitas air sungai menggunakan *Learning Vector Quantization* (LVQ). Kesimpulan diambil dari hasil pengujian dan analisis metode. Sehingga dapat ditarik kesimpulan untuk menjawab perumusan permasalahan yang telah dirumuskan.

Selain penarikan kesimpulan, terdapat evaluasi dalam melakukan penelitian ini seperti kekurangan maupun terjadi kesalahan. Penulisan saran berfungsi untuk memberikan masukan dan pertimbangan atas hasil yang telah dilakukan. Sehingga dapat dilakukan pengembangan ataupun perbaikan dalam penelitian selanjutnya, baik dari segi objek kualitas air sungai maupun metode *Learning Vector Quantization* (LVQ). Dengan harapan kedepannya, penelitian selanjutnya dapat melakukan penelitian yang lebih baik dari penelitian sebelumnya.



## BAB 4 PERANCANGAN

Pada bab ini dibahas proses perancangan dari “Implementasi *Learning Vector Quantization* untuk Klasifikasi Kualitas Air Sungai”. Perancangan proses kerja dari LVQ dibagi menjadi 2 bagian utama, yaitu pelatihan data yang bertujuan untuk menghasilkan bobot akhir (optimal) dan pengujian data yang bertujuan untuk mengetahui hasil klasifikasi. Selanjutnya, Perancangan utama tersebut dapat dikembangkan menjadi 7 bagian yang lebih spesifik yaitu perancangan proses alur kerja sistem secara umum, proses pelatihan dengan metode LVQ, proses pengujian dengan metode LVQ, proses perhitungan jarak, perhitungan manual, perancangan antarmuka dan skenario pengujian untuk mengetahui akurasi dari sistem. Alur perancangan ditunjukkan pada Gambar 4.1.



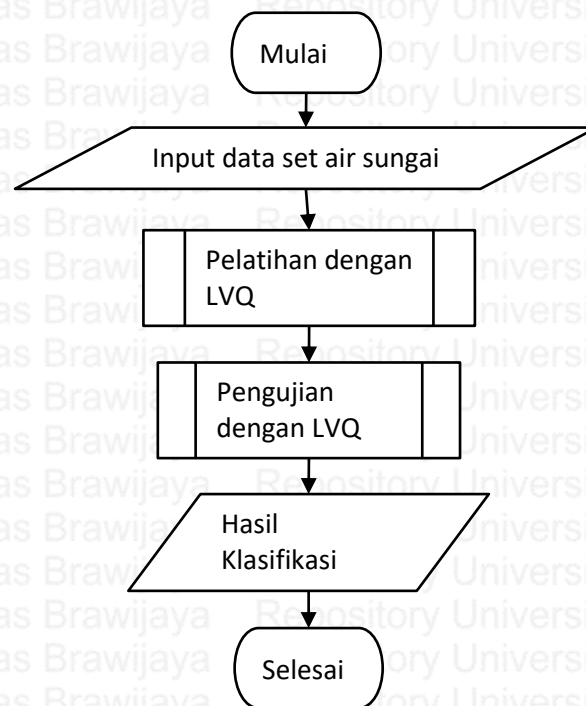
Gambar 4.1 Alur Perancangan

### 4.1 Proses alur kerja sistem secara umum

Proses dan alur kerja sistem secara umum terdiri dari beberapa tahapan mulai dari *input* data parameter kualitas air sungai hingga diketahui hasil klasifikasi air sungai tersebut. Beberapa tahapan tersebut meliputi *input* parameter kualitas air sungai, pelatihan dengan algoritma LVQ, pengujian dengan algoritma LVQ dan *output* hasil klasifikasi kualitas air sungai. Secara umum, dataset yang digunakan



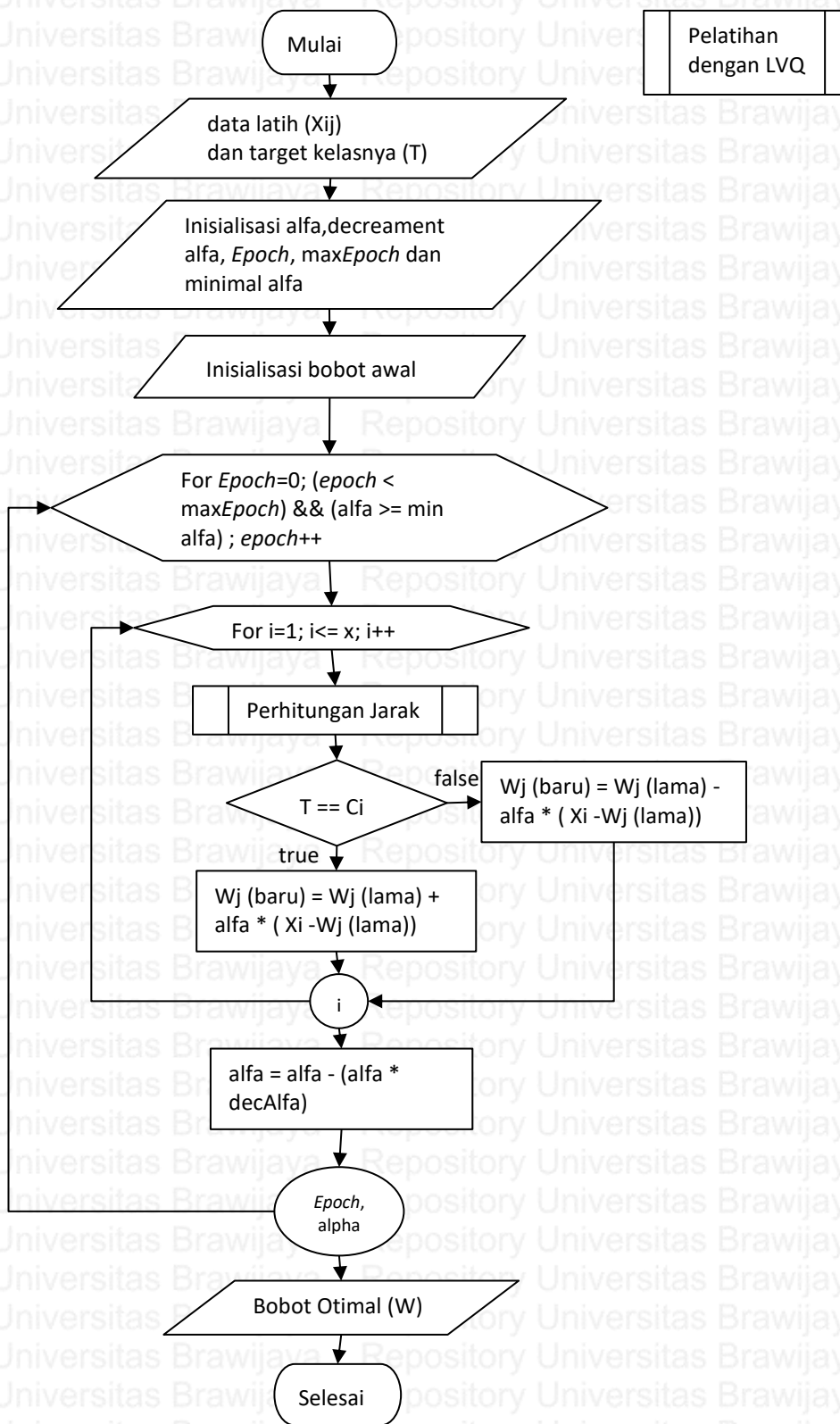
dibagi menjadi 3 bagian, yaitu data latih, data bobot awal (*centroid*) yang mewakili tiap kelas hasil klasifikasi dan data uji. Pada proses pelatihan data, pengguna memasukkan data latih yang akan digunakan sebagai bahan pembelajaran dan data parameter-parameter LVQ yang dibutuhkan. Hasil atau *output* dari pelatihan data yang didapatkan berupa data bobot akhir optimal yang mewakili jarak dari tiap kelas *output*. Selanjutnya pada proses pengujian, pengguna memilih data untuk dilakukan pengujian antara data uji tersebut terhadap data bobot optimal yang didapatkan dari proses pelatihan. Sehingga akan diproses untuk mendapatkan hasil klasifikasi dari kualitas air sungai yang dibagi menjadi ke dalam empat kelas (memenuhi baku mutu, tercemar ringan, sedang dan berat). Proses alur kerja sistem secara umum ditunjukkan pada Gambar 4.2.



**Gambar 4.2 Diagram Proses Alur Kerja Sistem secara Umum**

#### 4.2 Proses pelatihan dengan LVQ

Proses pelatihan merupakan tahapan awal dari algoritma *Learning Vector Quantization*. Proses pelatihan dengan LVQ dilakukan setelah pengguna memasukkan data latih yang akan digunakan untuk proses pelatihan. Pada proses pelatihan ini juga membutuhkan parameter masukan lainnya seperti alfa, *decrement* alfa (penurunan alfa), *minimum* alfa (nilai alfa terkecil) dan *MaxEpoch* (*epoch* atau iterasi maksimum). Proses pelatihan dimulai dari menginisialisasi vektor input pada data latih sampai proses perubahan bobot. Hasil atau output dari proses pelatihan adalah nilai bobot akhir optimal yang digunakan untuk dasar proses pengujian. Gambar 4.3 menunjukkan proses pelatihan dengan LVQ.



Gambar 4.3 Diagram Alir Proses Pelatihan LVQ



Tahapan proses pelatihan dengan LVQ yaitu sebagai berikut:

1. *Input* data latih, data latih yang digunakan berupa data kualitas air sungai beserta nilai variabel *input*, yaitu TSS, BOD, COD, DO, pH, fenol, minyak lemak dan skor serta kelas target (T) berupa kelas *output* klasifikasi, karena pada 135 dataset yang digunakan tidak terdapat kelas yang memenuhi baku mutu, maka kelas *output* yang digunakan adalah 1 (tercemar ringan), 2 (tercemar sedang) dan 3 (tercemar berat).
2. Inialisasi parameter awal yang diperlukan untuk proses pelatihan, yaitu nilai *alfa* (laju pembelajaran), *decrement alfa* (penurunan alfa), *minimum alfa* (nilai alfa terkecil) dan *MaxEpoch* (iterasi maksimum). Selanjutnya, inialisasi bobot awal (*W*) dengan nilai acak, dimana data yang dipilih adalah 3 data yang mewakili masing-masing kelas *output*. Lakukan langkah 3 sampai 6 dengan ketentuan iterasi atau perulangan belum mencapai *MaxEpoch* (iterasi maksimum) atau laju pembelajaran lebih besar dari *minimum alfa*.
3. Melakukan proses perulangan sebanyak jumlah data latih. Selanjutnya untuk setiap vektor pelatihan *x*, tentukan  $C_j$  (kelas *output* terdekat) dengan menghitung jarak *Euclidean Distance* dari  $||x - W_j||$ . Selanjutnya dicari nilai  $C_j$  yang paling kecil atau minimum diantara 3 kelas *output*.
4. Perbarui nilai  $W_j$  dengan ketentuan:  
Jika  $T = C_j$   
$$W_j(\text{baru}) = w_j(\text{lama}) + \alpha * [x_i - w_j(\text{lama})]$$
  
Jika  $T \neq C_j$  maka  
$$W_j(\text{baru}) = w_j(\text{lama}) - \alpha * [x_i - w_j(\text{lama})]$$
5. Kurangi laju pembelajaran dengan melakukan pengurangan laju pembelajaran terhadap konstanta *decrement alfa* dikali laju pembelajaran.
6. Mengembalikan bobot baru yang dihasilkan untuk dijadikan bobot optimal dan digunakan untuk proses selanjutnya.

### 4.3 Proses pengujian dengan LVQ

Setelah proses pelatihan dilakukan, tahapan selanjutnya adalah proses pengujian. Proses pengujian dilakukan mulai dengan menginisialisasi vektor input data uji sampai proses penentuan kelas hasil klasifikasi. Tahapan pengujian dilakukan untuk mengetahui hasil akurasi dari metode LVQ. Langkah-langkah pengujian ditunjukkan melalui Gambar 4.4.

Penjelasan tentang tahapan proses pengujian dengan LVQ meliputi:

1. *Input* data uji, data uji yang digunakan berupa data kualitas air sungai beserta nilai variabel *input*, yaitu TSS, BOD, COD, DO, pH, fenol, minyak lemak dan skor.
2. Inialisasi bobot optimal yang didapatkan dari proses pelatihan.



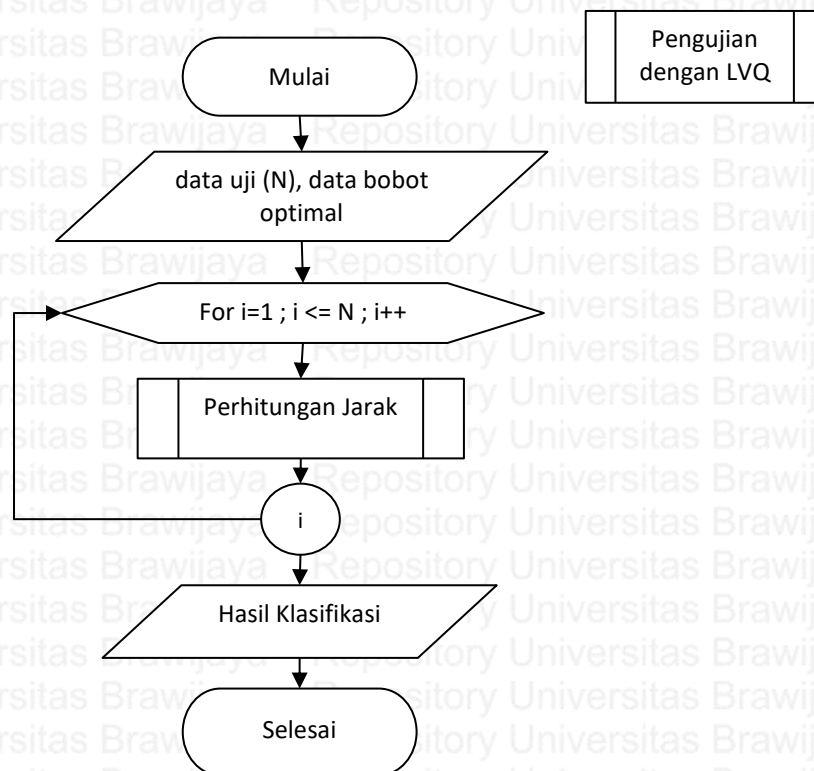
3. Melakukan proses perulangan sebanyak jumlah data uji. Selanjutnya untuk setiap vektor pelatihan  $x$ , tentukan  $C_j$  dengan menghitung jarak *Euclidean Distance* dari  $\|x - W_j\|$ . Selanjutnya dicari nilai  $C_j$  yang paling kecil atau minimum diantara 3 kelas *output*.
4. Mengembalikan data hasil klasifikasi berupa kelas dari hasil pengujian yaitu nilai  $C_j$  yang minimum diantara ketiga kelas *output*, yaitu kelas 1 (tercemar ringan), 2 (tercemar sedang) dan 3 (tercemar berat).

#### 4.4 Proses perhitungan jarak

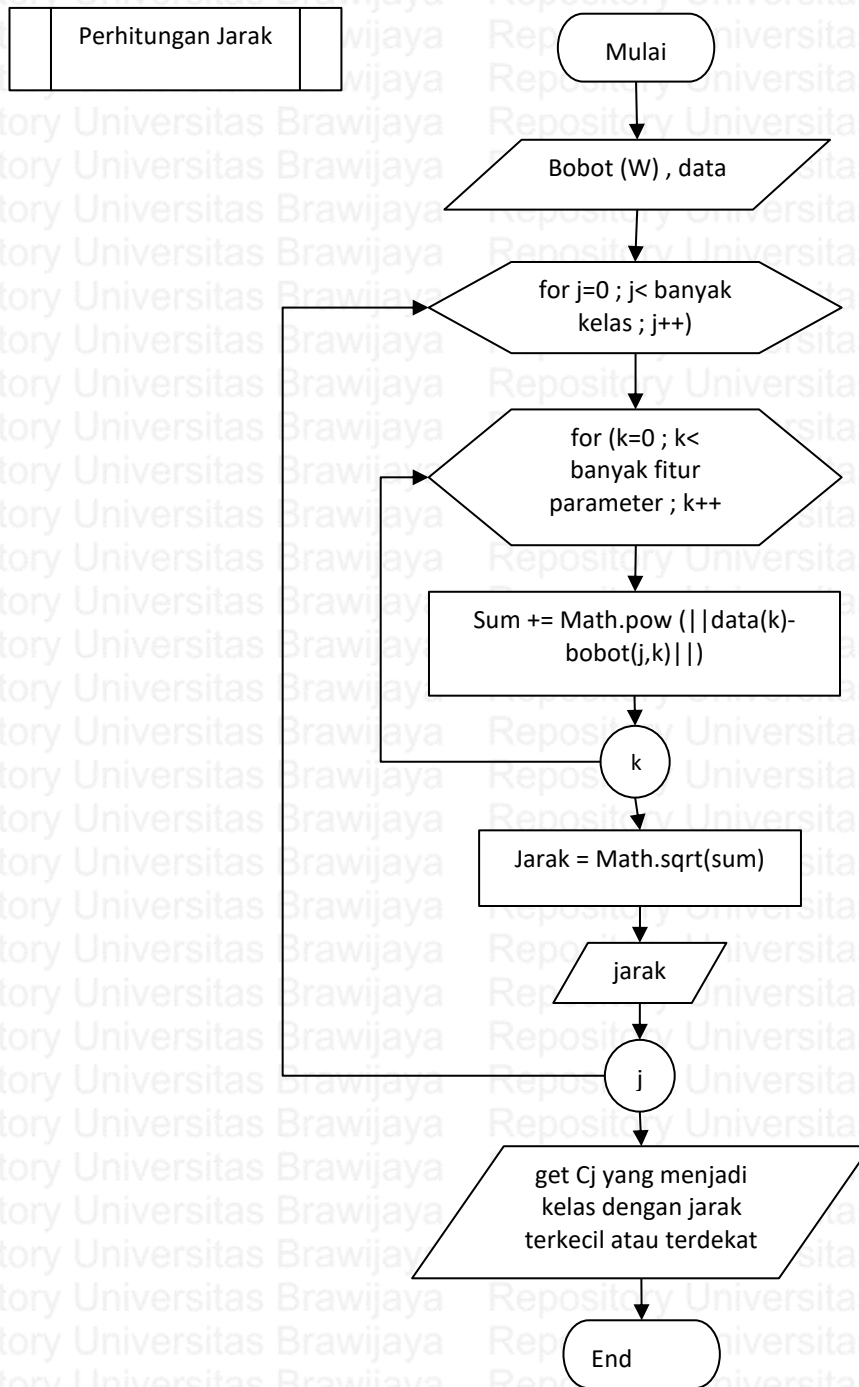
Proses perhitungan jarak dimulai dengan menginisialisasi vektor input data dan bobot sampai menghasilkan nilai jarak. Tahapan proses pencarian jarak dengan LVQ meliputi:

1. Inisialisasi variabel yang menyimpan nilai jarak.
2. Perulangan sebanyak jumlah fitur atau parameter dimana proses perulangan tersebut dilakukan untuk mengetahui jumlah seluruh hasil kuadrat dari perulangan vektor data ke- $n$  dengan bobot ke- $i$ .
3. Keluar dari proses perulangan dan mencari hasil akar dari proses penjumlahan sebelumnya.
4. Mencari jarak minimum untuk menentukan kelas bobot yang baru.

Diagram alir proses pencarian jarak ditunjukkan pada Gambar 4.5.



Gambar 4.4 Diagram Alir Proses Pengujian LVQ



Gambar 4.5 Diagram Alir Proses Perhitungan Jarak

#### 4.5 Perhitungan manual

Pada tahap ini akan dijelaskan proses manualisasi dari klasifikasi kualitas air sungai menggunakan metode LVQ. Proses manualisasi berfungsi untuk mengetahui perkiraan hasil sebelum sistem dibuat, sehingga akan diketahui kesamaan antara hasil manual dengan hasil setelah sistem dibuat mempunyai hasil yang sama. Proses manualisasi dimulai dengan mengambil beberapa data *sample* dari dataset dan membagi dataset tersebut menjadi data latih, data bobot



awal dan data uji. Selanjutnya dilakukan proses pelatihan data latih, pengujian data uji menggunakan metode LVQ dan perhitungan akurasi. Data yang digunakan untuk proses manualisasi yaitu 14 data dengan 3 data sebagai data bobot awal atau *centroid* yang mewakili ketiga kelas klasifikasi kualitas Air Sungai, 6 data latih yang mewakili ketiga kelas klasifikasi kualitas Air Sungai dengan 2 data tiap kelasnya serta 5 data uji.

#### 4.5.1 Inisialisasi bobot awal, data latih dan parameter awal

Tahapan pertama dari proses pelatihan *Learning Vector Quantization* adalah melakukan inisialisasi bobot awal, data latih dan data uji. Bobot awal didapatkan dengan cara mengambil secara acak dari dataset yang ada. Setiap bobot yang diambil harus mewakili masing-masing kelas *output* yang ada. Inisialisasi bobot atau *centroid* ( $W$ ) pada pelatihan manual ini ditunjukkan pada Tabel 4.1.

