

BAB II

LANDASAN TEORI

2.1 Stroke

Stroke adalah serangan otak yang timbul secara mendadak dimana terjadi gangguan fungsi otak sebagian atau menyeluruh sebagai akibat dari gangguan aliran darah oleh karena sumbatan atau pecahnya pembuluh darah tertentu di otak, sehingga menyebabkan sel-sel otak kekurangan darah, oksigen atau zat-zat makanan dan akhirnya dapat terjadi kematian sel-sel tersebut dalam waktu relatif singkat (Yastroki, 2011).

Stroke merupakan gangguan fungsi saraf pusat yang berkembang sangat cepat baik menit maupun jam dengan perburukan ringan sampai berat kemudian menetap atau bahkan membaik secara cepat atau perlahan-lahan tergantung tingkat keparahan stroke dan cepat serta tepatnya intervensi pengobatan. Karena setiap bagian otak memiliki fungsi-fungsi tertentu, maka gejala dan tanda stroke pada setiap individu sangat bervariasi, tergantung pembuluh darah mana yang terkena dan bagian otak mana yang terganggu (Yastroki, 2011).

2.2 Faktor Risiko Penyakit Stroke

Faktor risiko stroke adalah kondisi atau penyakit atau kelainan yang terdapat pada seseorang yang memiliki potensi untuk memudahkan orang tersebut mengalami serangan stroke pada suatu saat (Yastroki, 2011). Jika seseorang terdapat faktor-faktor risiko untuk terjadinya serangan stroke disebut sebagai *stroke prone profile* (Kemenkes, 2013).

Terdapat dua macam faktor risiko penyakit stroke, yaitu faktor risiko yang dapat diubah atau dikendalikan dan faktor risiko yang tidak dapat diubah atau dikendalikan. Faktor risiko yang tidak dapat diubah atau dikendalikan (Vallesia, S.A. 2015) meliputi:

1. Usia

Stroke sering terjadi pada orang yang telah lanjut usia (tua). Setiap penambahan 10 tahun setelah usia 55 tahun, terdapat peningkatan risiko penyakit stroke sebanyak dua kali lipat.

2. Jenis Kelamin

Stroke lebih mungkin pada pria dibandingkan pada wanita. Namun, lebih dari separuh kematian stroke total yang terjadi pada wanita. Penggunaan pil KB dan kehamilan meningkatkan risiko stroke bagi perempuan.

3. Ras

Kematian akibat penyakit stroke lebih banyak terjadi pada orang Afrika-Amerika daripada orang kulit putih. Hal ini dikarenakan mereka mempunyai risiko lebih tinggi menderita tekanan darah tinggi, diabetes, dan obesitas.

Sedangkan faktor risiko yang dapat diubah atau dikendalikan (Valensia, S.A. 2015) meliputi:

1. Tekanan Darah

Tekanan darah adalah tekanan yang terjadi pada pembuluh darah arteri saat darah dipompa oleh jantung untuk dialirkan ke seluruh tubuh. Tekanan darah sistolik adalah tekanan darah yang terjadi pada saat otot jantung berkontraksi. Sedangkan tekanan darah diastolik adalah tekanan darah yang terjadi pada saat otot jantung beristirahat atau tidak sedang berkontraksi

2. Kadar Gula Darah

Gula darah adalah bahan bakar tubuh yang dibutuhkan untuk kerja otak, sistem saraf, dan jaringan tubuh yang lain. Gula darah yang terdapat di dalam tubuh dihasilkan oleh makanan yang mengandung karbohidrat, protein, dan lemak. Rata-rata, kadar gula darah normal adalah sebagai berikut:

- a. Gula darah 8 jam sebelum makan atau setelah bangun pagi (70-110 mg/dl).

- b. Gula darah 2 jam setelah makan (100-150 mg/dl).
- c. Gula darah acak (70-125 mg/dl).