**Tabel 4.1 Inisialisasi Bobot Awal ( $W$ )**

FISIKA	KIMIA Anorganik			KIMIA Organik			Klasifikasi		
	TSS	BOD	COD	DO	pH	Fenol	Minyak & Lemak	Skor	Target Kelas (T)
35	2,5	11,2	8,9	7,4	0,188	0,8	-6	1	ringan
0	15,35	56,8	11,2	7,5	0,0775	0,8	-30	2	sedang
44,85	12,6	35,4	18,55	8,85	0,1185	4,25	-39	3	berat

Dengan penjelasan sebagai berikut:

$$W_1 = (35 \quad 2,5 \quad 11,2 \quad 8,9 \quad 7,4 \quad 0,188 \quad 0,8 \quad -6),$$

$$W_2 = (0 \quad 15,35 \quad 56,8 \quad 11,2 \quad 7,5 \quad 0,0775 \quad 0,8 \quad -30),$$

$$W_3 = (44,85 \quad 12,6 \quad 35,4 \quad 18,55 \quad 8,85 \quad 0,1185 \quad 4,25 \quad -39).$$

Langkah selanjutnya adalah menentukan data latih yang digunakan untuk menghitung bobot pada *Learning Vector Quantization*. Data latih yang digunakan pada pelatihan ini adalah 6 data latih dari dataset yang diambil secara rata di setiap kelasnya. Setiap kelas diambil 2 data yang ditunjukkan pada tabel 4.2.

**Tabel 4.2 Data Latih ( $x$ )**

FISIKA	KIMIA Anorganik			KIMIA Organik			Klasifikasi		
	TSS	BOD	COD	DO	pH	Fenol	Minyak & Lemak	Skor	Target Kelas (T)
0	10	34,8	15,3	8,05	0,0145	4,5	-42	3	berat
72,75	19,75	62,85	10,45	8,1	0,172	0	-32	3	berat
10	6,2	13,9	10,15	7,55	0,172	1,5	-30	2	sedang
6,2	11,3	35,1	16,925	8,45	0,086	14,25	-30	2	sedang
20	2,85	35,4	13	6,95	0	0,9	-6	1	ringan
44,85	2,4	62,85	6,3	7	0	0,8	-2	1	ringan

Setelah bobot awal dan data latih ditentukan, maka langkah selanjutnya yaitu menginisialisasi parameter masukan awal dalam pelatihan. Parameter tersebut adalah nilai laju pembelajaran ( $\alpha$ ) sebesar ( $\alpha = 0,25$ ) dengan nilai pengurangan  $\alpha$  (*decrement*  $\alpha = 0,05$ ) dan nilai minimum  $\alpha = 0,00001$ , selanjutnya menentukan





nilai awal *Epoch* (perulangan) yang selalu dimulai pada iterasi ke-1 dengan nilai iterasi maksimal (*MaxEpoch* = 2).

#### 4.5.2 Pelatihan dengan metode LVQ

Proses pelatihan diawali dari iterasi (*Epoch*) ke-1, ketika iterasi lebih kecil dari iterasi maksimum atau laju pembelajaran lebih besar dari laju pembelajaran maksimum, maka dilakukan langkah selanjutnya. Untuk vektor pelatihan dihitung sebanyak data latih (*x*) akan ditentukan kelas *Cj* terhadap kelas bobot dengan menghitung jarak *Euclidian Distance* minimum dari  $||x-W_j||$ .

**Iterasi ke-1:**

**Data ke-1 : (0 10 34,8 15,3 8,05 0,0145 4,5 -42) T=3**

- Jarak ke W1 dihitung dengan perhitungan:

$$\sqrt{(0 - 35)^2 + (10 - 2,5)^2 + (34,8 - 11,2)^2 + (15,3 - 8,9)^2 + (8,05 - 7,4)^2 + (0,0145 - 0,188)^2 + (4,5 - 0,8)^2 + ((-42) - (-6))^2}$$

$$= 56,47399935$$

- Jarak ke W2 dihitung dengan perhitungan:

$$\sqrt{(0 - 0)^2 + (10 - 15,35)^2 + (34,8 - 56,8)^2 + (15,3 - 11,2)^2 + (8,05 - 7,5)^2 + (0,0145 - 0,0775)^2 + (4,5 - 0,8)^2 + ((-42) - (-30))^2}$$

$$= 26,21886666$$

- Jarak ke W3 dihitung dengan perhitungan:

$$\sqrt{(0 - 44,85)^2 + (10 - 12,6)^2 + (34,8 - 35,4)^2 + (15,3 - 18,55)^2 + (8,05 - 8,85)^2 + (0,0145 - 0,1185)^2 + (4,5 - 4,25)^2 + ((-42) - (-39))^2}$$

$$= 45,15438313$$

Berdasarkan perhitungan, maka dipilih jarak terdekat, yaitu ke W2 sehingga *Cj* = 2. Sedangkan, kelas target (*T*) data ke-1 adalah 3, maka antara kelas target (*T*) dengan *Cj* terdapat ketidaksesuaian (*T* ≠ *Cj*). Selanjutnya update bobot data W2 menggunakan persamaan (2.3).

$$= \begin{pmatrix} 0 \\ 15,35 \\ 56,8 \\ 11,2 \\ 7,5 \\ 0,0775 \\ 0,8 \\ -30 \end{pmatrix} - 0,25 \times \left[ \begin{pmatrix} 0 \\ 10 \\ 34,8 \\ 15,3 \\ 8,05 \\ 0,0145 \\ 4,5 \\ -42 \end{pmatrix} - \begin{pmatrix} 0 \\ 15,35 \\ 56,8 \\ 11,2 \\ 7,5 \\ 0,0775 \\ 0,8 \\ -30 \end{pmatrix} \right]$$



$$= \begin{pmatrix} 0 \\ 16,6875 \\ 62,3 \\ 10,175 \\ 7,3625 \\ 0,09325 \\ -0,125 \\ -27 \end{pmatrix}$$

Sehingga diperoleh bobot baru:

W1	35	2,5	11,2	8,9	7,4	0,188	0,8	-6
W2	0	16,6875	62,3	10,175	7,3625	0,09325	-0,125	-27
W3	44,85	12,6	35,4	18,55	8,85	0,1185	4,25	-39

**Data ke-2 : (72,75 19,75 62,85 10,45 8,1 0,172 0 -32) T=3**

- Jarak ke W1 dihitung dengan perhitungan:

$$\sqrt{(72,5 - 35)^2 + (19,75 - 2,5)^2 + (62,85 - 11,2)^2 + (10,45 - 8,9)^2 + (8,1 - 7,4)^2 + (0,172 - 0,188)^2 + (0 - 0,8)^2 + ((-32) - (-6))^2}$$

$$= 71,20309162$$

- Jarak ke W2 dihitung dengan perhitungan:

$$\sqrt{(72,5 - 0)^2 + (19,75 - 16,6875)^2 + (62,85 - 62,3)^2 + (10,45 - 10,175)^2 + (8,1 - 7,3625)^2 + (0,172 - 0,09325)^2 + (0 - (-0,125))^2 + ((-32) - (-27))^2}$$

$$= 72,99236442$$

- Jarak ke W3 dihitung dengan perhitungan:

$$\sqrt{(72,5 - 44,85)^2 + (19,75 - 12,6)^2 + (62,85 - 35,4)^2 + (10,45 - 18,55)^2 + (8,1 - 8,85)^2 + (0,172 - 0,1185)^2 + (0 - 4,25)^2 + ((-32) - (-39))^2}$$

$$= 41,4279237$$

Berdasarkan perhitungan, maka dipilih jarak terdekat, yaitu ke W3 sehingga  $C_j = 3$ . Kelas target (T) data ke-2 adalah 3, maka antara kelas target (T) dengan  $C_j$  terdapat kesesuaian ( $T=C_j$ ). Selanjutnya update bobot data W3 menggunakan persamaan (2.2).



$$= \begin{pmatrix} 44,85 \\ 12,6 \\ 35,4 \\ 18,55 \\ 8,85 \\ 0,1185 \\ 4,25 \\ -39 \end{pmatrix} + 0,25 \times \begin{bmatrix} \begin{pmatrix} 72,25 \\ 19,75 \\ 62,85 \\ 10,45 \\ 8,1 \\ 0,172 \\ 0 \\ -32 \end{pmatrix} - \begin{pmatrix} 44,85 \\ 12,6 \\ 35,4 \\ 18,55 \\ 8,85 \\ 0,1185 \\ 4,25 \\ -39 \end{pmatrix} \end{bmatrix}$$

$$= \begin{pmatrix} 51,825 \\ 14,3875 \\ 42,2625 \\ 16,525 \\ 8,6625 \\ 0,131875 \\ 3,1875 \\ -37,25 \end{pmatrix}$$

Sehingga diperoleh bobot baru:

W1	35	2,5	11,2	8,9	7,4	0,188	0,8	-6
W2	0	16,6875	62,3	10,175	7,3625	0,09325	-0,125	-27
W3	51,825	14,3875	42,2625	16,525	8,6625	0,131875	3,1875	-37,25

**Data ke-3 : (10 6,2 13,9 10,15 7,55 0,172 1,5 -30) T=2**

- Jarak ke W1 dihitung dengan perhitungan:

$$\sqrt{(10 - 35)^2 + (6,2 - 2,5)^2 + (13,9 - 11,2)^2 + (10,15 - 8,9)^2 + (7,55 - 7,4)^2 + (0,172 - 0,188)^2 + (1,5 - 0,8)^2 + ((-30) - (-6))^2}$$

$$= 34,98650105$$

- Jarak ke W2 dihitung dengan perhitungan:

$$\sqrt{(10 - 0)^2 + (6,2 - 16,6875)^2 + (13,9 - 62,3)^2 + (10,15 - 10,175)^2 + (7,55 - 7,3625)^2 + (0,172 - 0,09325)^2 + (1,5 - (-0,125))^2 + ((-30) - (-27))^2}$$

$$= 50,63822927$$



- Jarak ke W3 dihitung dengan perhitungan:

$$\sqrt{(10 - 51,825)^2 + (6,2 - 14,3875)^2 + (13,9 - 42,2625)^2 + (10,15 - 16,525)^2 + (7,55 - 8,6625)^2 + (0,172 - 0,131875)^2 + (1,5 - 3,1875)^2 + ((-30) - (-37,25))^2}$$

$$= 52,13527822$$

Berdasarkan perhitungan, maka dipilih jarak terdekat, yaitu ke W1 sehingga  $C_j = 1$ . Sedangkan, kelas target (T) data ke-3 adalah 2, maka antara kelas target (T) dengan  $C_j$  terdapat ketidaksesuaian ( $T \neq C_j$ ). Selanjutnya update bobot data W1 menggunakan persamaan (2.3).

Sehingga diperoleh bobot baru:

W1	41,25	1,575	10,525	8,5875	7,3625	0,192	0,625	0
W2	0	16,6875	62,3	10,175	7,3625	0,09325	-0,125	-27
W3	51,825	14,3875	42,2625	16,525	8,6625	0,131875	3,1875	-37,25

**Data ke-4 : (6,2 11,3 35,1 16,925 8,45 0,086 14,25 -30) T=2**

- Jarak ke W1 dihitung dengan perhitungan:

$$\sqrt{(6,2 - 41,25)^2 + (11,3 - 1,575)^2 + (35,1 - 10,525)^2 + (16,925 - 8,5875)^2 + (8,45 - 7,3625)^2 + (0,086 - 0,192)^2 + (14,25 - 0,625)^2 + ((-30) - 0)^2}$$

$$= 55,5279855$$

- Jarak ke W2 dihitung dengan perhitungan:

$$\sqrt{(6,2 - 0)^2 + (11,3 - 16,6875)^2 + (35,1 - 62,3)^2 + (16,925 - 10,175)^2 + (8,45 - 7,3625)^2 + (0,086 - 0,09325)^2 + (14,25 - (-0,125))^2 + ((-30) - (-27))^2}$$

$$= 32,70613077$$

- Jarak ke W3 dihitung dengan perhitungan:

$$\sqrt{(6,2 - 51,825)^2 + (11,3 - 14,3875)^2 + (35,1 - 42,2625)^2 + (16,925 - 16,525)^2 + (8,45 - 8,6625)^2 + (0,086 - 0,131875)^2 + (14,25 - 3,1875)^2 + ((-30) - (-37,25))^2}$$

$$= 48,14170079$$

Berdasarkan perhitungan, maka dipilih jarak terdekat, yaitu ke W2 sehingga  $C_j = 2$ . Kelas target (T) data ke-4 adalah 2, maka antara kelas target (T) dengan  $C_j$  terdapat kesesuaian ( $T = C_j$ ). Selanjutnya update bobot data W2 menggunakan persamaan (2.2).



Sehingga diperoleh bobot baru:

W1	41,25	1,575	10,525	8,5875	7,3625	0,192	0,625	0
W2	1,55	15,34063	55,5	11,8625	7,634375	0,091438	3,46875	-27,75
W3	51,825	14,3875	42,2625	16,525	8,6625	0,131875	3,1875	-37,25

Data ke-5 : (20 2,85 35,4 13 6,95 0 0,9 -6) T=1

- Jarak ke W1 dihitung dengan perhitungan:

$$\sqrt{(20 - 41,25)^2 + (2,85 - 1,575)^2 + (35,4 - 10,525)^2 + (13 - 8,5875)^2 + (6,95 - 7,3625)^2 + (0 - 0,192)^2 + (0,9 - 0,625)^2 + ((-6) - 0)^2}$$

$$= 33,58134231$$

- Jarak ke W2 dihitung dengan perhitungan:

$$\sqrt{(20 - 1,55)^2 + (2,85 - 15,34063)^2 + (35,4 - 55,5)^2 + (13 - 11,8625)^2 + (6,95 - 7,634375)^2 + (0 - 0,91438)^2 + (0,9 - 3,46875)^2 + ((-6) - (-27,75))^2}$$

$$= 37,17337523$$

- Jarak ke W3 dihitung dengan perhitungan:

$$\sqrt{(6,2 - 51,825)^2 + (11,3 - 14,3875)^2 + (35,1 - 42,2625)^2 + (16,925 - 16,525)^2 + (8,45 - 8,6625)^2 + (0,086 - 0,131875)^2 + (14,25 - 3,1875)^2 + ((-30) - (-37,25))^2}$$

$$= 46,79967164$$

Berdasarkan perhitungan, maka dipilih jarak terdekat, yaitu ke W1 sehingga Cj = 1. Kelas target (T) data ke-5 adalah 1, maka antara kelas target (T) dengan Cj terdapat kesesuaian (T=Cj). Selanjutnya update bobot data W1 menggunakan persamaan (2.2).

Sehingga diperoleh bobot baru:

W1	35,9375	1,89375	16,74375	9,690625	7,259375	0,144	0,69375	-1,5
W2	1,55	15,34063	55,5	11,8625	7,634375	0,091438	3,46875	-27,75
W3	51,825	14,3875	42,2625	16,525	8,6625	0,131875	3,1875	-37,25



Data ke-6 : (44,85 2,4 62,85 6,3 7 0 0,8 -2) T=1

- Jarak ke W1 dihitung dengan perhitungan:

$$\sqrt{(44,85 - 35,9375)^2 + (2,4 - 1,89375)^2 + (62,85 - 16,74375)^2 + (6,3 - 9,690625)^2 + (7 - 7,259375)^2 + (0 - 0,144)^2 + (0,8 - 0,69375)^2 + ((-2) - (-1,5))^2}$$

$$= 47,08843672$$

- Jarak ke W2 dihitung dengan perhitungan:

$$\sqrt{(44,85 - 1,55)^2 + (2,4 - 15,34063)^2 + (62,85 - 55,5)^2 + (6,3 - 11,8625)^2 + (7 - 7,634375)^2 + (0 - 0,91438)^2 + (0,8 - 3,46875)^2 + ((-2) - (-27,75))^2}$$

$$= 52,89526634$$

- Jarak ke W3 dihitung dengan perhitungan:

$$\sqrt{(44,85 - 51,825)^2 + (2,4 - 14,3875)^2 + (62,85 - 42,2625)^2 + (6,3 - 16,525)^2 + (7 - 8,6625)^2 + (0 - 0,131875)^2 + (0,8 - 3,1875)^2 + ((-2) - (-37,25))^2}$$

$$= 44,40484789$$

Berdasarkan perhitungan, maka dipilih jarak terdekat, yaitu ke W3 sehingga  $C_j = 3$ . Sedangkan, kelas target (T) data ke-6 adalah 1, maka antara kelas target (T) dengan  $C_j$  terdapat ketidaksesuaian ( $T \neq C_j$ ). Selanjutnya update bobot data W3 menggunakan persamaan (2.3).

Sehingga diperoleh bobot baru:

W1	35,9375	1,89375	16,74375	9,690625	7,259375	0,144	0,69375	-1,5
W2	1,55	15,34063	55,5	11,8625	7,634375	0,091438	3,46875	-27,75
W3	53,56875	17,38438	37,11563	19,08125	9,078125	0,164844	3,784375	-46,0625

Berdasarkan proses pelatihan yang sudah dilakukan, kemudian dilanjutkan ke *Epoch* selanjutnya ( $Epoch = Epoch + 1$ ) yaitu *Epoch* ke-2 (iterasi ke-2). Selanjutnya, dengan menggunakan cara perhitungan yang sama dengan iterasi ke-1 tetapi bobot yang digunakan adalah bobot akhir dari hasil perhitungan pada iterasi sebelumnya dan mengurangi laju pembelajaran melalui persamaan 2.4, yaitu  $\alpha = 0,25 - 0,25 * 0,05$  sehingga  $\alpha$  adalah 0,2375.

**Iterasi ke-2:**

Nilai  $\alpha = 0,2375$  dengan bobot (W) berdasarkan pelatihan iterasi sebelumnya:

W1	35,9375	1,89375	16,74375	9,690625	7,259375	0,144	0,69375	-1,5
W2	1,55	15,34063	55,5	11,8625	7,634375	0,091438	3,46875	-27,75
W3	53,56875	17,38438	37,11563	19,08125	9,078125	0,164844	3,784375	-46,0625

**Data ke-1 : (0 10 34,8 15,3 8,05 0,0145 4,5 -42) T=3**

- Jarak ke W1 dihitung dengan perhitungan:

$$\sqrt{(0 - 35,9375)^2 + (10 - 1,89375)^2 + (34,8 - 16,74375)^2 + (15,3 - 9,690625)^2 + (8,05 - 7,259375)^2 + (0,0145 - 0,144)^2 + (4,5 - 0,69375)^2 + ((-42) - (-1,5))^2}$$

$$= 58,0524577$$

- Jarak ke W2 dihitung dengan perhitungan:

$$\sqrt{(0 - 1,55)^2 + (10 - 15,34063)^2 + (34,8 - 55,5)^2 + (15,3 - 11,8625)^2 + (8,05 - 7,634375)^2 + (0,0145 - 0,091438)^2 + (4,5 - 3,46875)^2 + ((-42) - (-27,75))^2}$$

$$= 25,99107196$$

- Jarak ke W3 dihitung dengan perhitungan:

$$\sqrt{(0 - 53,56875)^2 + (10 - 17,38438)^2 + (34,8 - 37,11563)^2 + (15,3 - 19,08125)^2 + (8,05 - 9,078125)^2 + (0,0145 - 0,164844)^2 + (4,5 - 3,784375)^2 + ((-42) - (-46,0625))^2}$$

$$= 54,42330026$$

Berdasarkan perhitungan, maka dipilih jarak terdekat, yaitu ke W2 sehingga  $C_j = 2$ . Sedangkan, kelas target (T) data ke-1 adalah 3, maka antara kelas target (T) dengan  $C_j$  terdapat ketidaksesuaian ( $T \neq C_j$ ). Selanjutnya update bobot data W2 menggunakan persamaan (2.3).

$$= \begin{pmatrix} 1,55 \\ 15,34063 \\ 55,5 \\ 11,8625 \\ 7,634375 \\ 0,091438 \\ 3,46875 \\ -27,75 \end{pmatrix} - 0,2375 \times \begin{pmatrix} 0 \\ 10 \\ 34,8 \\ 15,3 \\ 8,05 \\ 0,0145 \\ 4,5 \\ -42 \end{pmatrix} - \begin{pmatrix} 1,55 \\ 15,34063 \\ 55,5 \\ 11,8625 \\ 7,634375 \\ 0,091438 \\ 3,46875 \\ -27,75 \end{pmatrix}$$



$$= \begin{pmatrix} 1,918125 \\ 16,609023 \\ 60,41625 \\ 11,04609 \\ 7,535664 \\ 0,10971 \\ 3,223828 \\ -24,3656 \end{pmatrix}$$

Sehingga diperoleh bobot baru:

W1	35.9375	1.89375	16.74375	9.690625	7.259375	0.144	0.69375	-1.5
W2	1.918125	16.60902	60.41625	11.04609	7.535664	0.10971	3.223828	-24.3656
W3	53.56875	17.38438	37.11563	19.08125	9.078125	0.164844	3.784375	-46.0625

Selanjutnya proses yang sama dilakukan terhadap pelatihan data ke-2 sampai data ke-6. Sehingga bobot yang didapatkan dari pelatihan data ke-2 adalah sebagai berikut:

W1	35.9375	1.89375	16.74375	9.690625	7.259375	0.144	0.69375	-1.5
W2	1.918125	16.60902	60.41625	11.04609	7.535664	0.10971	3.223828	-24.3656
W3	58.1243	17.94621	43.22754	17.03133	8.84582	0.166543	2.885586	-42.7227

bobot yang didapatkan dari pelatihan data ke-3 adalah sebagai berikut:

W1	42.09766	0.871016	17.41914	9.581523	7.190352	0.13735	0.502266	5.26875
W2	1.918125	16.60902	60.41625	11.04609	7.535664	0.10971	3.223828	-24.3656
W3	58.1243	17.94621	43.22754	17.03133	8.84582	0.166543	2.885586	-42.7227

bobot yang didapatkan dari pelatihan data ke-4 adalah sebagai berikut:

W1	42.09766	0.871016	17.41914	9.581523	7.190352	0.13735	0.502266	5.26875
W2	2.93507	15.34813	54.40364	12.44233	7.752819	0.104079	5.842544	-25.7038
W3	58.1243	17.94621	43.22754	17.03133	8.84582	0.166543	2.885586	-42.7227

bobot yang didapatkan dari pelatihan data ke-5 adalah sebagai berikut:

W1	36.84946	1.341024	21.68959	10.39341	7.133268	0.104729	0.596728	2.592422
W2	2.93507	15.34813	54.40364	12.44233	7.752819	0.104079	5.842544	-25.7038
W3	58.1243	17.94621	43.22754	17.03133	8.84582	0.166543	2.885586	-42.7227

bobot yang didapatkan dari pelatihan data ke-6 adalah sebagai berikut:

W1	38.74959	1.592531	31.46519	9.421226	7.101617	0.079856	0.645005	1.501722
W2	2.93507	15.34813	54.40364	12.44233	7.752819	0.104079	5.842544	-25.7038
W3	58.1243	17.94621	43.22754	17.03133	8.84582	0.166543	2.885586	-42.7227

*Epoch* selanjutnya adalah iterasi ke-3, tetapi karena nilai iterasi lebih besar dibandingkan nilai iterasi maksimum ( $maxEpoch = 2$ ), maka proses pelatihan data telah selesai dilakukan. Sehingga, dilanjutkan ke proses pengujian data dengan menggunakan nilai bobot dari bobot akhir pada proses pelatihan. Bobot akhir dari pelatihan ditunjukkan pada Tabel 4.3.



**Tabel 4.3 Tabel Bobot Akhir atau Optimal (W)**

FISIKA	KIMIA Anorganik			KIMIA Organik			Klasifikasi	
TSS	BOD	COD	DO	pH	Fenol	Minyak & Lemak	Skor	Target Kelas (T)
38.74959	1.592531	31.46519	9.421226	7.101617	0.079856	0.645005	1.501722	1
2.93507	15.34813	54.40364	12.44233	7.752819	0.104079	5.842544	-25.7038	2
58.1243	17.94621	43.22754	17.03133	8.84582	0.166543	2.885586	-42.7227	3

#### 4.5.3 Pengujian dengan metode LVQ

Proses pengujian data terdiri dari beberapa tahapan, yaitu dimulai dengan menentukan data bobot atau centroid (W) berdasarkan hasil pelatihan, menginisialisasi data uji dan menentukan jarak terdekat tiap data uji, sehingga akan didapatkan kelas klasifikasi kualitas air sungai. Data uji yang digunakan pada pengujian ini adalah 5 data uji dari dataset yang diambil secara acak yang ditunjukkan pada Tabel 4.4.

**Tabel 4.4 Data Uji**

FISIKA	KIMIA Anorganik			KIMIA Organik			Klasifikasi		
TSS	BOD	COD	DO	pH	Fenol	Minyak & Lemak	Skor	Keterangan	
155	13,15	44	29	7,5	0,0775	6	-35	berat	3
117	8,5	47	10	8,05	0,0145	3,5	-45	berat	3
24,25	1,925	27,803	12,7	7,8	0,0001	0,4	-10	ringan	1
7,1	3,6	9,85	5,85	7,35	0,104	1,3	-26	sedang	2
15,7	5,85	19,6	6,7	7,2	0,187	2,9	-26	sedang	2

**Data Uji ke-1 : (155 13,15 54 29 7,5 0,0775 6 -35)**

- Jarak ke W1 dihitung dengan perhitungan:

$$\sqrt{(155 - 38,7495)^2 + (13,13 - 1,59253)^2 + (54 - 31,46519)^2 + (29 - 9,421226)^2 + (7,5 - 7,101617)^2 + (0,0775 - 0,079856)^2 + (6 - 0,645005)^2 + ((-35) - 1,501722)^2}$$

$$= 124,6972052$$

- Jarak ke W2 dihitung dengan perhitungan:

$$\sqrt{(155 - 2,93507)^2 + (13,13 - 15,34813)^2 + (54 - 54,40364)^2 + (29 - 12,44233)^2 + (7,5 - 7,75281)^2 + (0,0775 - 0,10407)^2 + (6 - 5,84254)^2 + ((-35) - (-25,7038))^2}$$

$$= 153,6146986$$



- Jarak ke W3 dihitung dengan perhitungan:

$$\sqrt{(155 - 58,1243)^2 + (13,13 - 17,94621)^2 + (54 - 43,22754)^2 + (29 - 17,03133)^2 + (7,5 - 8,84582)^2 + (0,0775 - 0,166543359)^2 + (6 - 2,885586)^2 + ((-35) - (-42,7227))^2}$$

$$= 98,09642936$$

Berdasarkan perhitungan pengujian pada data ke-1, maka dipilih jarak terdekat, yaitu ke W3 sehingga  $C_j = 3$ .

**Data Uji ke-2 : (117      8,5      47      10      8,05      0,0145      3,5      -45)**

- Jarak ke W1 dihitung dengan perhitungan:

$$\sqrt{(117 - 40,3527)^2 + (8,5 - 1,683592)^2 + (47 - 39,75844)^2 + (10 - 9,421226)^2 + (8,05 - 7,054962)^2 + (0,0145 - 0,079856)^2 + (3,5 - 0,645005)^2 + ((-45) - 1,501722)^2}$$

$$= 92,64971565$$

- Jarak ke W2 dihitung dengan perhitungan:

$$\sqrt{(117 - 2,93507)^2 + (8,5 - 15,34813)^2 + (47 - 54,40364)^2 + (10 - 12,44233)^2 + (8,05 - 7,101617)^2 + (0,0145 - 0,10407)^2 + (3,5 - 5,84254)^2 + ((-45) - (-25,7038))^2}$$

$$= 116,1740572$$

- Jarak ke W3 dihitung dengan perhitungan:

$$\sqrt{(117 - 58,1243)^2 + (8,5 - 17,94621)^2 + (47 - 43,22754)^2 + (10 - 17,03133)^2 + (8,05 - 8,84582)^2 + (0,0145 - 0,166543359)^2 + (3,5 - 2,885586)^2 + ((-45) - (-42,7227))^2}$$

$$= 60,21188091$$

Berdasarkan perhitungan pengujian pada data ke-2, maka dipilih jarak terdekat, yaitu ke W3 sehingga  $C_j = 3$ .

Selanjutnya proses yang sama dilakukan untuk data uji ke-3 sampai data ke-5, sehingga dihasilkan data hasil klasifikasi berdasarkan pengujian pada Tabel 4.5.

Tabel 4.5 Data Hasil klasifikasi perhitungan manual

Data Hasil Pengujian					
Data Uji	Jarak Euclidean 1	Jarak Euclidean 2	Jarak Euclidean 3	Kelas Aktual	Hasil Pengujian
Data Uji 1	124.697	153,614	98,096	3	3
Data Uji 2	92.649	116,174	60,211	3	3
Data Uji 3	19.166	40,2294	52,334	1	1
Data Uji 4	47.355	46,954	65,823	2	2
Data Uji 5	38.197	38.813	53.791	2	1

#### 4.5.4 Perhitungan akurasi

Setelah dilakukan pengujian dengan menggunakan algoritma LVQ, maka perhitungan akurasi secara umum dapat dilakukan melalui perhitungan sebagai berikut:

$$\text{Akurasi} = \frac{\text{Jumlah data yang relevan}}{\text{Jumlah seluruh data uji}} \times 100\%$$

Jumlah data yang relevan adalah jumlah kelas *output* dari hasil pengujian yang sama atau relevan dengan kelas *output* dari data aktual. Berdasarkan pengujian data secara manual, maka didapatkan akurasi sistem sebagai berikut.

$$\text{Akurasi} = \frac{4}{5} \times 100 = 80 \%$$

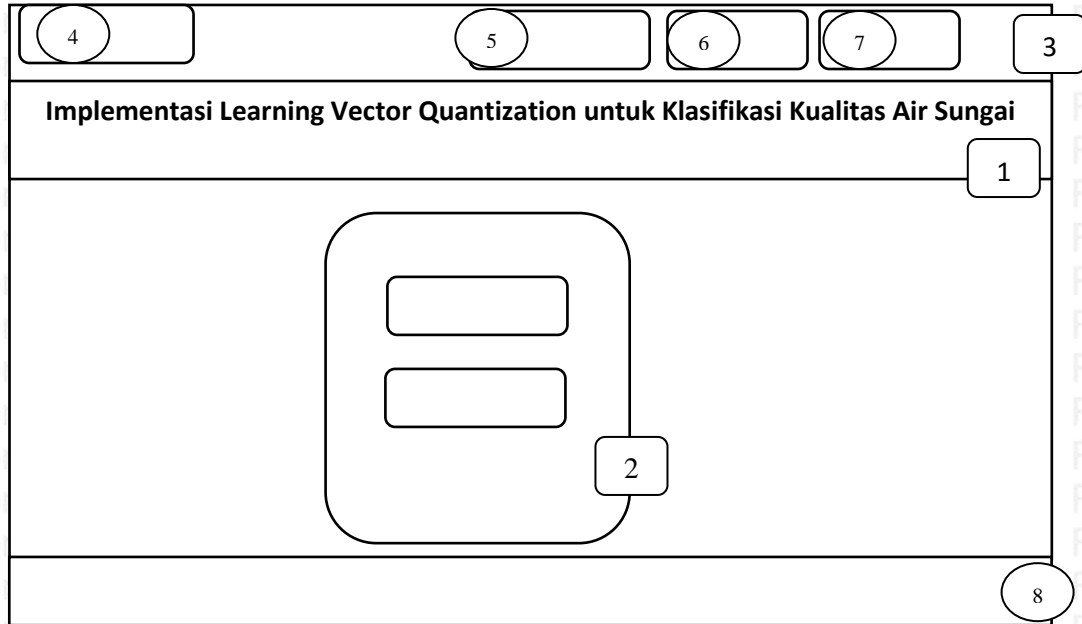
Berdasarkan hasil perhitungan, akurasi yang diperoleh dinyatakan dalam persentase yaitu 80%.

#### 4.6 Perancangan antarmuka

Perancangan antarmuka bertujuan untuk memberikan gambaran umum tentang tampilan sistem yang dibangun, sehingga dapat dijadikan sebagai acuan dalam implementasi sistem dan memudahkan implementasi sistem. Antarmuka dirancang berdasarkan kebutuhan yang harus dipenuhi dalam perhitungan klasifikasi kualitas air sungai menggunakan metode LVQ berupa halaman-halaman antarmuka.

##### 4.6.1 Antarmuka halaman beranda

Halaman beranda atau halaman utama adalah halaman awal ketika aplikasi pertama kali dijalankan. Halaman Beranda berisi judul sistem, logo Universitas Brawijaya, *form login* untuk masuk kedalam sistem, informasi mengenai petunjuk penggunaan sistem klasifikasi air sungai dengan algoritma *Learning Vector Quantization*, informasi mengenai pembuatan sistem dan menu-menu yang dijalankan. Perancangan halaman beranda dapat dilihat pada Gambar 4.6.



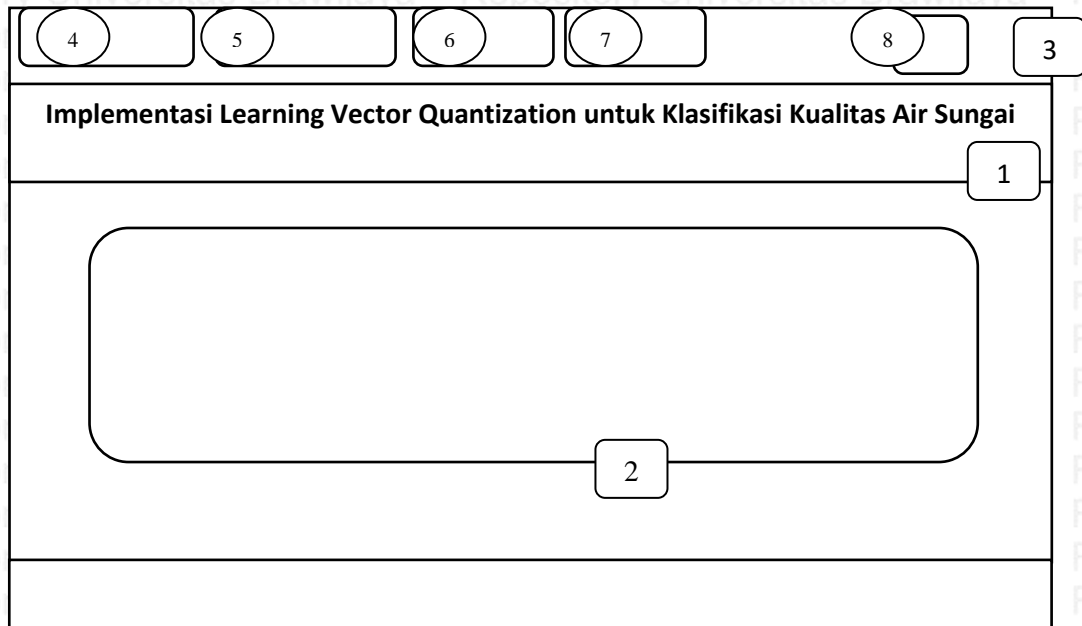
**Gambar 4.6 Antarmuka Halaman Beranda Utama**

Keterangan Gambar 4.6.

1. Judul sistem.
2. Panel untuk menampilkan form *login*.
3. *Tabbed pane* untuk menampilkan menu bar.
4. *Button* halaman beranda dan nama sistem.
5. *Button* halaman beranda.
6. *Button* halaman petunjuk penggunaan.
7. *Button* halaman informasi tentang sistem.
8. *Footer* yang berisi nama fakultas dan *copyright*.

#### 4.6.2 Antarmuka halaman beranda admin

Halaman beranda admin atau halaman utama admin adalah halaman awal ketika aplikasi dijalankan pertama kali menggunakan hak akses sebagai admin. Halaman beranda admin berisi judul sistem, logo Universitas Brawijaya, *button logout* untuk keluar dari hak akses sistem, informasi mengenai wewenang dan petunjuk penggunaan sistem klasifikasi air sungai dengan algoritma *Learning Vector Quantization*, informasi mengenai parameter sistem dan menu-menu yang dijalankan. Perancangan halaman beranda admin dapat dilihat pada Gambar 4.7.



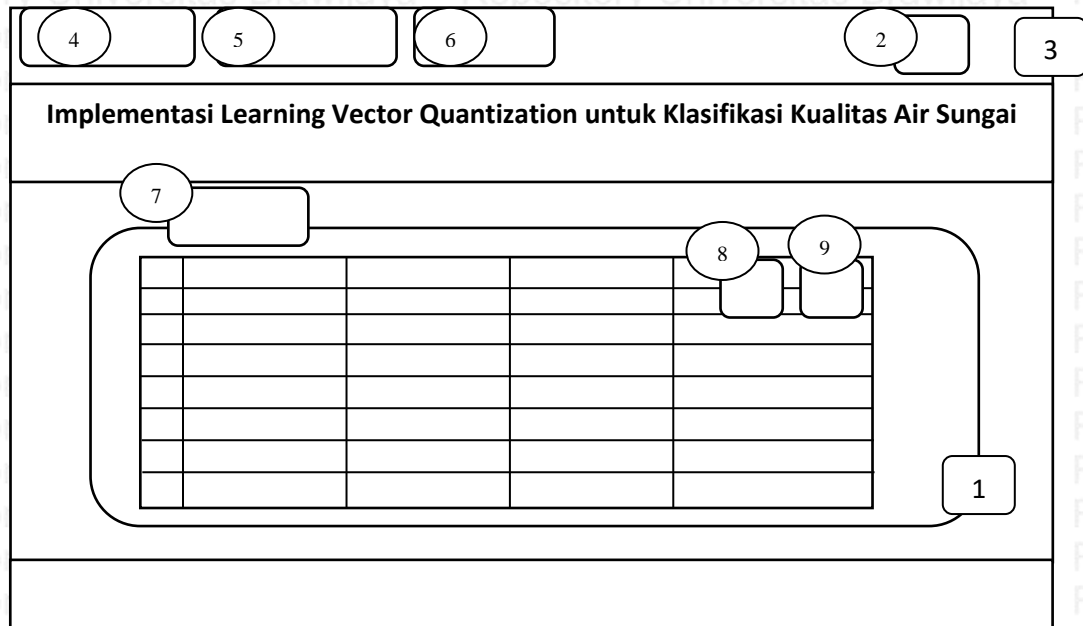
**Gambar 4.7 Antarmuka Halaman Beranda Admin**

Keterangan Gambar 4.7.

1. Judul sistem.
2. Panel yang menampilkan informasi admin.
3. *Tabbed pane* untuk menampilkan menu bar.
4. *Button* halaman beranda dan nama hak akses yang sedang aktif.
5. *Button* halaman beranda.
6. *Button* halaman informasi parameter.
7. *Button* halaman dataset klasifikasi kualitas air sungai.
8. *Button* untuk menampilkan form *logout*.

#### 4.6.3 Antarmuka Halaman CRUD data admin

Halaman data admin atau halaman (*Create, Read, Update* dan *Delete*) CRUD admin adalah halaman khusus untuk membaca data, menambahkan data, melakukan perubahan dan hapus dataset klasifikasi kualitas air sungai yang disimpan pada *database* dan digunakan sebagai perhitungan klasifikasi sebagai hak akses admin. Halaman ini diperlukan karena dibutuhkannya akses, perubahan dan pengolahan dataset yang disimpan pada *database*. Perancangan halaman beranda dapat dilihat pada Gambar 4.8.



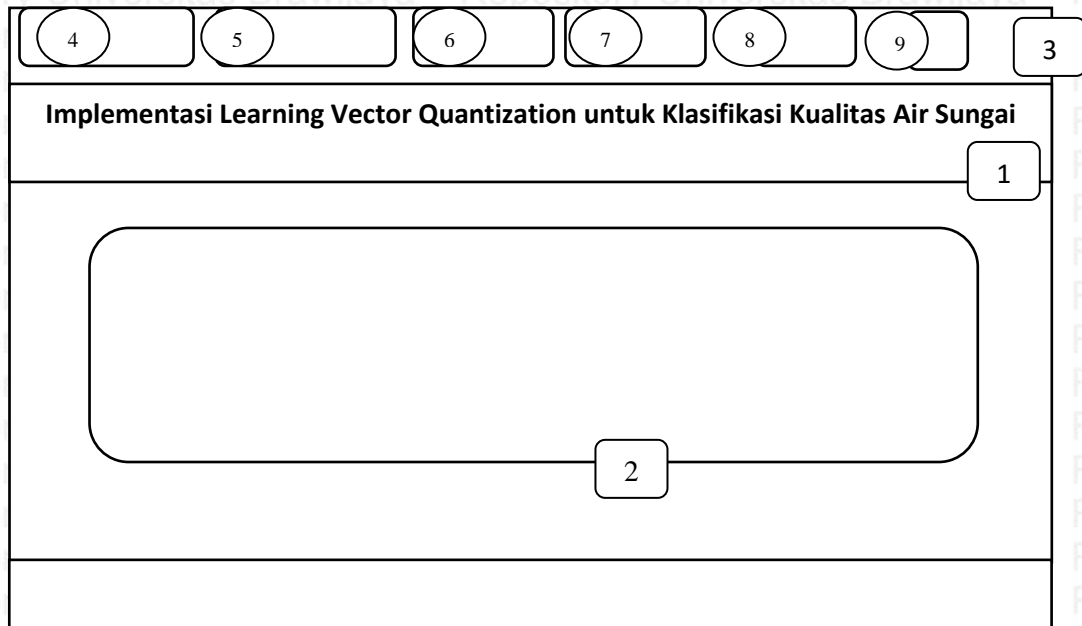
**Gambar 4.8 Antarmuka Halaman Data Admin**

Keterangan Gambar 4.8.

1. Panel yang menampilkan dataset kualitas air sungai.
2. *Button* untuk menampilkan form *logout*.
3. *Tabbed pane* untuk menampilkan menu bar.
4. *Button* halaman beranda dan nama hak akses yang sedang aktif.
5. *Button* halaman informasi parameter.
6. *Button* halaman dataset klasifikasi kualitas air sungai
7. *Button* untuk menambah dataset kualitas air sungai.
8. *Button* untuk *update* dataset kualitas air sungai.
9. *Button* untuk menghapus dataset kualitas air sungai.

#### 4.6.4 Antarmuka halaman beranda *user* (pengguna)

Halaman beranda *user* adalah halaman awal ketika aplikasi dijalankan pertama kali menggunakan hak akses sebagai *user*. Halaman beranda *user* berisi judul sistem, logo Universitas Brawijaya, *button logout* untuk keluar dari hak akses sistem, informasi mengenai wewenang dan petunjuk penggunaan sistem klasifikasi air sungai dengan algoritma *Learning Vector Quantization*, informasi mengenai parameter sistem, *input* perbandingan data, pelatihan data, pengujian data hasil klasifikasi dan menu-menu yang dijalankan. Perancangan halaman beranda admin dapat dilihat pada Gambar 4.9.



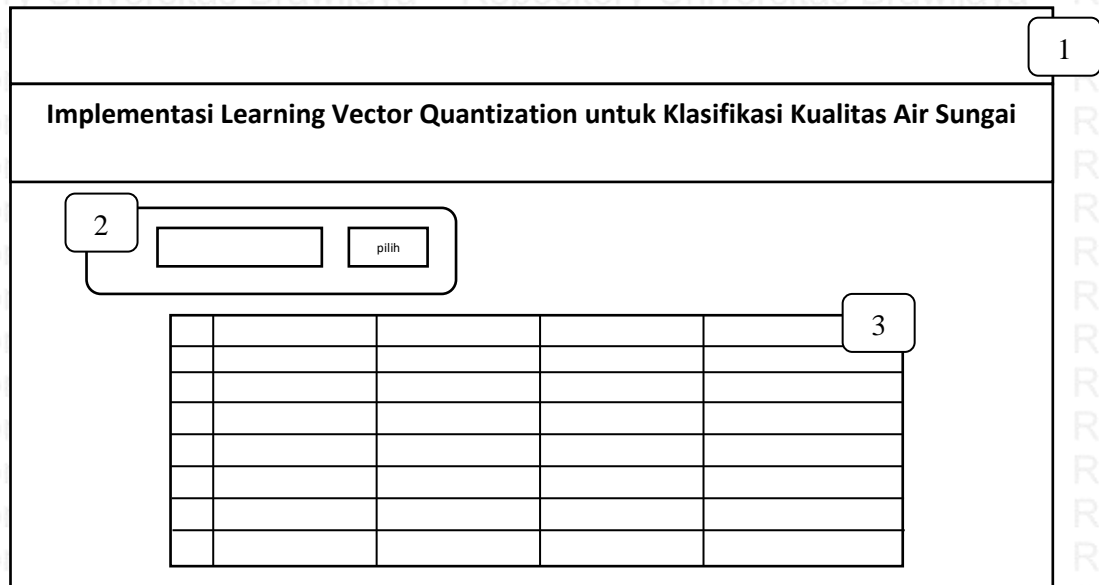
**Gambar 4.9** Antarmuka Halaman Beranda *User*

Keterangan Gambar 4.9.

1. Judul sistem.
2. Panel yang menampilkan informasi *user*.
3. *Tabbed pane* untuk menampilkan menu bar.
4. *Button* halaman melihat dataset kualitas air sungai.
5. *Button* halaman analisis data.
6. *Button* halaman pelatihan data.
7. *Button* halaman pengujian dan klasifikasi data.
8. *Button* halaman pengujian dari manualisasi klasifikasi data.
9. *Button* untuk menampilkan form *logout*.

#### 4.6.5 Antarmuka Halaman *generate* dan analisis data

Halaman *input* dan analisis data adalah halaman untuk memasukkan nilai jumlah perbandingan *dataset* atau data latih dan data uji untuk dilakukan pelatihan dan pengujian. Setelah dataset berhasil diakses dari *database* sistem, maka akan muncul menu pilihan untuk menentukan persentase perbandingan jumlah data latih dengan jumlah data uji. Setelah ditentukan perbandingan antara jumlah data latih dan data uji, maka akan muncul kolom yang menampilkan tabel data latih, data uji dan data bobot awal yang dipilih secara *random*. Perancangan halaman input data dapat dilihat pada Gambar 4.10.



**Gambar 4.10** Antarmuka Halaman *generate Data*

Keterangan gambar 4.10.

1. *Tabbed pane* untuk menampilkan nama sistem dan menu bar.
2. Panel untuk memilih dan memasukkan perbandingan *dataset*.
3. Panel untuk menampilkan data *training*, data *testing* dan data *centroid* atau bobot awal.

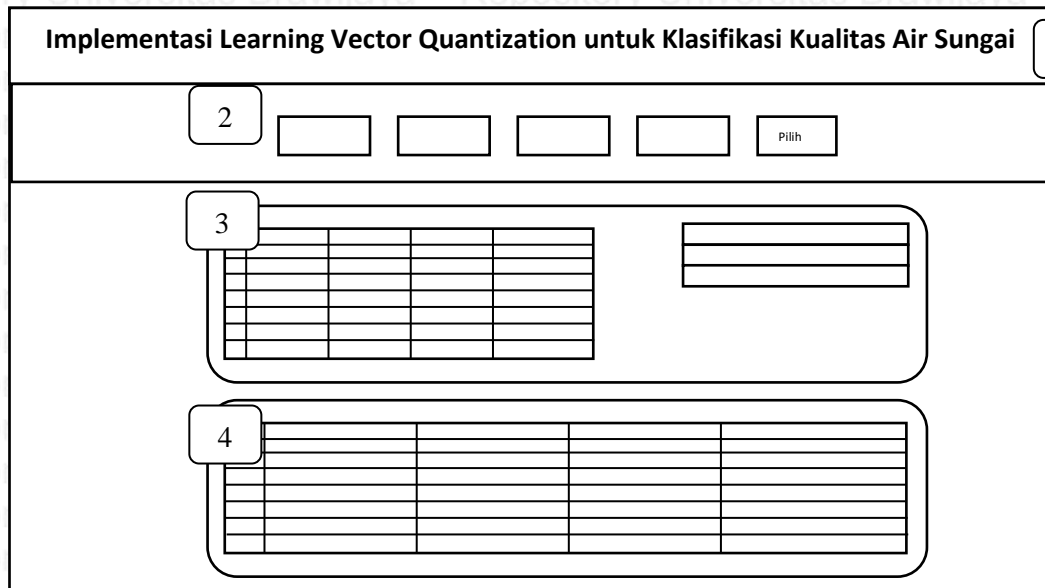
#### 4.6.6 Antarmuka halaman pelatihan

Halaman pelatihan memuat proses dan hasil dari pelatihan yang telah dilakukan. Tampilan dari hasil berupa data tabel. Pada halaman ini akan ditampilkan tabel bobot awal, tabel data latih, tabel perhitungan jarak *euclidian*, tabel perubahan bobot dan tabel bobot optimal. Perancangan halaman pelatihan dapat dilihat pada Gambar 4.11.

#### 4.6.7 Antarmuka halaman pengujian

Halaman pengujian memuat proses dan hasil dari pengujian yang telah dilakukan. Tampilan dari hasil pengujian berupa data tabel. Pada halaman ini akan ditampilkan tabel bobot akhir, tabel data uji, tabel perhitungan klasifikasi, hasil klasifikasi kualitas Air Sungai dan akurasi sistem secara umum. Perancangan halaman pengujian dapat dilihat pada Gambar 4.12.

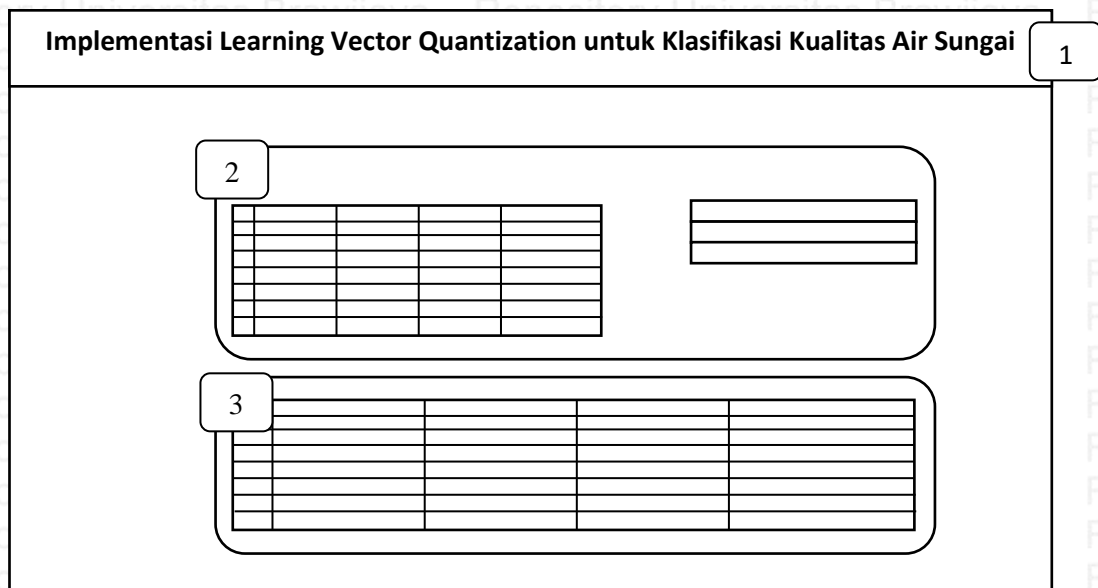




**Gambar 4.11 Antarmuka halaman pelatihan**

Keterangan Gambar 4.11.

1. *Tabbed pane* untuk menampilkan nama sistem dan menu bar.
2. Panel untuk memilih dan memasukkan parameter awal, yaitu *text box* nilai *alfa*, *DecAlfa*, *MinAlfa* dan *MaxEpoch*.
3. Panel yang menampilkan data *training* dan data bobot awal.
4. Panel yang menampilkan tabel perhitungan jarak *euclidian*, tabel perhitungan perubahan bobot dan tabel data *centroid* atau bobot akhir.



**Gambar 4.12 Antarmuka Halaman Pengujian**



Keterangan Gambar 4.12.

5. *Tabbed pane* untuk menampilkan nama sistem dan menu bar.
6. Panel yang menampilkan tabel data *testing* dan data bobot akhir.
7. Panel yang menampilkan tabel perhitungan jarak *euclidian*, tabel perhitungan perubahan bobot, tabel hasil klasifikasi dan hasil akurasi sistem secara umum.

#### 4.6.8 Antarmuka halaman manualisasi

Pada halaman manualisasi akan ditampilkan informasi mengenai proses manualisasi yang dilakukan di excel. Halaman manualisasi berfungsi untuk mengetahui kesesuaian hasil antara perhitungan sistem dengan perhitungan manualisasi yang telah dilakukan di excel. Sehingga, melalui halaman manualisasi bisa diketahui apakah perhitungan sistem sudah berjalan dengan sesuai dan memiliki hasil yang akurat. Informasi yang ditampilkan adalah semua data yang berhubungan dengan manualisasi, yaitu data masukan berupa data *training*, data *testing*, data bobot awal, nilai *alfa*, *MinAlfa*, *DecAlfa*, *MaxEpoch* serta data keluaran berupa data bobot akhir, data hasil klasifikasi dan akurasi. Dimana setiap data inputan yang digunakan adalah data yang digunakan pada proses manualisasi di *excel*. Perancangan halaman manualisasi dapat dilihat pada Gambar 4.13.

The screenshot shows a web-based manualization interface. At the top, there is a title bar (1) with the text 'Implementasi Learning Vector Quantization untuk Klasifikasi Kualitas Air Sungai'. Below the title bar is a parameter input section (2) containing four text boxes for 'alfa', 'DecAlfa', 'MinAlfa', and 'MaxEpoch', followed by a 'Hitung' button. The main area contains two data tables (3 and 4). Table 3 is a 5x5 grid for input data, and Table 4 is a 5x5 grid for output data. The interface is designed to allow users to input parameters and data for the Learning Vector Quantization process.

**Gambar 4.13 Antarmuka Halaman Manualisasi**

Keterangan Gambar 4.13.

1. *Tabbed pane* untuk menampilkan nama sistem dan menu bar.
2. Panel yang menampilkan parameter awal, yaitu *text box* nilai *alfa*, *DecAlfa*, *MinAlfa* dan *MaxEpoch*.
3. Panel yang menampilkan data *training*, data *testing* dan data bobot awal.

4. Panel yang menampilkan tabel data *centroid* atau bobot akhir, tabel hasil klasifikasi dan akurasi sistem secara umum.

#### 4.7 Skenario pengujian

Perancangan skenario pengujian bertujuan untuk dapat mengetahui akurasi dari sistem klasifikasi kualitas air sungai menggunakan metode *Learning Vector Quantization*. Sehingga dapat diketahui seberapa besar keberhasilan dan keakuratan sistem dalam melakukan klasifikasi terhadap kualitas Air Sungai. Selain menghitung akurasi sistem secara umum dengan cara membandingkan antara kelas hasil klasifikasi dengan kelas aktual skenario pengujian algoritma LVQ yang lebih spesifik, yaitu pengujian pengaruh *learning rate* (*Alfa*) awal, pengurangan *learning rate*, iterasi maksimum, laju pembelajaran minimum dan perbandingan jumlah data latih dengan data uji. Percobaan yang dilakukan pada penelitian ini adalah sebanyak 10 kali percobaan mulai percobaan pertama sampai kesepuluh dengan menghitung nilai rata-rata persentase akurasi dari 10 kali percobaan tersebut.

##### 4.7.1 Pengujian *learning rate* (*alfa*) awal

Pengujian *learning rate* awal atau laju pembelajaran awal (*alfa*) digunakan untuk mengetahui pengaruh nilai *learning rate* terhadap tingkat akurasi yang akan dihasilkan. *Learning rate* awal yang digunakan adalah 0,1, 0,2, 0,3, 0,4, 0,5, 0,6, 0,7, 0,8, 0,9, 1. Skenario pengujian *learning rate* awal ditunjukkan pada Tabel 4.6.

**Tabel 4. 6 Pengujian *Learning Rate Awal***

<i>learning rate</i> Awal	Hasil Akurasi (%)					Rata-rata akurasi sistem (%)
	Percobaan ke- i					
	1	2	3	.....	10	
0,1						
0,2						
0,3						
0,4						
0,5						
0,6						
0,7						
0,8						
0,9						
1						



#### 4.7.2 Pengujian pengurangan learning rate (*decAlfa*)

Pengujian pengurangan *learning rate* bertujuan untuk mengetahui besarnya pengaruh nilai pengurangan *learning rate* setiap iterasi terhadap tingkat akurasi yang akan dihasilkan. Pengurangan *learning rate* yang digunakan adalah 0,1, 0,2, 0,3, 0,4, 0,5, 0,6, 0,7, 0,8, 0,9, 1. Percobaan yang dilakukan pada penelitian ini adalah sebanyak 10 kali percobaan. Skenario pengujian pengurangan *learning rate* ditunjukkan pada Tabel 4.7.

**Tabel 4.7 Pengujian Pengurangan *Learning Rate***

Pengurangan <i>learning rate</i>	Hasil Akurasi (%)					Rata-rata akurasi sistem (%)
	Percobaan ke- i					
	1	2	3	...	10	
0,1						
0,2						
0,3						
0,4						
0,5						
0,6						
0,7						
0,8						
0,9						
1						

#### 4.7.3 Perbandingan jumlah data Latih dan data Uji

Pengujian perbandingan jumlah data latih dan data uji digunakan untuk mengetahui pengaruh persentase perbandingan jumlah data yang digunakan terhadap tingkat akurasi yang akan dihasilkan. Dari 135 *dataset* yang ada, maka akan ditentukan jumlah data yang akan dijadikan sebagai data pelatihan dan data yang digunakan sebagai pengujian. Penentuan jumlah data latih dan data uji tersebut dijabarkan dalam bentuk perbandingan, sehingga akan didapatkan komposisi perbandingan data terbaik dengan hasil akurasi yang tinggi. Perbandingan jumlah data latih dan data uji yang digunakan adalah 130:5, 125:10, 110:25, 100:35, 85:50, 68:67, 50:85, 35:100, 25:110, 10:125 atau dalam persentase 7,4%, 18,5%, 25,9%, 37%, 50,3%, 62,9%, 74,1%, 81,4%, 92,5% jumlah data latih dari seluruh *dataset*. Percobaan yang dilakukan pada penelitian ini adalah sebanyak 10 kali percobaan. Skenario perbandingan data latih dan data uji ditunjukkan pada Tabel 4.8.



Tabel 4.8 Pengujian Perbandingan Data Latih dan Data Uji

Jumlah Data Latih	Hasil Akurasi (%)					Rata-rata akurasi sistem (%)
	Percobaan ke- i					
	1	2	3	...	10	
10						
25						
35						
50						
68						
85						
100						
110						
125						
130						

#### 4.7.4 Pengujian maksimal perulangan (*epoch*)

Pengujian maksimal perulangan atau iterasi maksimal (*epoch*) bertujuan untuk mengetahui pengaruh nilai iterasi maksimum terhadap tingkat akurasi yang akan dihasilkan. Iterasi maksimum yang digunakan adalah 1, 2, 4, 6, 8, 10, 14, 16, 18, 20, 40 dan 60. Percobaan yang dilakukan pada penelitian ini adalah sebanyak 10 kali percobaan. Skenario pengujian *learning rate* awal ditunjukkan pada Tabel 4.9.

Tabel 4.9 Pengujian Iterasi Maksimum

Iterasi Maks	Hasil Akurasi (%)					Rata-rata akurasi sistem (%)
	Percobaan ke- i					
	1	2	3	....	10	
1						
2						
4						
6						
8						
10						



14					
16					
18					
20					
40					
60					

**4.7.5 Pengujian minimal alfa**

Pengujian pengaruh laju pembelajaran atau *learning rate* (*minAlfa*) digunakan untuk mengetahui besarnya pengaruh nilai minimal *learning rate* terhadap tingkat akurasi yang akan dihasilkan. Nilai minimal *alfa* yang digunakan adalah  $10^{-1}$ , 0,05,  $10^{-2}$ , 0,005,  $10^{-3}$ ,  $10^{-4}$ ,  $10^{-5}$ ,  $10^{-6}$ ,  $10^{-7}$ ,  $10^{-8}$ ,  $10^{-9}$ ,  $10^{-10}$ . Percobaan yang dilakukan pada penelitian ini adalah sebanyak 10 kali percobaan. Skenario pengujian pengaruh *learning rate* awal ditunjukkan pada Tabel 4.10.

**Tabel 4.10 Pengujian Pengaruh Laju Pembelajaran Minimum**

<i>learning rate</i> Min	Hasil Akurasi (%)					Rata-rata akurasi sistem (%)
	Percobaan ke- i					
	1	2	3	...	5	
$10^{-1}$						
0,05						
$10^{-2}$						
0,005						
$10^{-3}$						
$10^{-4}$						
$10^{-5}$						
$10^{-6}$						
$10^{-7}$						
$10^{-8}$						
$10^{-9}$						
$10^{-10}$						



#### 4.7.6 Pengujian akurasi dan validasi parameter optimal

Pengujian akurasi dan validasi parameter optimal digunakan untuk mengetahui besarnya rata-rata akurasi yang didapatkan berdasarkan hasil pengujian kelima parameter yang telah didapatkan sebelumnya, yaitu dengan menggabungkan nilai *learning rate*, *decAlfa*, perbandingan jumlah data latih dan data uji, maksimum *epoch* dan minimum *alfa* yang optimal berdasarkan hasil scenario pengujian sebelumnya. Percobaan yang dilakukan pada penelitian ini adalah sebanyak 10 kali percobaan. Skenario pengujian pengaruh *learning rate* awal ditunjukkan pada Tabel 4.11.

**Tabel 4.11 Pengujian Akurasi dan Validasi Parameter Optimal**

Hasil Akurasi (%)					Rata-rata akurasi sistem (%)
Percobaan ke- i					
1	2	3	...	10	



## BAB 5 IMPLEMENTASI

Bab implementasi menjelaskan mengenai lingkup implementasi penerapan *Learning Vector Quantization* untuk klasifikasi kualitas air sungai sesuai dengan perancangan yang sudah dijelaskan pada bab sebelumnya. Lingkup penerapan yang dimaksud adalah penjelasan dari perangkat keras dan perangkat lunak yang digunakan, batasan-batasan dalam pengimplementasian sistem, implementasi metode yang digunakan pada sistem yaitu membangun metode LVQ berupa kode program dan implementasi dari antarmuka yang sudah dibangun.

### 5.1 Lingkup perangkat keras

Informasi mengenai perangkat keras (*hardware*) yang digunakan dalam proses implementasi adalah sebagai berikut:

1. Laptop Intel(R) Core(TM) i7-3537U CPU @ 2.00GHz (4 CPUs).
2. Memori (RAM) 4096MB.

### 5.2 Lingkup perangkat lunak

Perangkat lunak (*software*) yang diperlukan dalam membangun implementasi LVQ untuk klasifikasi kualitas air sungai adalah sebagai berikut:

1. Sistem Operasi (OS) Windows 10 Pro 64-bit.
2. Microsoft office dan excel professional 2016 sebagai aplikasi penyusunan dan perhitungan pada laporan.
3. ApacheFriends XAMPP Version 5.6.30.
4. Sublimetext version 3 Build 3126 64 bit dan notepad++ sebagai aplikasi yang mengimplementasikan algoritma metode dan bahasa pemrograman.
5. MySQL sebagai penyimpanan dan pengolahan database untuk dataset kualitas air sungai.
6. Google Chrome Version 58.0.3029.110 (64-bit) dan Microsoft Edge 38.14393.1066.0 sebagai *web browser* dan *running* dari implementasi sistem.

### 5.3 Batasan-batasan implementasi

Beberapa Batasan yang digunakan dalam mengimplementasikan perangkat lunak untuk klasifikasi kualitas air sungai adalah sebagai berikut:

1. Bahasa pemrograman yang diimplementasikan adalah php menggunakan *framework* codeigniter versi 3.1.4.
2. Data disimpan dan diolah dalam *database* menggunakan mySQL dalam format .sql.





3. Perangkat lunak system implementasi LVQ untuk klasifikasi kualitas air sungai dirancang, dibangun dan dijalankan menggunakan *Web Based Application*.
4. File program yang digunakan memiliki format .php, .html, .css dan javascript.
5. Masukan yang diterima sistem berupa nilai-nilai parameter alfa, pengurangan alfa, minimal alfa dan maksimal *epoch*. Selain itu, masukan yang diterima berupakan dataset, jumlah kelas, pembagian data latih dan data ujian jumlah perbandingan antara banyaknya data latih dengan data uji.
6. *Output* yang dihasilkan sistem berupa hasil bobot optimal dari pelatihan data dan jumlah akurasi yang didapatkan berdasarkan hasil pengujian.

## 5.4 Implementasi kode program

Implementasi kode program menjelaskan fungsi kode program yang digunakan dalam membangun sistem klasifikasi kualitas air sungai dan menerjemahkan metode LVQ kedalam bahasa pemrograman dan *framework* yang digunakan. Sistem diimplementasikan kedalam *framework* codeigniter berbasis MVC (*Model, View* dan *Controller*) yang terdapat beberapa modul, fungsi dan kelas-kelas yang saling berinteraksi satu sama lain. Berikut pokok pembahasan pada Implementasi *Learning Vector Quantization* untuk klasifikasi kualitas air sungai kedalam kode program yang dijelaskan sebagai berikut:

### 5.4.1 Implementasi proses input data

Tahapan awal sebelum proses pelatihan dan pengujian data dilakukan adalah melakukan inialisasi data bobot awal, data latih dan data uji terhadap dataset yang diinputkan. Proses inialisasi diambil dari dataset format excel yang sudah diinputkan dan dikonversi serta disimpan dalam *database* MySQL. Inialisasi data bobot awal, data latih dan data uji dilakukan secara acak berdasarkan data-data pada dataset. Jumlah data bobot awal adalah 3 data mewakili 3 kelas output pada dataset, yaitu kelas ringan, sedang dan berat, sedangkan jumlah data latih dan data uji ditentukan oleh nilai perbandingan yang diinputkan oleh *user* (pengguna). Kode program untuk proses input data dapat dilihat pada *Source Code* 5.1.

```

1  $perbandingan = $this->input->post('perbandingan');
2  $data_sungai = $this->usermodel->selectdata('data_sungai');
3  $del_data_pelatihan = $this->usermodel->delalldata('data_pelatihan');
4  $data_pelatihan = $this->usermodel->copydatapelatihan('data_sungai ORDER
BY Rand() LIMIT '.$perbandingan);
5  $data_pelatihan = $this->usermodel->selectdata('data_pelatihan');
6  $data_jumlah = array('jumlah'=> $pengujian);
7  $del_data_pengujian=$this->usermodel->delalldata('jumlah_data_uji');
8  $insert_jumlah_data_uji = $this->usermodel->insertdata('jumlah_data_uji',
$data_jumlah);

```



```

9   $del_data_pengujian = $this->usermodel->delalldata('data_pengujian');
10  $data_pengujian = $this->usermodel->join();
11  $data_pengujian = $this->usermodel->selectdata('data_pengujian');
12  $del_data_centroid = $this->usermodel->delalldata('data_centroid');
13  $data_centroid = $this->usermodel->copydatacentroid('data_sungai WHERE
14  Ket="Ringan" ORDER BY Rand() LIMIT 1');
15  $data_centroid = $this->usermodel->copydatacentroid('data_sungai WHERE
16  Ket="Sedang" ORDER BY Rand() LIMIT 1');
17  $data_centroid = $this->usermodel->copydatacentroid('data_sungai WHERE
18  Ket="Berat" ORDER BY Rand() LIMIT 1');
19  $data_centroid = $this->usermodel->selectdata('data_centroid');

```

#### Source Code 5.1 Proses Input data

Penjelasan *Source Code* 5.1:

- Baris 1 berisi variabel untuk mendapatkan nilai perbandingan data latihan dan data uji yang diinputkan pengguna menggunakan *method post*.
- Baris 2-6 merupakan proses pengambilan dataset kualitas air sungai dan diinputkan menjadi data latihan yang diambil secara acak berdasarkan fungsi *order by rand()* dengan batasan jumlah sesuai masukan pengguna.
- Baris 7-13 menjelaskan proses inialisasi data uji yang disimpan berdasarkan hasil model *left outer join* antara dataset dengan data latihan.
- Baris 14-21 merupakan proses inialisasi data bobot awal (data *centroid*) yang mewakili setiap kelas hasil, yaitu diambil dari dataset secara acak dengan fungsi model *where* ringan, sedang atau berat.

#### 5.4.2 Implementasi proses pelatihan *Learning Vector Quantization* (LVQ)

Tahapan awal proses pelatihan dilakukan dengan menginisialisasi parameter perhitungan LVQ yaitu nilai alfa, *decAlfa*, *minAlfa* dan *maxEpoch*. Selanjutnya dilakukan perhitungan selisih jarak antara setiap data latihan dengan data bobot yang ada. Jarak terdekat (terkecil) dari ketiga kelas hasil tersebut kemudian menjadi pemilihan untuk pembobotan data bobot baru yang akan digunakan untuk data berikutnya. Hal ini dilakukan berulang sampai mencapai iterasi maksimal dan alfa minimal, sehingga menghasilkan keluaran berupa nilai bobot optimal. Kode program untuk proses pelatihan dapat dilihat pada *Source Code* 5.2.

```

1   $data_pelatihan = $this->usermodel->selectdata('data_pelatihan');
2   $data_centroid = $this->usermodel->selectdata('data_centroid order byskor
3   desc');
4   $epoh = $this->input->post('epoh');
5   $alfa = $this->input->post('alfa');
6   $decAlfa = $this->input->post('decAlfa');
7   $minAlfa = $this->input->post('minAlfa');

```



```

8     for ($epohawal=0; ($alfa >= $minAlfa) && ($epohawal < $epoh);
9         $epohawal++) {
10        foreach($data_pelatihan->result_array() as $pm){
11            $data_centroid_ringan=$this->usermodel->selectdata
12            ('data_centroid_akhir where ket = "Ringan"');
13            $data_centroid_sedang = $this->usermodel->selectdata
14            ('data_centroid_akhir where ket = "Sedang"');
15            $data_centroid_berat = $this->usermodel->selectdata
16            ('data_centroid_akhir where ket = "Berat"');
17            foreach($data_centroid_ringan->result_array() as $crm){
18                $sum1m = pow($pm['tss']- $crm['tss'],2)+
19                pow($pm['bod']- $crm['bod'],2)+ pow($pm['cod']-
20                $crm['cod'],2)+
21                pow($pm['do']- $crm['do'],2)+ pow($pm['pH']- $crm['pH'],2)+
22                pow($pm['fenol']- $crm['fenol'],2)+
23                pow($pm['minyakLemak']- $crm['minyakLemak'],2)+
24                pow($pm['skor']- $crm['skor'],2);
25                $jarak1mm = sqrt($sum1m);
26                $jarak1m = ($jarak1mm,5);
27            }
28            foreach($data_centroid_sedang->result_array() as $csm){
29                $sum2m = pow($pm['tss']- $csm['tss'],2)+ pow($pm['bod']-
30                $csm['bod'],2)+
31                pow($pm['cod']- $csm['cod'],2)+ pow($pm['do']-
32                $csm['do'],2)+
33                pow($pm['pH']- $csm['pH'],2)+ pow($pm['fenol']-
34                $csm['fenol'],2)+
35                pow($pm['minyakLemak']- $csm['minyakLemak'],2)+
36                pow($pm['skor']- $csm['skor'],2);
37                $jarak2mm = sqrt($sum2m);
38                $jarak2m = ($jarak2mm,5);
39            }
40            foreach($data_centroid_berat->result_array() as $cbm){
41                $sum3m = pow($pm['tss']- $cbm['tss'],2)+ pow($pm['bod']-
42                $cbm['bod'],2)+
43                pow($pm['cod']- $cbm['cod'],2)+ pow($pm['do']-
44                $cbm['do'],2)+
45                pow($pm['pH']- $cbm['pH'],2)+ pow($pm['fenol']-
46                $cbm['fenol'],2)+
47                pow($pm['minyakLemak']- $cbm['minyakLemak'],2)+

```



```

48     pow($pm['skor']- $cbm['skor'],2);
49     $jarak3mm = sqrt($sum3m);
50     $jarak3m = ($jarak3mm,5);
51     }
52     $minimalm = min($jarak1m,$jarak2m,$jarak3m);
53     if ($minimalm == $jarak1m) {
54     $minimal2m = "Ringan";
55     }else if ( $minimalm == $jarak2m) {
56     $minimal2m = "Sedang";
57     }else{
58     $minimal2m = "Berat";
59     }
60     if ($minimal2m == $pm['ket']) {
61     $keterangan = "SAMA";
62     if ($minimal2m == 'Ringan') {
63     foreach($data_centroid_ringan->result_array() as $crb){
64     $tss = $crb['tss'] + ($alfa* ($pm['tss']- $crb['tss'])) );
65     $bod = $crb['bod'] + ($alfa* ($pm['bod']- $crb['bod'])) );
66     $cod = $crb['cod'] + ($alfa* ($pm['cod']- $crb['cod'])) );
67     $do = $crb['do'] + ($alfa* ($pm['do']- $crb['do'])) );
68     $pH = $crb['pH'] + ($alfa* ($pm['pH']- $crb['pH'])) );
69     $fenol = $crb['fenol'] + ($alfa* ($pm['fenol']-
70     $crb['fenol'])) );
71     $minyakLemak = $crb['minyakLemak'] + ($alfa*
72     ($pm['minyakLemak']- $crb['minyakLemak'])) );
73     $skor = $crb['skor'] + ($alfa* ($pm['skor']-
74     $crb['skor']));
75     $this->usermodel->updatecentroidringan($tss,$bod, $cod,
76     $do,$pH, $fenol,$minyakLemak, $skor );
77     }
78     } else if ($minimal2m == 'Sedang') {
79     foreach($data_centroid_sedang->result_array() as $csbm){
80     $tss2m = $csbm['tss'] + ($alfa* ($pm['tss']- $csbm['tss'])
81     );
82     $bod2m = $csbm['bod'] + ($alfa* ($pm['bod']- $csbm['bod'])
83     );
84     $cod2m = $csbm['cod'] + ($alfa* ($pm['cod']- $csbm['cod'])
85     );
86     $do2m = $csbm['do'] + ($alfa* ($pm['do']- $csbm['do'])) );
87     $pH2m = $csbm['pH'] + ($alfa* ($pm['pH']- $csbm['pH']))

```



```

88     );
89     $fenol2m = $csbm['fenol'] + ($alfa* ($pm['fenol']-
90     $csbm['fenol'])) );
91     $minyakLemak2m = $csbm['minyakLemak'] + ($alfa*
92     ($pm['minyakLemak']- $csbm['minyakLemak'])) );
93     $skor2m = $csbm['skor'] + ($alfa* ($pm['skor']-
94     $csbm['skor'])) );
95     $this->usermodel->updatecentroidsedang($tss2m,$bod2m,$cod2m,
96     $do2m,$pH2m, $fenol2m,$minyakLemak2m, $skor2m );
97     }
98     } else{
99     foreach($data_centroid_berat->result_array() as $cbb){
100     $tss3 = $cbb['tss'] + ($alfa* ($pm['tss']-
101     $cbb['tss'])) );
102     $bod3 = $cbb['bod'] + ($alfa* ($pm['bod']- $cbb['bod'])) );
103     $cod3 = $cbb['cod'] + ($alfa* ($pm['cod']- $cbb['cod'])) );
104     $do3 = $cbb['do'] + ($alfa* ($pm['do']- $cbb['do'])) );
105     $pH3 = $cbb['pH'] + ($alfa* ($pm['pH']- $cbb['pH'])) );
106     $fenol3 = $cbb['fenol'] + ($alfa* ($pm['fenol']- $cbb['fenol']
107     ));
108     $minyakLemak3 = $cbb['minyakLemak'] + ($alfa*
109     ($pm['minyakLemak']- $cbb['minyakLemak'])) );
110     $skor3 = ($cbb['skor'] + ($alfa* ($pm['skor']- $cbb['skor']
111     ));
112     $this->usermodel->updatecentroidberat($tss3,$bod3,$cod3,
113     $do3,$pH3, $fenol3,$minyakLemak3, $skor3 );
114     }
115     }
116     } else {
117     $keterangan = "BEDA";
118     if ($minimal2m == 'Ringan') {
119     foreach($data_centroid_ringan->result_array() as $crb2){
120     $tss4 = $crb2['tss'] - ($alfa* ($pm['tss']- $crb2['tss'])) );
121     $bod4 = $crb2['bod'] - ($alfa* ($pm['bod']- $crb2['bod'])) );
122     $cod4 = $crb2['cod'] - ($alfa* ($pm['cod']- $crb2['cod'])) );
123     $do4 = $crb2['do'] - ($alfa* ($pm['do']- $crb2['do'])) );
124     $pH4 = $crb2['pH'] - ($alfa* ($pm['pH']- $crb2['pH'])) );
125     $fenol4 = $crb2['fenol'] - ($alfa* ($pm['fenol']-
126     $crb2['fenol'])) );
127     $minyakLemak4 = $crb2['minyakLemak'] - ($alfa*

```



```

128      ($pm['minyakLemak'] $crb2['minyakLemak']) );
129      $skor4 = $crb2['skor'] - ($alfa* ($pm['skor']- $crb2['skor']))
130      );
131      $this->usermodel->updatecentroidringan($tss4, $bod4, $cod4,
132      $do4, $pH4, $fenol4, $minyakLemak4, $skor4);
133      }
134      } else if ($minimal2m == 'Sedang') {
135      foreach($data_centroid_sedang->result_array() as $csb3m){
136      $tss5m = $csb3m['tss'] - ($alfa* ($pm['tss']- $csb3m['tss']))
137      );
138      $bod5m = $csb3m['bod'] - ($alfa* ($pm['bod']- $csb3m['bod']))
139      );
140      $cod5m = $csb3m['cod'] - ($alfa* ($pm['cod']- $csb3m['cod']))
141      );
142      $do5m = $csb3m['do'] - ($alfa* ($pm['do']- $csb3m['do'])) );
143      $pH5m = $csb3m['pH'] - ($alfa* ($pm['pH']- $csb3m['pH'])) );
144      $fenol5m = $csb3m['fenol'] - ($alfa* ($pm['fenol']-
145      $csb3m['fenol'])) );
146      $minyakLemak5m = $csb3m['minyakLemak'] - ($alfa*
147      ($pm['minyakLemak']- $csb3m['minyakLemak'])) );
148      $skor5m = $csb3m['skor'] - ($alfa* ($pm['skor']-
149      $csb3m['skor'])) );
150      $this->usermodel->updatecentroidsedang($tss5m,$bod5m,$cod5m,
151      $do5m,$pH5m, $fenol5m,$minyakLemak5m, $skor5m);
152      }
153      } else{
154      foreach($data_centroid_berat->result_array() as $cbb2){
155      $tss6 = $cbb2['tss'] - ($alfa* ($pm['tss']- $cbb2['tss'])) );
156      $bod6 = $cbb2['bod'] - ($alfa* ($pm['bod']- $cbb2['bod'])) );
157      $cod6 = $cbb2['cod'] - ($alfa* ($pm['cod']- $cbb2['cod'])) );
158      $do6 = $cbb2['do'] - ($alfa* ($pm['do']- $cbb2['do'])) );
159      $pH6 = $cbb2['pH'] - ($alfa* ($pm['pH']- $cbb2['pH'])) );
160      $fenol6 = $cbb2['fenol'] - ($alfa* ($pm['fenol']-
161      $cbb2['fenol'])) );
162      $minyakLemak6 = $cbb2['minyakLemak'] - ($alfa*
163      ($pm['minyakLemak']- $cbb2['minyakLemak'])) );
164      $skor6 = $cbb2['skor'] - ($alfa* ($pm['skor']- $cbb2['skor']))
165      );
166      $this->usermodel->updatecentroidberat($tss6,$bod6,
167      $cod6, $do6, $pH6, $fenol6,$minyakLemak6, $skor6);

```



168	}
169	}
170	\$alfa = \$alfa-(\$alfa*\$decAlfa);
171	}

### Source Code 5.2 Proses Pelatihan *Learning Vector Quantization* (LVQ)

Penjelasan *Source Code* 5.2:

- Baris 1-7 berisi variabel untuk mendapatkan data latih, data bobot awal dan nilai parameter awal sesuai yang diinputkan oleh pengguna menggunakan *method post*.
- Baris 9-10 merupakan mulainya proses perulangan data pelatihan mulai dari *epoch* pertama sampai *epoch* maksimal dengan kondisi *true* yaitu nilai alpha besar sama dengan nilai minAlpha dan *epoch* lebih kecil dari nilai *maxepoch*.
- Baris 12-18 menjelaskan proses inialisasi data bobot awal yang mewakili kelas *output*.
- Baris 20-31 merupakan proses perhitungan selisih jarak antara setiap data latih dengan data bobot yang mewakili kelas ringan.
- Baris 32-43 merupakan proses perhitungan selisih jarak antara setiap data latih dengan data bobot yang mewakili kelas sedang.
- Baris 45-56 merupakan proses perhitungan selisih jarak antara setiap data latih dengan data bobot yang mewakili kelas berat.
- Baris 68-65 menentukan kelas dengan jarak terdekat dari perhitungan *euclidian distance*.
- Baris 66-119 merupakan proses kondisi untuk melakukan pembobotan penjumlahan data kelas bobot yang baru dengan kondisi kelas target sama dengan kelas terdekat.
- Baris 120-173 merupakan proses kondisi untuk melakukan pembobotan pengurangan data kelas bobot yang baru dengan kondisi kelas target tidak sama dengan kelas terdekat.
- Baris 173-175 menentukan perubahan nilai *alpha* untuk perhitungan iterasi selanjutnya dan menandakan kondisi perulangan telah berhenti.

### 5.4.3 Implementasi proses pengujian *Learning Vector Quantization* (LVQ)

Tahapan setelah proses pelatihan dilakukan adalah melakukan pengujian data uji dan akurasi. Inputan pada proses pengujian adalah data uji yang telah diinisialisasi diawal proses dan data bobot optimal hasil dari pelatihan data. Pengujian data dilakukan dengan melakukan perhitungan selisih jarak antara setiap data uji dengan data bobot optimal yang ada. Jarak terdekat (terkecil) dari ketiga kelas hasil tersebut kemudian menjadi penentuan untuk perhitungan akurasi perhitungan apakah sama atau berbeda dengan kelas target yang



diharapkan. Hal ini dilakukan berulang sampai untuk setiap data uji sehingga menghasilkan perbandingan dan akurasi system secara umum. Kode program untuk proses pengujian pada *Source Code* 5.3.

```

1  foreach($jumlah->result_array() as $j){
2      $count = 0;
3      foreach($data_pengujian->result_array() as $p){
4          foreach($data_centroid_ringan->result_array() as $cr){
5              $sum1 = pow($p['tss']- $cr['tss'],2)+
6                  pow($p['bod']- $cr['bod'],2)+
7                  pow($p['cod']- $cr['cod'],2)+
8                  pow($p['do']- $cr['do'],2)+
9                  pow($p['pH']- $cr['pH'],2)+
10                 pow($p['fenol']- $cr['fenol'],2)+
11                 pow($p['minyakLemak']- $cr['minyakLemak'],2)+
12                 pow($p['skor']- $cr['skor'],2);
13             $jarak1 = sqrt($sum1);
14         }
15         foreach($data_centroid_sedang->result_array() as $cs){
16             $sum2 = pow($p['tss']- $cs['tss'],2)+
17                 pow($p['bod']- $cs['bod'],2)+
18                 pow($p['cod']- $cs['cod'],2)+
19                 pow($p['do']- $cs['do'],2)+
20                 pow($p['pH']- $cs['pH'],2)+
21                 pow($p['fenol']- $cs['fenol'],2)+
22                 pow($p['minyakLemak']- $cs['minyakLemak'],2)+
23                 pow($p['skor']- $cs['skor'],2);
24             $jarak2 = sqrt($sum2);
25         }
26         foreach($data_centroid_berat->result_array() as $cb){
27             $sum3 = pow($p['tss']- $cb['tss'],2)+
28                 pow($p['bod']- $cb['bod'],2)+
29                 pow($p['cod']- $cb['cod'],2)+
30                 pow($p['do']- $cb['do'],2)+
31                 pow($p['pH']- $cb['pH'],2)+
32                 pow($p['fenol']- $cb['fenol'],2)+
33                 pow($p['minyakLemak']- $cb['minyakLemak'],2)+
34                 pow($p['skor']- $cb['skor'],2);
35             $jarak3 = sqrt($sum3);
36         }
37         $minimal = min($jarak1,$jarak2,$jarak3);

```





```

38     if ($minimal == $jarak1) {
39         $minimal2 = "Ringan";
40     }else if ( $minimal == $jarak2) {
41         $minimal2 = "Sedang";
42     }else{
43         $minimal2 = "Berat";
44     }
45     if ($minimal2 == $p['ket']) {
46         $keterangan = 'SAMA' ;
47         $count += 1;
48     }else{
49         $keterangan = "TIDAK SAMA";
50     }
51 }
52 $akurasi = ($count/$j['jumlah'])*100 ;

```

### Source Code 5. 3 Implementasi Proses Pngujian Learning Vector Quantization (LVQ)

Penjelasan *Source Code* 5.3:

- Baris 1 merupakan kondisi perulangan untuk melakukan perhitungan sebanyak jumlah data uji.
- Baris 2 berisi variabel untuk *count* untuk menentukan jumlah data yang sama, dimana nantinya variabel ini digunakan untuk menghitung akurasi hasil perhitungan data.
- Baris 4 menjelaskan proses perulangan untuk memanggil setiap data uji.
- Baris 6-16 merupakan proses perhitungan selisih jarak antara setiap data uji dengan data bobot yang mewakili kelas ringan.
- Baris 18-28 merupakan proses perhitungan selisih jarak antara setiap data uji dengan data bobot yang mewakili kelas sedang.
- Baris 30-40 merupakan proses perhitungan selisih jarak antara setiap data uji dengan data bobot yang mewakili kelas berat.
- Baris 42-49 merupakan kondisi untuk menentukan kelas terdekat yang memiliki jarak terkecil.
- Baris 51-57 merupakan kondisi untuk menghitung jumlah data yang memiliki kelas target yang sama dengan kelas hasil perhitungan data uji, dan menjadi *stop* kondisi untuk perulangan data uji.
- Baris 58 merupakan perhitungan persentase akurasi hasil perhitungan data uji.



## 5.5 Implementasi antarmuka

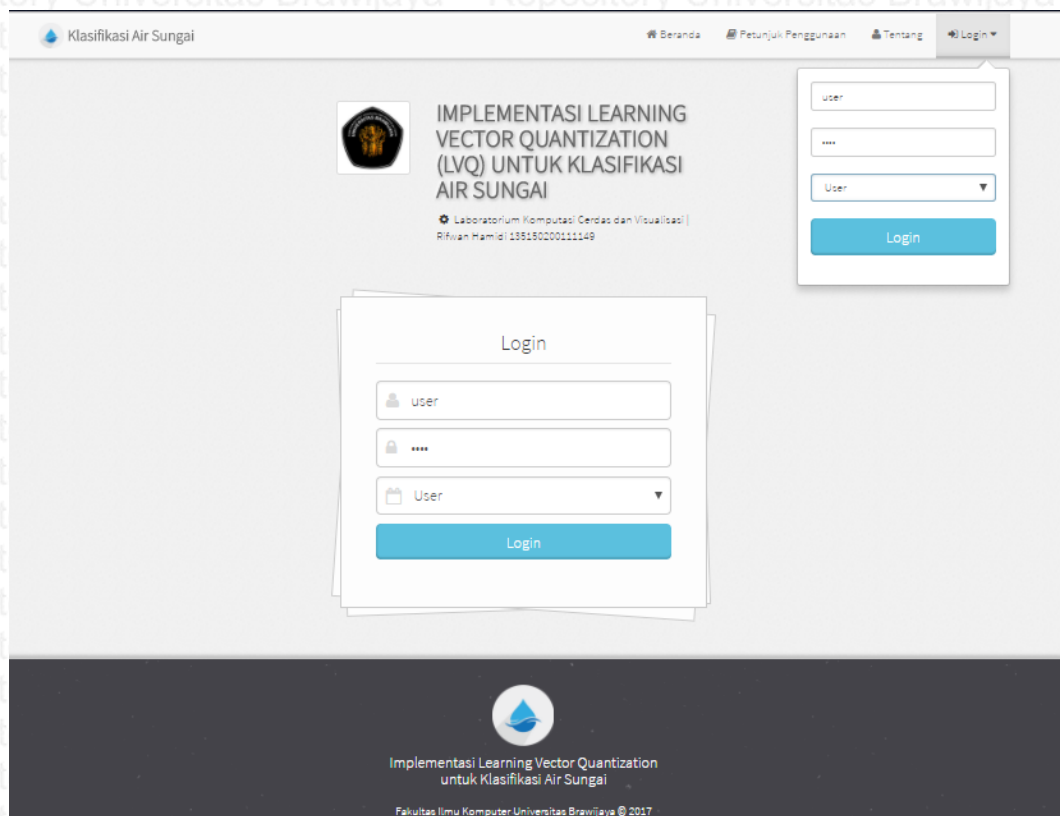
Sub bab ini menjelaskan dan menggambarkan mengenai implementasi halaman antarmuka klasifikasi kualitas air sungai berdasarkan perancangan yang telah dibuat pada bab sebelumnya. Implementasi antarmuka berbasis pada *desktop dan web based application*.

### 5.5.1 Antarmuka halaman beranda

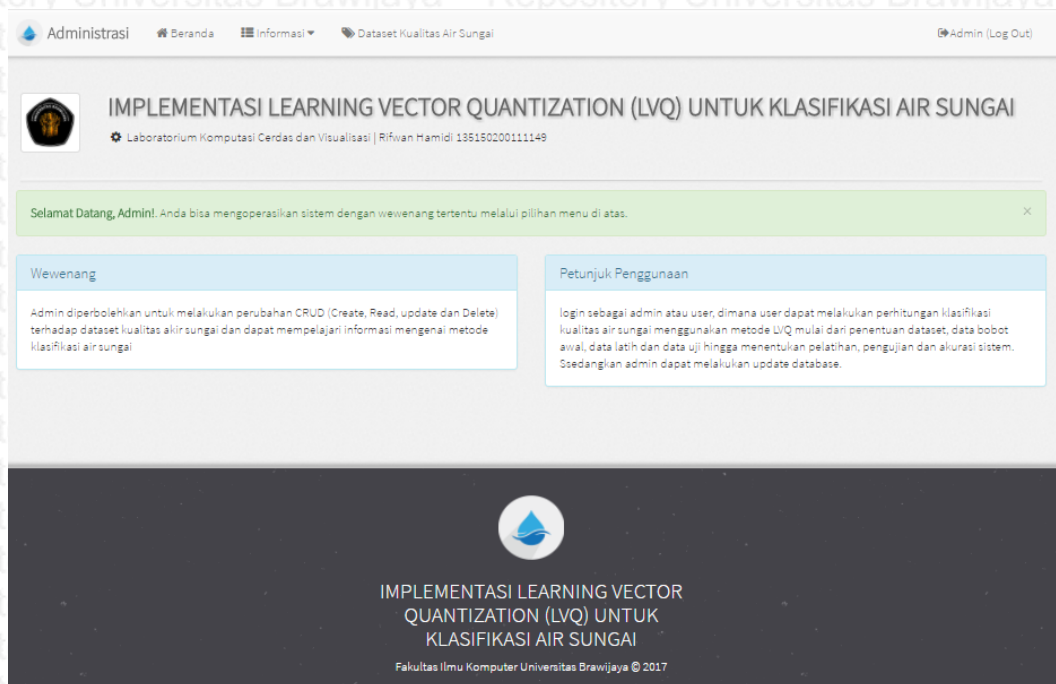
Halaman beranda (*home*) merupakan halaman antarmuka yang pertama kali muncul setelah sistem diaktifkan. Halaman *home* ini berisi fungsi untuk login kedalam sistem sebagai hak akses pengguna (*user*) atau sebagai hak akses *admin* dan tampilan petunjuk penggunaan dan informasi mengenai sistem klasifikasi kualitas air sungai dengan LVQ. Berikut tampilan halaman beranda seperti Gambar 5.1.

### 5.5.2 Antarmuka halaman beranda admin

Halaman beranda (*home*) pada admin merupakan halaman antarmuka yang pertama kali muncul setelah masuk sebagai hak akses admin diaktifkan. Halaman *home* ini berisi informasi hak akses admin dalam perubahan database dan dataset sistem klasifikasi kualitas air sungai dengan LVQ. Berikut tampilan halaman beranda admin seperti Gambar 5.2.



**Gambar 5.1 Antarmuka Halaman Beranda (Home)**



**Gambar 5.2 Antarmuka Halaman Beranda Admin**

### 5.5.3 Antarmuka halaman CRUD data admin

Halaman dataset kualitas air sungai ini berisi fungsi untuk admin dalam melakukan proses CRUD (*Create, Read, Update* dan *Delete*) terhadap dataset yang digunakan. Berikut tampilan halaman dataset kualitas air sungai pada Gambar 5.3.

### 5.5.4 Antarmuka halaman beranda *user* (pengguna)

Halaman beranda (*home*) pada pengguna (*user*) merupakan halaman antarmuka yang pertama kali muncul setelah masuk sebagai hak akses pengguna. Halaman *home* ini berisi informasi hak akses pengguna dalam melakukan perhitungan klasifikasi metode LVQ.

### 5.5.5 Antarmuka halaman *generate* dan analisis data

Halaman *generate* data merupakan halaman untuk menentukan inisialisasi data bobot awal, data latih dan data uji berdasarkan masukan pengguna berupa perbandingan jumlah data latih dan data uji. Antarmuka Halaman Generate Data bobot awal, Data Pelatihan dan Data Pengujian dapat dilihat pada Gambar 5.5.

### 5.5.6 Antarmuka halaman pelatihan

Halaman pelatihan merupakan halaman yang digunakan untuk melatih data latih dengan data bobot sehingga dihasilkan data bobot optimal. Masukan pada halaman ini adalah parameter awal perhitungan *Learning Vector Quantization* (LVQ) yang diinputkan oleh pengguna. Hasil dari pelatihan adalah bobot optimal yang akan digunakan untuk proses selanjutnya yaitu proses pengujian data. Berikut antarmuka halaman pelatihan sesuai Gambar 5.6.



### 5.5.7 Antarmuka halaman pengujian

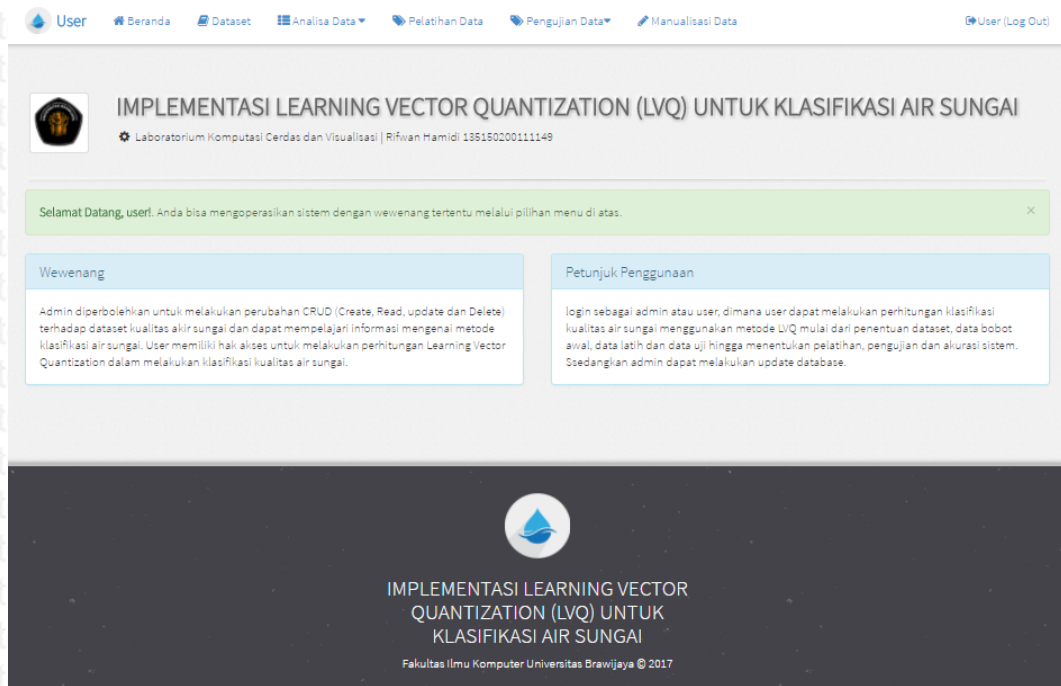
Halaman pengujian merupakan halaman yang digunakan untuk melakukan pengujian terhadap data uji dengan data bobot optimal hasil pelatihan. Hasil dari pelatihan adalah akurasi dari perhitungan *Learning Vector Quantization* (LVQ). Berikut antarmuka halaman pengujian sesuai Gambar 5.7.

### 5.5.8 Antarmuka halaman manualisasi

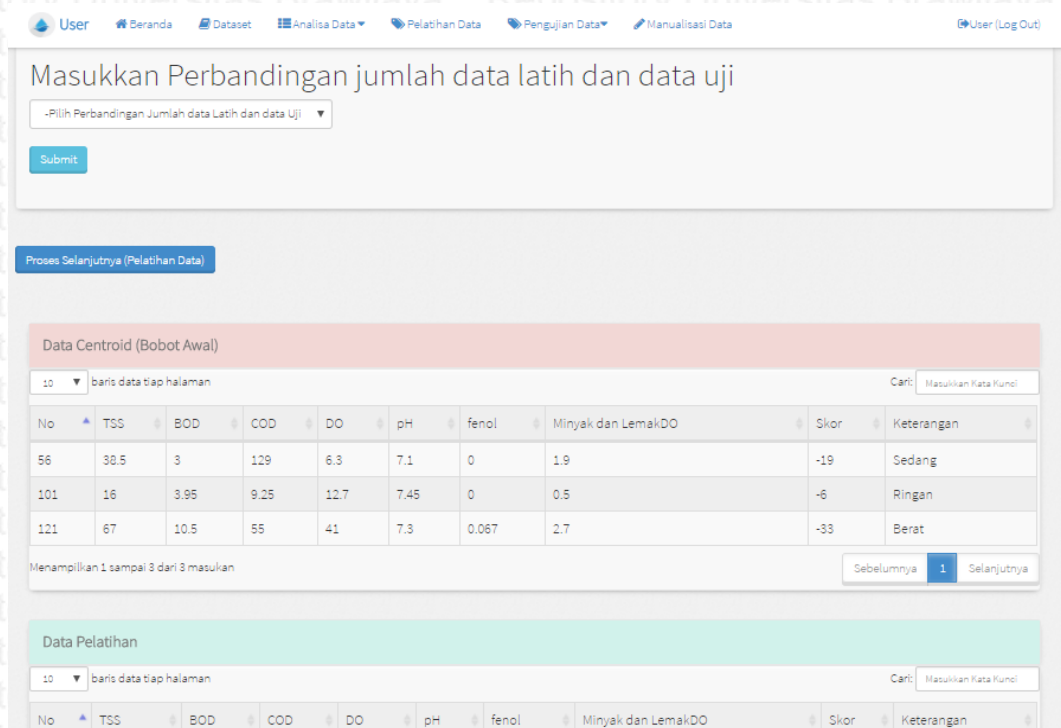
Halaman manualisasi merupakan halaman tambahan yang digunakan untuk melakukan validasi terhadap perhitungan yang telah diimplementasikan pada kode program. Validasi yang dimaksud bertujuan untuk memastikan kebenaran perhitungan *Learning Vector Quantization* (LVQ) yang diimplementasikan dengan cara membandingkan kesamaan hasil dengan perhitungan manual pada bab sebelumnya. Berikut antarmuka halaman manualisasi sesuai Gambar 5.8.

No	TSS	BOD	COD	DO	pH	fenol	Minyak dan Lemak	Skor	Keterangan	Aksi
1	0	15.35	56.0	11.2	7.5	0.0775	0.0	-30	Sedang	<a href="#">Edit</a> <a href="#">Hapus</a>
2	0	10	34.0	15.3	8.05	0.0145	4.5	-42	Berat	<a href="#">Edit</a> <a href="#">Hapus</a>
3	44.85	12.6	35.4	18.55	8.85	0.1185	4.25	-39	Berat	<a href="#">Edit</a> <a href="#">Hapus</a>
4	72.75	19.75	62.85	10.45	8.1	0.172	0	-32	Berat	<a href="#">Edit</a> <a href="#">Hapus</a>
5	10	6.2	13.9	10.15	7.55	0.172	1.5	-30	Sedang	<a href="#">Edit</a>

**Gambar 5.3** Antarmuka Halaman CRUD Dataset Kualitas Air Sungai



**Gambar 5.4** Antarmuka Halaman Beranda Pengguna



**Gambar 5.5** Antarmuka Halaman Generate Data bobot awal, Data Pelatihan dan Data Pengujian

## Pelatihan Data

Parameter Awal ( maxEpoch = 3, Alfa = 0.25, DecAlfa = 0.05, dan minAlfa = 0.1 )

Data Centroid (Bobot) Awal

TSS	BOD	COD	DO	pH	fenol	Minyak dan Lemak	DO	Skor	Keterangan
18	3.95	9.25	12.7	7.45	0	0.5		-6	Ringan
38.5	3	129	6.3	7.1	0	1.9		-19	Sedang
67	10.5	55	41	7.3	0.067	2.7		-33	Berat

EPOCH (ITERASI) KE - 1 Nilai Alfa : 0.25

TSS	BOD	COD	DO	pH	fenol	Minyak dan Lemak	Skor	Kelas Aktual (T)	Jarak 1 (Ringan)	Jarak 2 (Sedang)	Jarak 3 (Berat)	Kelas Terdekat (Cj)	Keterangan
0	15.35	56.0	11.2	7.5	0.0775	0.0	-30	Sedang	56.79102	83.63013	73.59655	Ringan	BEDA
44.85	12.6	35.4	10.55	8.05	0.1105	4.25	-39	Berat	61.35344	97.22193	37.73596	Berat	SAMA
10	6.2	13.9	10.15	7.55	0.172	1.5	-30	Sedang	36.10369	119.19192	60.13104	Ringan	BEDA
6.2	11.3	35.1	16	8.45	0.086	1	-30	Sedang	59.69506	100.73291	60.66369	Ringan	BEDA
26.25	7.9	21	11.75	7	0.196	0	-22	Sedang	55.64003	108.99740	53.11776	Berat	BEDA
17.4	6.9	16.3	7.95	6.8	0.120	0.9	-20	Sedang	51.93271	114.74554	77.06194	Ringan	BEDA
21.25	6.25	13.4	7.45	6.95	0.0935	0.8	-20	Sedang	62.42295	116.94036	76.37830	Ringan	BEDA

Gambar 5.6 Antarmuka Halaman Pelatihan

22.9	5.15	10.25	12.7	7.95	0	0.3	-6	Ringan	96.16836067014	277.41002473468	71.864327989099	Berat	TIDAK SAMA
155	13.15	44	29	7.5	0.0775	6	-35	Berat	234.0790678184	289.8046706693	91.934423085016	Berat	SAMA
18	6.7	30	35	8.05	0.1105	3.6	-36	Berat	121.7408327051	251.95249057674	57.088652794804	Berat	SAMA
181	20.15	29	40	6.8	0.128	4	-45	Berat	260.61493688579	310.66805273947	122.0058622854	Berat	SAMA
186	16.4	18	40	7.4	0.188	2.85	-33	Berat	260.25489354042	325.79262468284	129.35065664734	Berat	SAMA
95	4.8	65	30	7.95	0.0915	3.75	-45	Berat	192.47461728688	245.57009490991	34.552900121622	Berat	SAMA
64	10.43	14	22	7.8	0.151	1	-32	Berat	145.23885494585	269.03323346117	45.252327059333	Berat	SAMA
40	4.9	21	29	7.55	0.1	1.2	-32	Berat	129.56840601776	262.27213695758	46.882210008966	Berat	SAMA
160	14.9	73	32	6.55	0.1575	0	-34	Berat	246.99878190536	279.75025492775	96.854520709024	Berat	SAMA
104	21.7	86	19	7.95	0.126	0	-33	Berat	202.80558709262	237.48093041584	48.492141609593	Berat	SAMA
81	20.1	80	6	8.2	0.0685	0.5	-45	Berat	184.4436481008	219.29670390551	32.897317807609	Berat	SAMA
73	19	14	19	8.25	0.085	0.8	-33	Berat	153.23467355002	268.12493081788	46.20146298468	Berat	SAMA
119	5	82	30	7.45	0.029	0.83	-35	Berat	215.03407578936	254.00878625892	60.145628198272	Berat	SAMA

## Tingkat Akurasi %

tingkat akurasi dari 50 data uji dengan jumlah data yang sama adalah 18

$$(18/50) \times 100\% = 36\%$$

Page rendered in 0.0524 seconds

Gambar 5.7 Antarmuka Halaman Pengujian



Sedang	2.935074125	15.34012775	54.403640625	12.442331125	7.75201575	0.104070075	5.042545375	-25.703792075
Berat	58.1243	17.94621	43.22754	17.03133	0.84502	0.16655	2.00559	-42.72266

AKURASI dari 5 DATA UJI													
TSS	BOD	COD	DO	pH	fenol	Minyak dan Lemak	Skor	Kelas Aktual (T)	Jarak 1 (Ringan)	Jarak 2 (Sedang)	Jarak 3 (Berat)	Kelas Terdekat (C)	Keterangan (T=C)
155	13.15	44	29	7.5	0.0775	6	-35	Berat	124.69720321046	153.61469402080	98.096426160045	Berat	SAMA
117	0.5	47	10	8.05	0.0145	3.5	-45	Berat	92.649713621620	116.17405267733	60.21187769596	Berat	SAMA
24.25	1.925	27.003	12.7	7.0	0.0001	0.4	-10	Ringan	19.166493201744	40.229471129246	52.334465740401	Ringan	SAMA
7.1	3.6	9.05	5.05	7.35	0.104	1.3	-26	Sedang	47.355307572724	46.954611631419	65.023041364457	Sedang	SAMA
15.7	5.05	19.6	6.7	7.2	0.107	2.9	-26	Sedang	38.197492029001	38.013455449935	53.79120057047	Ringan	TIDAK SAMA

Tingkat Akurasi %  
 tingkat akurasi dari 5 data uji dengan jumlah data yang sama adalah 4  
 $(4/5) \times 100\% = 80\%$

Gambar 5.8 Antarmuka Halaman Manualisasi

## BAB 6 PENGUJIAN

Bab pengujian menjelaskan mengenai tahapan analisis dari perhitungan *Learning Vector Quantization* (LVQ) untuk klasifikasi kualitas air sungai, sehingga menghasilkan keluaran berupa klasifikasi yang akurat. Pada pengujian pengklasifikasian ini digunakan 135 data yaitu dataset kualitas air sungai yang berasal dari dataset sungai di kota Malang dan dari pakar yaitu Kasubbid Pengendalian Lingkungan dan Pengelolaan Limbah Badan Lingkungan Hidup Kota Malang. Pengujian dilakukan sesuai dengan skenario pengujian yang telah dirancang pada bab sebelumnya. Pengujian data dilakukan secara bertahap menggunakan nilai yang beragam pada setiap parameter yang terdapat pada perhitungan *Learning Vector Quantization* (LVQ), yaitu mulai dari tahapan pengujian terhadap jumlah nilai *learning rate* (alfa), selanjutnya nilai pengurang alfa (*Decrement Alfa*), jumlah data latih dan data uji, nilai maksimal *epoch* dan terakhir pengujian terhadap nilai minimal alfa.

Setiap skenario dari semua paramater dilakukan pengujian sebanyak sepuluh kali, dimana setiap pengujian menggunakan dataset yang telah dipilih data latih dan data ujinya secara *random*. Misalkan, untuk pengujian pada percobaan pertama akan menggunakan data latih dan data uji yang sama untuk setiap nilai parameter alfa, selanjutnya akan dipilih data latih dan data uji secara *random* lagi untuk percobaan yang kedua dan seterusnya sampai percobaan kesepuluh. Selanjutnya, hasil akurasi dari kesepuluh percobaan akan dihitung nilai rata-ratanya dan dipilih nilai parameter yang memiliki nilai rata-rata akurasi tertinggi dengan ukuran nilai jarak (*range*) yang kecil dan dianggap menjadi nilai yang terbaik dan stabil untuk mewakili nilai parameter alfa. Sebelum memulai pengujian, ditentukan terlebih dahulu parameter awal yang menjadi indikator awal pengujian, yaitu nilai *decAlfa* = 0,1, jumlah data latih 50,33% (68 data latih : 67 data uji), maksimum *epoch* = 10 dan minimum alfa adalah  $1.0^{-10}$ . Pengujian dilakukan dengan menggunakan perhitungan akurasi pada setiap skenario pengujian sehingga dapat diketahui tingkat akurasi atau kebenaran hasil klasifikasi air sungai.

### 6.1 Hasil analisis pengujian parameter *learning rate* (alfa) awal

Pengujian ini dilakukan untuk mengetahui pengaruh dari *learning rate* atau laju pembelajaran awal terhadap hasil akurasi pengujian. Pengujian menggunakan indikator awal, yaitu *decAlfa* = 0,1, jumlah data latih 50,33% (68 data latih : 67 data uji), maksimal *epoch* = 10 dan minimal alfa adalah 0,0000000001. Nilai alfa yang digunakan adalah 0,1, 0,2, 0,3, 0,4, 0,5, 0,6, 0,7, 0,8, 0,9 dan 1. Nilai diambil pada *range* tersebut karena batasan untuk nilai alfa antara 0 sampai 1, selain itu *learning rate* (alfa) merupakan parameter jaringan dalam mengendalikan proses penyesuaian bobot. Nilai alfa yang optimal bergantung pada kasus yang dihadapi. Laju pembelajaran yang terlalu kecil menyebabkan konvergensi jaringan menjadi lebih lambat dan tidak terlihat hasil yang signifikan, sedangkan laju pembelajaran yang terlalu besar dapat menyebabkan ketidakstabilan pada jaringan untuk setiap



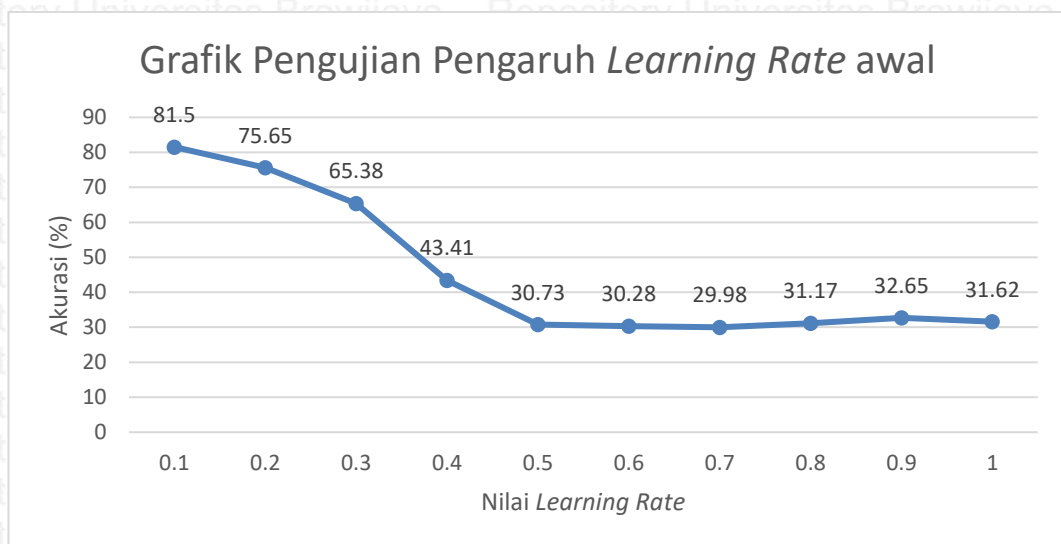


uji percobaan yang sudah dibuktikan. Pengujian dilakukan sebanyak sepuluh kali percobaan dengan data yang diacak dan sama pada tiap percobaannya. Hasil pengujian yang dihasilkan ditunjukkan Tabel 6.1.

**Tabel 6.1 Hasil Pengujian Pengaruh *Learning Rate* terhadap Hasil Akurasi**

Nilai Alfa	Hasil Akurasi (%) pada percobaan ke - i										rata-rata akurasi(%)
	1	2	3	4	5	6	7	8	9	10	
0,1	82,1	88,1	83,6	79,1	79,1	77,6	82,1	80,6	82,1	80,6	81,5
0,2	76,1	77,6	73,1	76,1	80,6	74,6	70,1	74,6	77,6	76,1	75,65
0,3	73,1	58	55,2	64,1	72,1	65,7	76,1	59,7	64,1	65,7	65,38
0,4	59,7	38,8	28,3	64,1	38,8	28,3	38,8	38,8	38,8	59,7	43,41
0,5	28,3	31,3	28,4	26,9	26,9	34,3	34,3	34,3	31,3	31,3	30,73
0,6	28,3	29,8	28,4	26,9	31,3	34,3	31,3	31,3	26,9	34,3	30,28
0,7	28,3	31,3	34,3	26,9	26,9	26,9	34,3	28,3	34,3	28,3	29,98
0,8	28,3	31,3	34,3	26,9	26,9	31,3	29,8	41,8	31,3	29,8	31,17
0,9	34,3	29,8	34,3	38,8	41,8	28,3	29,8	28,3	29,8	31,3	32,65
1	34,3	29,8	47,7	26,9	26,9	34,3	31,3	29,8	28,3	26,9	31,62

Untuk mengetahui dan memudahkan analisis hasil pengujian pengaruh *learning rate* terhadap akurasi, akan dijelaskan dengan grafik pada Gambar 6.1.



**Gambar 6.1 Pengujian Pengaruh *Learning Rate* terhadap Hasil Akurasi**

Pada proses pembelajaran, nilai *learning rate* mempengaruhi perubahan nilai bobot. Semakin besar nilai *learning rate*, maka semakin besar pula perubahan



nilai bobot dan jarak antara bobot awal (data *centroid*) dengan bobot optimal semakin besar. Selain itu, pengujian nilai *learning rate* akan mempengaruhi lamanya proses perhitungan jarak dan pembelajaran serta perubahan nilai bobot. Berdasarkan grafik pada Gambar 6.1, rata-rata akurasi terbaik dari 10 kali percobaan dihasilkan oleh pengujian nilai *learning rate* awal sebesar 0,1 dengan akurasi 81,5%. Pada grafik menunjukkan tren perubahan yang terus menurun dan stabil pada nilai 0,5 sampai 1. Sehingga pada kasus ini, dengan semakin besar nilai *learning rate* maka semakin rendah akurasi yang dihasilkan. Dengan demikian nilai 0,1 dijadikan sebagai parameter yang terbaik dan stabil untuk nilai *learning rate* awal pada proses pengujian selanjutnya.

## 6.2 Hasil analisis pengujian parameter pengurang *learning rate* (*decAlfa*)

Pengujian ini bertujuan untuk mengetahui besarnya pengaruh yang dihasilkan dari pengurangan *learning rate* (*decAlfa*) terhadap hasil akurasi dan menentukan nilai pengurang *learning rate* yang memiliki akurasi tertinggi. Pengujian menggunakan nilai *decAlfa* = 0,1, jumlah data latih 50,33% (68 data latih : 67 data uji), maksimal *epoch* = 10, minimal *epoch* = 0,0000000001 dan nilai alfa yang terbaik dari pengujian sebelumnya yaitu 0,1. Koefisien pengurang *learning rate* berfungsi untuk mengkonvergenkan nilai perubahan bobot. Nilai dari *decAlfa* ini berkisar antara 0 sampai 1, namun jika nilai yang diambil terlalu kecil maka tidak terlihat signifikan untuk setiap uji percobaan yang sudah dibuktikan. Sehingga nilai *decAlfa* yang digunakan adalah 0,1, 0,2, 0,3, 0,4, 0,5, 0,6, 0,7, 0,8, 0,9, 1. Pengujian dilakukan sebanyak sepuluh kali percobaan dengan data yang diacak dan sama pada tiap percobaannya. Hasil pengujian yang dihasilkan ditunjukkan Tabel 6.2.

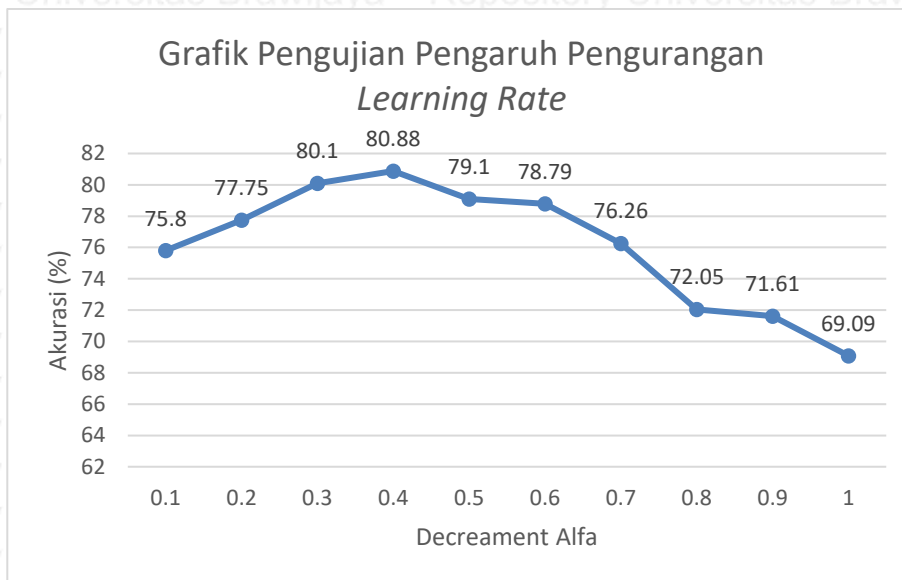
**Tabel 6.2 Hasil Pengujian Pengaruh Pengurang *Learning Rate* terhadap Akurasi**

<i>decAlpha</i>	Hasil akurasi pada percobaan ke=i										rata-rata akurasi (%)
	1	2	3	4	5	6	7	8	9	10	
0,1	71,6	82,1	70,1	77,6	70,1	76,1	74,6	74,6	85,1	76,1	75,8
0,2	80,6	82,1	71,6	80,6	68,6	76,1	79,1	80,6	80,6	77,6	77,75
0,3	82,1	88,1	70,1	80,6	83,6	76,1	80,6	80,6	80,1	79,1	80,1
0,4	85,1	82,1	71,6	86,5	86,5	76,1	79,1	79,1	83,6	79,1	80,88
0,5	80,6	77,6	71,6	83,6	83,6	76,1	79,1	77,6	85,1	76,1	79,1
0,6	80,6	82,1	68,6	79,1	82,1	76,1	79,1	82,1	86,5	71,6	78,79
0,7	77,6	77,6	67,2	74,6	85,1	76,1	77,6	79,1	77,6	70,1	76,26
0,8	70,1	73,1	65,6	70,1	65,6	74,6	77,6	76,1	79,1	68,6	72,05
0,9	64,2	68,6	65,6	70,1	79,1	74,6	76,1	73,1	76,1	68,6	71,61



1	62,7	68,6	65,6	67,2	77,6	74,6	62,7	67,2	79,1	65,6	69,09
---	------	------	------	------	------	------	------	------	------	------	-------

Untuk memudahkan analisis hasil pengujian pengaruh pengurang *learning rate* terhadap akurasi, akan dijelaskan dengan grafik pada Gambar 6.2.



**Gambar 6.2 Pengujian Pengaruh Pengurang *Learning Rate* terhadap Hasil Akurasi**

Pada proses pembelajaran, nilai pengurang *learning rate* mempengaruhi konvergensi nilai bobot, pada saat pengurang *learning rate* memiliki nilai yang besar, maka nilai nilai *learning rate* akan lebih cepat mendekati nilai minimum *alpha*. Jika nilai *learning rate* sudah mendekati atau sama dengan nilai minimum *learning rate*, maka hampir tidak ada jarak antara bobot awal dengan bobot optimal atau nilai bobot telah konvergen. Berdasarkan grafik pada Gambar 6.2, rata-rata akurasi terbaik dari 10 kali percobaan dihasilkan oleh pengujian nilai pengurang *learning rate* awal sebesar 0,4 dengan akurasi 80,88%. Pada grafik menunjukkan tren perubahan yang terus naik mulai dari nilai 0,1 sampai 0,4 dan tren perubahan yang semakin turun dengan akurasi yang lebih rendah mulai nilai 0,5 sampai 1. Dengan demikian nilai 0,4 dijadikan sebagai parameter yang terbaik dan stabil untuk nilai pengurang *learning rate* pada proses pengujian selanjutnya.

### 6.3 Hasil analisis pengujian parameter jumlah data latih dan data uji

Pengujian pengaruh jumlah data latih dan data uji merupakan proses untuk mengetahui rata-rata tingkat akurasi terbaik dari 10 kali percobaan pada masing-masing perubahan jumlah data. Perubahan jumlah data merupakan nilai yang digunakan untuk membagi dataset menjadi data latih dan data uji. Pengujian menggunakan nilai alfa dan *decAlfa* yang terbaik dari pengujian sebelumnya yaitu 0,1 dan 0,4, nilai maksimal *epoch* = 10 dan minimal alfa = 0,000000001. Perubahan jumlah data dari 135 dataset klasifikasi kualitas air sungai yang



digunakan dalam pengujian ini adalah berupa perbandingan jumlah data latih dan data uji yaitu 130:5, 125:10, 110:25, 100:35, 85:50, 68:67, 50:85, 35:100, 25:110, 10:125 atau dalam persentase 7,4%, 18,5%, 25,9%, 37%, 50,3%, 62,9%, 74,1%, 81,4% dan 92.5% jumlah data latih dari seluruh dataset untuk masing-masing kelas. Jumlah data diambil dari *range* tersebut karena jika jumlah data diambil terlalu kecil maka tidak terlihat data signifikan untuk setiap uji percobaan yang telah dibuktikan. Pengujian dilakukan sebanyak sepuluh kali percobaan dengan data yang diacak dan sama pada tiap percobaannya. Pengujian dilakukan dengan mengambil data latih secara acak dari seluruh dataset, kemudian mengambil data uji dari sisa dataset yang telah digunakan sebagai data latih. Hasil pengujian yang dihasilkan ditunjukkan Tabel 6.3.

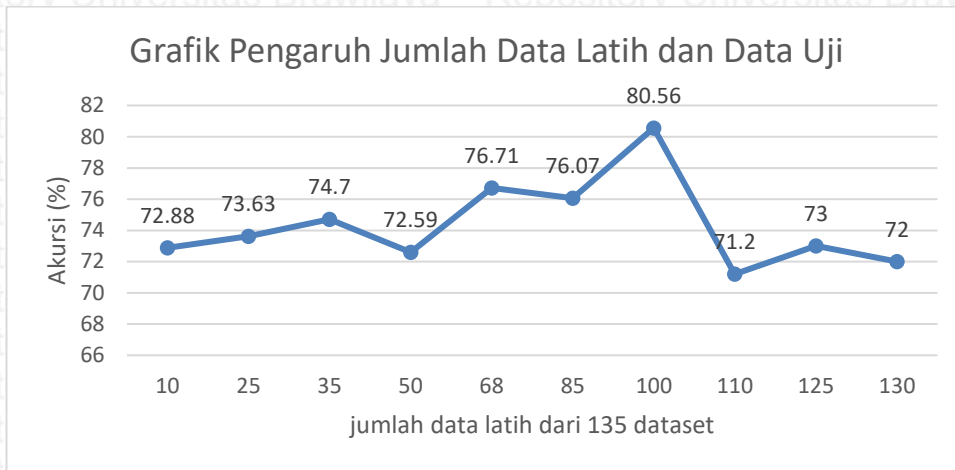
**Tabel 6.3 Hasil Pengujian Pengaruh Jumlah Data Latih dan Data Uji terhadap Akurasi**

Jumlah data latih	Hasil akurasi pada percobaan ke-i										rata-rata akurasi (%)
	1	2	3	4	5	6	7	8	9	10	
10	73,6	60,8	75,2	76,8	71,2	79,2	77,6	67,2	68,8	78,4	72,88
25	75,4	76,4	67,3	78,2	65,4	79,1	80,9	74,5	60,9	78,2	73,63
35	75	80	68	78	61	74	82	85	66	78	74,7
50	74,1	73,1	82,3	75,3	75,3	62,3	75,3	65,9	67	75,3	72,59
68	76,1	71,6	83,6	85,1	76,1	80,6	76,1	61,2	77,6	79,1	76,71
85	74	77,6	76	84	84	86	66	74	60	79,1	76,07
100	80	88,6	85,7	85,7	74,3	85,7	77,1	68,6	77,1	82,8	80,56
110	80	68	76	76	56	72	72	72	64	76	71,2
1	70	70	70	60	80	80	60	80	90	70	73
130	60	70	80	60	80	70	80	80	60	80	72

Untuk melakukan analisis hasil pengujian pengaruh jumlah data latih dan data uji terhadap akurasi, akan dijelaskan dengan grafik pada Gambar 6.3. Pada proses *training* data, pengujian jumlah data latih mempengaruhi nilai akurasi. Secara umum, semakin besar jumlah data latih yang digunakan, maka semakin banyak data yang digunakan sebagai acuan pembelajaran dan semakin tinggi tingkat akurasi yang dihasilkan. Hal tersebut disebabkan karena semakin banyak variasi data latih yang digunakan dengan ketentuan tergantung ukuran dan sebaran data latih tersebut. Berdasarkan grafik pada Gambar 6.3, rata-rata akurasi terbaik dan stabil dari 10 kali percobaan dihasilkan dengan jumlah data latih sebesar 100 dan data uji sebesar 35 dengan akurasi 80,56%. Pada grafik menunjukkan tren perubahan yang terus naik dan stabil dari jumlah data latih 10 sampai 85 dan tren



perubahan yang semakin naik dan mencapai akurasi maksimum pada jumlah data latih 100 dan semakin turun pada jumlah data 110 sampai 130. Hal tersebut dikarenakan penggunaan jumlah data uji yang sedikit pada jumlah data latih 110, 125 dan 130, sehingga menyebabkan hasil akurasi dari data uji yang sedikit dan tidak stabil. Dengan demikian jumlah data 100 berbanding 35 dijadikan sebagai parameter yang terbaik dan stabil untuk nilai perbandingan data latih dan data uji pada proses pengujian selanjutnya.



**Gambar 6. 3 Pengujian Pengaruh Jumlah Data Latih dan Data Uji terhadap Hasil Akurasi**

#### 6.4 Hasil analisis pengujian parameter maksimal perulangan (*epoch*)

Pengujian pengaruh maksimum *epoch* merupakan proses untuk mengetahui rata-rata tingkat akurasi terbaik pada masing-masing nilai maksimal *epoch*. Nilai maksimal *epoch* yang digunakan adalah 1, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 40, 60. Nilai maksimal *epoch* diambil dari *range* tersebut karena jika diambil terlalu kecil dan terlalu besar maka tidak terlihat signifikan untuk setiap percobaan yang sudah dibuktikan. Pengujian dilakukan sebanyak sepuluh kali percobaan dengan data yang diacak dan sama pada tiap percobaannya. Dengan parameter alfa = 0,1, *decAlfa* = 0,4 dan jumlah data latih 100 berbanding 35 data uji yang diambil dari nilai terbaik pada pengujian sebelumnya serta nilai minimal alfa adalah 0,0000000001. Hasil pengujian yang dihasilkan ditunjukkan Tabel 6.4.

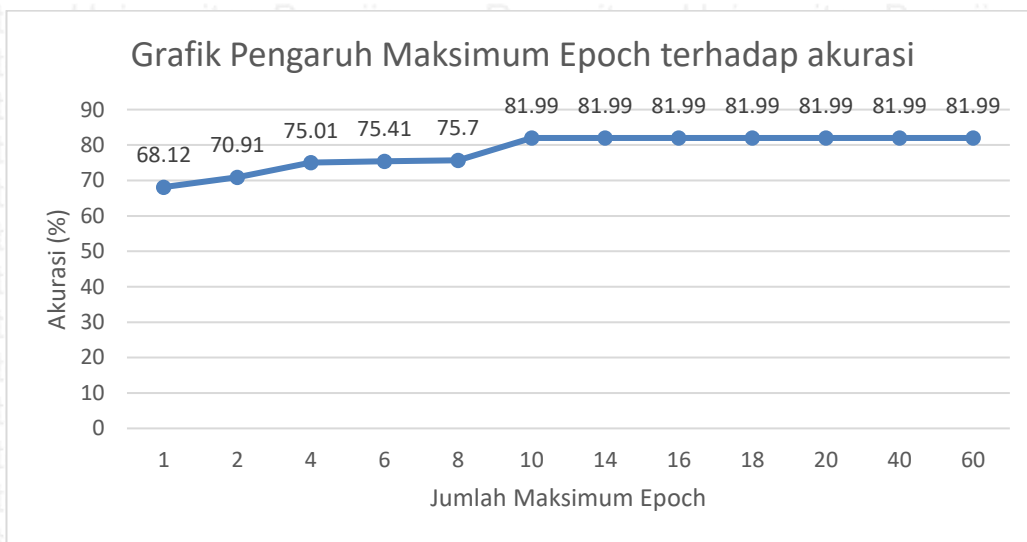
**Tabel 6. 4 Hasil Pengujian Pengaruh Maksimum *Epoch* terhadap Akurasi**

Maksimum <i>Epoch</i>	Hasil akurasi pada percobaan ke-i										rata-rata akurasi(%)
	1	2	3	4	5	6	7	8	9	10	
1	61,4	71,4	71,4	74,3	57,1	65,7	71,4	62,8	80	65,7	68,12
2	60	77,1	71,4	81,8	65,7	71,4	81,8	62,8	65,7	71,4	70,91
4	71,4	82,8	77,1	85,7	77,1	65,7	81,8	65,7	71,4	71,4	75,01



6	74,3	77,1	77,1	88,6	65,7	77,1	77,1	68,6	71,4	77,1	75,41
8	77,1	77,1	77,1	85,7	88,6	71,4	65,7	68,6	68,6	77,1	75,7
10	80	88,6	77,1	88,6	85,7	85,7	85,7	68,6	77,1	82,8	81,99
12	80	88,6	77,1	88,6	85,7	85,7	85,7	68,6	77,1	82,8	81,99
14	80	88,6	77,1	88,6	85,7	85,7	85,7	68,6	77,1	82,8	81,99
16	80	88,6	77,1	88,6	85,7	85,7	85,7	68,6	77,1	82,8	81,99
18	80	88,6	77,1	88,6	85,7	85,7	85,7	68,6	77,1	82,8	81,99
20	80	88,6	77,1	88,6	85,7	85,7	85,7	68,6	77,1	82,8	81,99
40	80	88,6	77,1	88,6	85,7	85,7	85,7	68,6	77,1	82,8	81,99
60	80	88,6	77,1	88,6	85,7	85,7	85,7	68,6	77,1	82,8	81,99

Untuk melakukan analisis hasil pengujian pengaruh maksimum *epoch* terhadap akurasi, akan dijelaskan dengan grafik pada Gambar 6.4.



**Gambar 6. 4 Pengujian Pengaruh Maksimum *Epoch* terhadap Hasil Akurasi**

Maksimum *epoch* digunakan sebagai kondisi berhenti (*stop condition*) pada setiap perulangan dalam perhitungan *Learning Vector Quantization* (LVQ). Berbeda dengan pengurang *learning rate*, semakin besar maksimum *epoch* maka akan memperpanjang dan memperlama proses konvergensi pada bobot, karena maksimum *epoch* akan memperpanjang proses perulangan. Berdasarkan grafik pada Gambar 6.4, rata-rata akurasi terbaik dan stabil dari 10 kali percobaan dihasilkan oleh maksimum *epoch* sebesar 10 sampai 60 dengan akurasi 81,9%. Pada grafik menunjukkan tren perubahan yang terus naik mulai dari *epoch* 1 dan mencapai akurasi maksimum pada *epoch* 10 dan selanjutnya terdapat tren perubahan yang linier dan sama yaitu mulai *epoch* 10 sampai *epoch* dengan ukuran besar yaitu 60 yang sudah mencapai konvergen. Hal tersebut disebabkan oleh pengaruh nilai alfa dan *decAlfa* yang kecil sehingga mulai dari iterasi 10 dan



seterusnya menyebabkan nilai *update* bobot data yang sangat kecil dan tidak terlalu berpengaruh. Iterasi maksimum yang dicapai adalah *epoch* 41 karna telah mencapai kondisi berhenti karna nilai alfa lebih kecil dari 0,0000000001. sehingga nilai *epoch* dengan akurasi terbaik dan stabil dipengaruhi oleh nilai alfa dan *decAlfa*. Dengan demikian *maxEpoch* sebesar 10 dengan nilai yang paling kecil dari nilai hasil yang sudah mencapai konvergen tersebut dijadikan sebagai parameter yang terbaik dan stabil dan sebagai data uji pada proses pengujian selanjutnya.

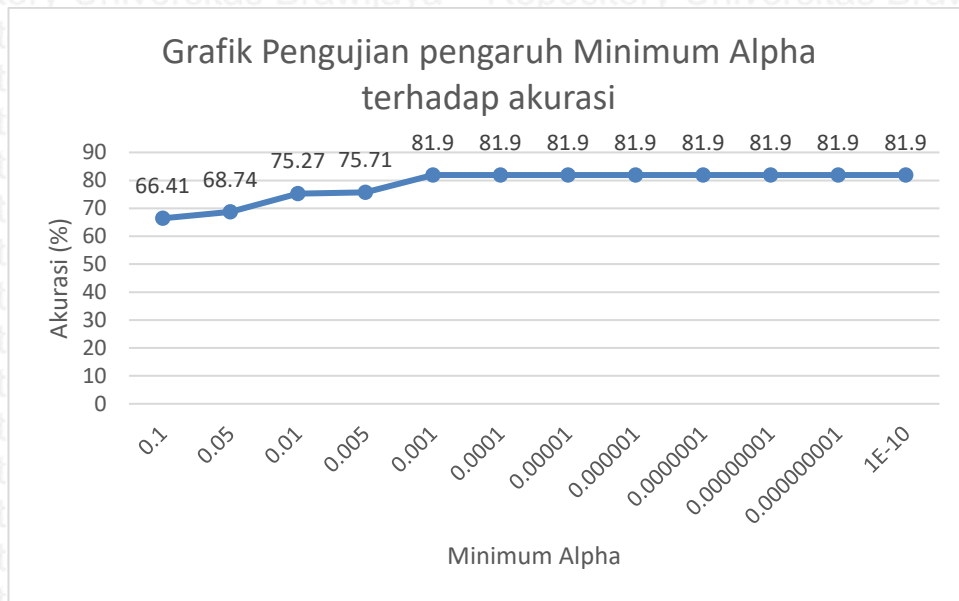
### 6.5 Hasil analisis pengujian parameter minimal alfa

Pengujian ini dilakukan untuk mengetahui pengaruh dari laju *learning rate* (alfa) minimum terhadap akurasi pengujian. Pengujian dilakukan menggunakan hasil parameter-parameter terbaik berdasarkan percobaan sebelumnya, yaitu alfa = 0,1, *decAlfa* = 0,4 dan jumlah data latih 100 berbanding 35 data uji dan *maxEpoch* = 10. Minimum alfa yang digunakan adalah 0,1, 0,05, 0,01, 0,005, 0,001, 0,0005,  $1.0^{-4}$ ,  $1.0^{-5}$ ,  $1.0^{-6}$ ,  $1.0^{-7}$ ,  $1.0^{-8}$ ,  $1.0^{-9}$ ,  $1.0^{-10}$ . Pengujian dilakukan sebanyak sepuluh kali percobaan dengan data yang diacak dan sama pada tiap percobaannya. Hasil pengujian yang dihasilkan ditunjukkan Tabel 6.5.

**Tabel 6. 5 Hasil Pengujian Pengaruh Minimum Alfa terhadap Akurasi**

Minimum <i>epoch</i>	Hasil akurasi pada percobaan ke-i										rata-rata akurasi(%)
	1	2	3	4	5	6	7	8	9	10	
0,1	54,3	77,1	54,3	77,1	74,3	57,1	65,7	71,4	61,4	71,4	66,41
0,05	65,7	68,6	68,6	65,7	65,7	71,4	81,8	62,8	65,7	71,4	68,74
0,01	82,8	74,3	64,3	77,1	77,1	88,6	65,7	77,1	77,1	68,6	75,27
0,005	82,8	74,3	74,3	74,3	71,4	88,6	77,1	74,3	71,4	68,6	75,71
0,001	82,8	74,3	80	88,6	88,6	88,6	85,7	68,6	85,7	76,1	81,9
0,0005	82,8	74,3	80	88,6	88,6	88,6	85,7	68,6	85,7	76,1	81,9
0,0001	82,8	74,3	80	88,6	88,6	88,6	85,7	68,6	85,7	76,1	81,9
0,00001	82,8	74,3	80	88,6	88,6	88,6	85,7	68,6	85,7	76,1	81,9
0,000001	82,8	74,3	80	88,6	88,6	88,6	85,7	68,6	85,7	76,1	81,9
0,0000001	82,8	74,3	80	88,6	88,6	88,6	85,7	68,6	85,7	76,1	81,9
0,00000001	82,8	74,3	80	88,6	88,6	88,6	85,7	68,6	85,7	76,1	81,9
0,000000001	82,8	74,3	80	88,6	88,6	88,6	85,7	68,6	85,7	76,1	81,9
$1.0^{-10}$	82,8	74,3	80	88,6	88,6	88,6	85,7	68,6	85,7	76,1	81,9

Untuk melakukan analisis hasil pengujian pengaruh mainimum alfa terhadap akurasi, akan dijelaskan dengan grafik pada Gambar 6.5.



**Gambar 6. 5 Pengujian Pengaruh Minimum Alfa terhadap Hasil Akurasi**

Sama halnya dengan maksimal *epoch*, minimal alfa digunakan sebagai kondisi berhenti (*stop condition*) pada setiap perulangan dalam perhitungan *Learning Vector Quantization* (LVQ). Semakin kecil nilai minimum alfa maka akan memperpanjang proses konvergensi pada data bobot. Berdasarkan Gambar 6.5, tren akurasi naik ditunjukkan mulai dari *minAlfa* 0,1 sampai 0,001 dan menjadi stabil linier dan selalu sama sampai nilai *minAlfa* terkecil yaitu  $1.0^{-10}$ . Hal tersebut terjadi karena pengaruh dari nilai alfa 0,1, pengurangan *decAlfa* 0,4 tiap iterasinya dan *epoch* 10 yang digunakan, sehingga pada *decAlfa* 0,1 sudah berhenti pada iterasi 1, *decAlfa* 0,05 berhenti pada iterasi 2, *decAlfa* 0,01 berhenti pada iterasi 5, *decAlfa* 0,005 berhenti pada iterasi 6 dan *decAlfa* 0,001 sampai  $1.0^{-10}$  berhenti pada iterasi yang sama yaitu *epoch* 10. Sehingga dapat diambil batas awal dan minimal nilai *minAlfa* yang mencapai akurasi terbesar adalah pada nilai terkecil yaitu 0,01 dengan akurasi 81,9%. Sehingga nilai 0,001 menjadi solusi terbaik dan stabil pada parameter minimal alfa.

## 6.6 Hasil analisis pengujian akurasi dan validasi parameter optimal

Pengujian ini dilakukan untuk mengetahui akurasi akhir dari implementasi *Learning Vector Quantization* (LVQ) untuk klasifikasi kualitas air sungai. Pengujian ini menggunakan parameter-parameter hasil pengujian yang dilakukan sebelumnya. Sehingga hasil akurasi klasifikasi dari parameter tersebut bisa menjadi hasil akhir klasifikasi sistem secara umum. Parameter optimal yang digunakan tersebut adalah alfa 0,1, *decAlfa* 0,4, *epoch* 10, perbandingan data latih dan data uji adalah 100 berbanding 35 dari 135 dataset dan *minAlfa* 0,001. Pengujian dilakukan sebanyak sepuluh kali percobaan dengan data yang diacak dan sama pada tiap percobaannya. Hasil pengujian yang dihasilkan ditunjukkan Tabel 6.6.



**Tabel 6. 6 Hasil Pengujian Akurasi dan Validasi Parameter Optimal**

Hasil akurasi pada percobaan ke-i										Maks	Min	rata-rata akurasi(%)
1	2	3	4	5	6	7	8	9	10			
77,1	71,4	80	88,6	80	88,6	68,5	82,8	88,6	85,7	88,6	68,5	81,13

Berdasarkan Tabel 6.6 dapat disimpulkan bahwa hasil pengujian LVQ untuk akurasi klasifikasi kualitas air sungai secara umum menggunakan parameter-parameter optimal menghasilkan akurasi tertinggi dan stabil dengan rata-rata 81,13%.

## BAB 7 PENUTUP

### 7.1 Kesimpulan

Berdasarkan penelitian yang telah diimplementasikan dan dilakukan pengujian, maka terdapat beberapa kesimpulan yang dapat diambil, yaitu:

1. Proses awal yang harus dilakukan dalam mengimplementasikan *Learning Vector Quantization* (LVQ) untuk klasifikasi kualitas air sungai adalah mengumpulkan dataset klasifikasi air sungai dengan batasan yang telah ditentukan. Delapan fitur yang diambil dari data kualitas air sungai adalah TSS, BOD, COD, DO, minyak dan lemak, pH, Fenol dan skor dengan keluaran yang dihasilkan yaitu klasifikasi kategori kualitas air sungai dengan tingkat memenuhi baku mutu (kondisi baik), tercemar ringan, tercemar sedang dan tercemar berat. Proses klasifikasi kualitas air sungai diawali dengan memecah seluruh dataset secara acak menjadi tiga bagian, yaitu sebagai data bobot awal (*centroid*), data latih (*training*) dan data uji (*testing*). Selanjutnya, dilakukan proses pelatihan pada data latih menggunakan metode LVQ dengan menghitung jarak tiap data latih dengan data bobot dan melakukan *update* nilai data bobot sehingga mendapatkan data bobot akhir yang optimal. Bobot akhir yang optimal tersebut digunakan untuk menguji dan mengklasifikasikan hasil klasifikasi kualitas air sungai.

2. Berdasarkan pengujian yang telah dilakukan yaitu pengujian terhadap nilai *alpha* (*learning rate*), nilai pengurang *learning rate* (*decalpha*), perbandingan jumlah data latih dan data uji, maksimum *epoch* (*maxEpoch*) dan minimum *alpha* (*minAlpha*) pada percobaan dengan data acak dan sama untuk setiap percobaan ujinya. Hasil pengujian yang didapatkandari lima skenario pengujian tersebut adalah 0,1 untuk nilai *alpha* (*learning rate*), 0,4 untuk nilai pengurang *learning rate* (*decalpha*), 100 berbanding 35 untuk perbandingan jumlah data latih dan data uji, 10 untuk maksimum *epoch* (*maxEpoch*) dan 0,001 untuk nilai minimum *alpha* (*minAlpha*) dengan akurasi rata-rata akhir sebesar 81,13%.

### 7.2 Saran

Peneliti menganggap bahwa penelitian ini masih jauh dari sempurna, oleh karena itu berikut adalah beberapa saran yang dapat peneliti berikan untuk pengembangan penelitian selanjutnya.

1. Perlu dilakukan penelitian lebih lanjut untuk parameter kualitas air sungai lainnya, dengan menambahkan parameter fisika, kimia dan biologi berdasarkan tingkat ketelitian dan kondisi daerah air sungai lainnya sebagai fitur masukan.
2. Perlu dilakukan penelitian lebih lanjut agar sistem dapat melakukan klasifikasi air sungai berdasarkan lebih dari 8 fitur parameter pencemaran air sungai.



3. Perlu dilakukan penelitian lebih lanjut dengan menggunakan dataset kualitas air sungai yang lebih banyak dan bervariasi pada daerah – daerah atau kota – kota lainnya. Sehingga diketahui kehandalan perhitungan sistem dalam mengklasifikasikan kualitas air sungai pada berbagai daerah dan berbagai kondisi sungai di Indonesia.

4. Perlu dilakukan penelitian lebih lanjut dengan membandingkan hasil klasifikasi LVQ tidak hanya dengan hasil laboratorium yang akurat tetapi juga dengan metode-metode manual yang terdapat pada klasifikasi air sungai, seperti membandingkan dengan metode *STORET*, *WQI* dan *IP*. Sehingga dapat dijadikan solusi dan perbandingan terhadap kelemahan pada setiap metode manual tersebut.

5. Sistem dapat dikembangkan menjadi sistem informasi terpadu untuk membantu instansi terkait pengolahan dan penyimpana data klasifikasi air sungai.

6. Sistem dapat dikembangkan dengan metode LVQ2 dan LVQ3 untuk mendapatkan hasil akursi yang lebih baik dan minimal error yang lebih kecil.

7. Sistem dapat dikembangkan dengan membandingkan metode LVQ dengan metode Jaringan Syaraf Tiruan (*JST*) lainnya. Sehingga didapatkan metode *JST* yang paling optimal untuk data klasifikasi air sungai.

## DAFTAR PUSTAKA

- Agustira, Riyanda, Sari, K. L., & Jamilah, 2013. *Kajian Karakteristik Kimia Air, Fisika Air dan Debit Sungai pada Kawasan DAS Padang Akibat Pembuangan Limbah Tapioka*. *Jurnal Online Agroekoteknologi Vol.1 , No.3*. pp. 615-625.
- Areerachakul, Sirilak & Sanguansintukul, Siripun, 2009. *Water Quality Classification using Neural Networks: Case Study of Canals in Bangkok, Thailand*. Chulalongkorn University, Thailand.
- Hartono, Djoko M., Sulistyoweni & Sutjningsih, Dwita, 2009. *Penentuan Indikator Pencemaran Air dengan Pendekatan Indeks Kualitas Air pada Air Baku Air Minum dari Saluran Tarum Barat*. Departemen Teknik Sipil, Fakultas Teknik, Universitas Indonesia, Depok.
- Islamiah, Nurul, Chollisodin, I. & Mahmudy, W. F., 2016. *Klasifikasi Penyakit Gigi dan Mulut menggunakan Metode Learning Vector Quantization (LVQ)*. Fakultas Ilmu Komputer, Universitas Brawijaya, Malang.
- Jamadar, Aliaja & Kakade, Vaibhav, 2015. *Brain Tumor Detection and Analysis using SVM and LVQ Classifier* International Journal for Scientific Research & Development, Vol. 3, Issue 05. Pp. 754-757.
- Keputusan Menteri Negara Lingkungan Hidup nomor 115 Tahun 2003 *tentang Pedoman Penentuan Status Mutu Air*. Menteri Negara Lingkungan Hidup, Jakarta.
- Nugroho, S.P. 2008. *Analisis Kualitas Air Danau Kaskade Sebagai Sumber Imbuhan Waduk Resapan di Kampus UI Depok*. *Jurnal Sains dan Teknologi Indonesia*, Vol. 10. Pp. 99-105.
- Nugroho, F. F. Kurniawan, Arif, Y. M.& Dermawan, D. A., 2011. *Simulasi Multi Atribut di Dasarkan pada Agen untuk Kehandalan Distribusi Energi Listrik menggunakan Metode LVQ*. Seminar on Electrical, Informatics and ITS education, Pp. 55-63.
- Peraturan Menteri Negara Lingkungan Hidup nomor 01 Tahun 2010 *tentang Tata Laksana Pengendalian Pencemar Air*. Menteri Negara Lingkungan Hidup, Jakarta.
- Peraturan Pemerintah Republik Indonesia nomor 82 Tahun 2001 *tentang Pengelolaan Kualitas Air dan Pengendalian Pencemaran Air*. Lembaga Negara Republik Indonesia, Jakarta.
- Puspitaningrum, D., 2006. *Pengantar Jaringan Syaraf Tiruan*. Penerbit Andi, Yogyakarta.
- Rao, D. Rajeswara & Murthy, J., 2016. *Comparative Analysis: Heart Diagnosis Classification using Bp-LVQ Neural Network Models for Analog and Digital Data*. *Global Journal of Computer Science and Technology ( E ) Volume XVI Issue V Version I*.



Ramadhani, Febian, T., W., Harisuseno, Donny & Yuliani, Emma, 2016. *Penerapan Metode Water Quality Index (WQI) dan Metode STORET untuk Menentukan Status Mutu Air pada Ruas Sungai Brantas Hilir*. Fakultas Teknik, Universitas Brawijaya, Malang.

Sofia, Y., Tontowi, dan S. Rahayu. 2010. *Penelitian Pengolahan Air Sungai Yang Tercemar Oleh Bahan Organik*. Jurnal Sumber Daya Air, Vol. 6. Pp. 145-160.

Undang-undang Republik Indonesia nomor 32 Tahun 2009 tentang *Perlindungan dan Pengelolaan Lingkungan Hidup*. Kementerian Sekretariat Negara Republik Indonesia, Jakarta.

Wendyartaka, Anung, 2016. *Air Sungai di Indonesia Tercemar Berat*. Kompas Print 29 April 2016. <http://print.kompas.com/baca/2016/04/29/Air-Sungai-di-Indonesia-Tercemar-Berat> Tanggal akses: 12 Februari 2017.

Wuryandari, M. D. & Afrianto, I., 2012. Perbandingan Metode Jaringan Syaraf Tiruan Backpropagation dan Learning Vector Quantization pada Pengenalan Wajah. Jurnal Komputer dan Informatika vol. I No. 01, Pp. 45-51.

Ying, Zhang & Mei, Li, 2016. *An Evaluation Model of Water Quality Based of Learning Vector Quantization Neural Network*. Proceedings of the 35th Chinese Control Conference, Pp.3658-3689.

Yusrizal, Heri, 2015. *Efektifitas Metode Perhitungan STORET, IP dan CCME WQI dalam Menentukan Status Kualitas Air Way Sekampung Provinsi Lampung*. Tesis Magister Ilmu Lingkungan, Universitas Lampung, Bandar Lampung.

## LAMPIRAN

### Lampiran 1 Form Validasi Dataset



**KEMENTERIAN RISET, TEKNOLOGI, DAN PENDIDIKAN TINGGI**  
**UNIVERSITAS BRAWIJAYA**  
**FAKULTAS ILMU KOMPUTER**

Gedung A FILKOM Lt. 1, Jl. Veteran No.8, Malang, 65145, Indonesia  
Telp. : +62-341-577911; Fax : +62-341-577911  
http://filkom.ub.ac.id E-mail : filkom@ub.ac.id

Nomor : 314 /UN10.F15.05/PP/2017  
Perihal : *Permohonan Data Skripsi*

02 JUN 2017

**Yth. Kasubbid Pengendalian Lingkungan dan Pengelolaan Limbah, Badan Lingkungan Hidup Kota Malang**  
**Jl. Mojopahit No. 1C**  
**Malang**

Untuk mendukung penyelesaian skripsi mahasiswa berikut :

**Nama** : Rifwan Hamidi  
**NIM** : 13515020011149  
**Judul Skripsi** : Implementasi Learning Vector Quantization (LVQ) untuk Klasifikasi Kualitas Air Sungai  
**Dosen Pembimbing** : 1. M. Tanzil Furqon, S.Kom, M.CompSc  
2. Bayu Rahayudi, S.T, M.T  
**Prodi** : Teknik Informatika

Guna melakukan survey atau pengambilan data untuk skripsi mahasiswa tersebut di instansi saudara, jenis survey atau data yang diperlukan dan rencana waktu pelaksanaan adalah :

**Data** : Melakukan konsultasi, survey, pengambilan data dan kesediaan Menjadi pakar untuk klasifikasi air sungai  
**Waktu** : 24 Mei - 24 Juni 2017

Atas perhatian dan kerjasamanya kami ucapkan terima kasih.

Mengetahui,  
Dosen Pembimbing I

M. Tanzil Furqon, S.Kom, M.CompSc  
NIK/ NIP. 198209302008011004

a.n. Dekan  
Plt. Kepala Tata Usaha,



Njomo, S.AB  
NIP. 196008061980031001

**Tembusan Kepada Yth:**

1. Ketua Program Studi Teknik Informatika
2. Mahasiswa yang bersangkutan