3. Kadar Kolesterol Total

Kolesterol total merupakan kadar keseluruhan kolesterol yang beredar dalam tubuh manusia. Kolesterol adalah lipid amfipatik dan merupakan komponen struktural esensial pada membran plasma. Senyawa kolesterol total ini disintesis dibanyak jaringan dari asetil-KoA dan merupakan prekusor utama semua steroid lain di dalam tubuh termasuk kortikosteroid, hormone seks, asam empedu, dan vitamin D.

4. *Low Density Lipoprotein (LDL)*

Kolesterol LDL disebut sebagai kolesterol jahat disebabkan peranannya membawa kolesterol total ke banyak jaringan di dalam tubuh. Sehingga memberikan peluang terjadinya penumpukan kolesterol di berbagai jaringan tubuh, termasuk diantaranya dalam pembuluh darah.

5. Asam Urat

Penyakit asam urat adalah penyakit yang timbul akibat kadar asam urat darah yang berlebihan. Yang menyebabkan kadar asam urat darah berlebihan adalah produksi asam urat di dalam tubuh lebih banyak dari pembuangannya. Organ yang bisa terserang adalah sendi, otot, jaringan di sekitar sendi, telinga, kelopak mata, jantung, ginjal, dan lain-lain.

6. *Blood Urea Nitrogen (BUN)*

Blood Urea Nitrogen (BUN) dapat didefinisikan sebagai jumlah nitrogen urea yang hadir dalam darah. Urea adalah produk limbah yang dibentuk dalam tubuh selama proses pemecahan protein. Selama metabolisme protein, protein diubah menjadi asam amino yang juga menghasilkan amonia. Urea tidak lain adalah substansi yang dibentuk oleh beberapa molekul amonia. Metabolisme protein berlangsung dalam hati dan dengan demikian urea juga diproduksi oleh hati. Selanjutnya, urea ditransfer ke ginjal melalui aliran darah dan dikeluarkan dari tubuh dalam bentuk urin. Dengan demikian, setiap disfungsi ginjal akan menyebabkan kadar tinggi atau rendah BUN dalam darah.

7. Kreatinin (*creatinine*)

Kreatinin (*creatinine*) adalah produk penguraian dari kreatin fosfat dalam metabolisme otot dan dihasilkan dari kreatin (*creatine*). Kreatinin pada dasarnya merupakan limbah kimia yang selanjutnya diangkut ke ginjal melalui aliran darah untuk dikeluarkan melalui urin. Kadar kreatinin dapat diukur dalam urin serta darah. Tingkat kreatinin dalam darah umumnya tetap normal karena massa otot relatif konstan. Dengan demikian, ginjal yang berfungsi normal juga akan menunjukkan tingkat normal kreatinin dalam darah. Tapi ketika ginjal tidak berfungsi dengan baik, jumlah kreatinin dalam darah akan meningkat.

2.3 Pengendalian Stroke

Untuk menurunkan angka kesakitan, kecacatan dan kematian diperlukan pengendalian stroke. Kegiatan pengendalian stroke meliputi (Kemenkes, 2013):

1. Pelayanan pra stroke

Pelayanan pra stroke adalah kegiatan deteksi dini, penemuan dan monitoring faktor risiko stroke pada individu sehat dan berisiko di masyarakat. Pelayanan pra stroke dilakukan di:

- a. Puskesmas
- b. Klinik kesehatan
- c. Posbindu PTM

2. Pelayanan serangan stroke

Pelayanan serangan stroke dilakukan di:

- a. Rumah sakit dipusatkan pada unit stroke atau pojok stroke
- b. Rumah sakit khusus

3. Pelayanan paska stroke

Pelayanan paska stroke dilakukan di:

- a. Rumah sakit
- b. Puskesmas
- c. Posbindu PTM

2.4 Sistem Pendukung Keputusan

Sistem Pendukung Keputusan (SPK) adalah bagian dari sistem informasi berbasis komputer yang dipakai untuk mendukung pengambilan keputusan dalam suatu organisasi atau perusahaan. Suatu sistem pendukung keputusan memiliki beberapa subsistem yang menentukan kapabilitas teknis sistem pendukung keputusan (Turban, 2005; Iqbal, M. 2013) antara lain:

1. Manajemen data yaitu termasuk database, yang mengandung data yang relevan untuk berbagai situasi dan diatur oleh software yang disebut Database Management Systems (DBMS).
 2. Manajemen model yaitu melibatkan model finansial, statistika, manajemen pengetahuan, atau berbagai model kuantitatif lainnya, sehingga dapat memberikan ke sistem suatu kemampuan analitis, dan manajemen software yang diperlukan.
 3. Interaksi yaitu pengetahuan pekerja dapat berinteraksi pada sistem pendukung keputusan untuk melakukan analisis.
 4. Manajemen pengetahuan yaitu model Manajemen Pengetahuan juga berinterkoneksi dengan sistem integrasi manajemen pengetahuan perusahaan.

2.5 Normalisasi

Pada atribut data cenderung memiliki nilai dengan rentang yang sangat bervariasi dan jenis data masukan berupa data *string*. Misalnya nilai jangkaun data yang besar akan memiliki pengaruh besar dalam menentukan jarak perhitungan dibandingkan nilai jangkauan data yang kecil. Untuk itu dilakukan proses normalisasi data dengan menggunakan persamaan 2.1

$$\text{Rumus Normalisasi Data} = \frac{(\text{Nilai} - \text{Nilai min})}{(\text{Nilai max} - \text{Nilai min})} \quad \dots \dots \dots \quad (2.1)$$

2.6 Learning Vector Quantization (LVQ)

Learning Vector Quantization (LVQ) merupakan salah satu jenis jaringan saraf tiruan yang berbasis *Competitive Learning* atau *Winner Take*

All, dimana dari nilai keluaran yang diberikan neuron dalam layer keluaran hanya neuron pemenang (neuron yang mempunyai nilai terkecil) saja yang diperhatikan. Neuron pemenang tersebut yang akan mengalami pembaharuan bobot. Pembaharuan bobot yang dilakukan pada neuron pemenang ini bisa menambah atau mengurangi (Prasetyo, E. 2014).

LVQ menggunakan konsep kuantisasi perbedaan antara vektor masukan dengan bobot yang dimiliki oleh setiap neuron. LVQ tidak seperti perceptron dimana dalam perceptron setiap kelas diwakili hanya oleh 1 neuron. Dalam LVQ satu kelas dapat diwakili oleh lebih dari 1 neuron. Itulah sebabnya LVQ tidak membutuhkan layer tersembunyi seperti pada *Multi-Layer Perceptron* (MLP). Operasi yang dilakukan antara vektor dengan bobot tidak menggunakan inner-product, melainkan menggunakan kuantisasi perbedaan euclidean kuadrat. Kuantisasi untuk vektor x dengan neuron ke- j seperti dinyatakan dalam persamaan 2.2 (Prasetyo, E. 2014).

Dimana: d_j = vektor pelatihan

r = jumlah fitur yang digunakan

x_i = nilai atribut yang digunakan

w_{ij} = bobot yang digunakan

w_{ij} = bobot yang digunakan

Sedangkan untuk pembaharuan bobot pada nuron pemenang menggunakan formula pada persamaan 2.3.

$$w_{ij}(\text{baru}) = \begin{cases} w_{ij}(\text{lama}) + \eta (x_i - w_{ij}(\text{lama})), \text{ jika } \omega_{wj} = \omega_x \\ w_{ij}(\text{lama}) - \eta (x_i - w_{ij}(\text{lama})), \text{ jika } \omega_{wj} \neq \omega_x \\ \text{tidak ada update} \end{cases}, \text{neuron lainnya} \quad \dots(2.3)$$

Dimana: w_{ij} = bobot

x_i = nilai atribut yang digunakan

$$\eta = \text{laju pembelajaran}$$

ω_{wj} = kelas target

$$\omega_x = \text{kelas neuron}$$

Untuk ω_{wj} menyatakan label kelas yang diwakili oleh neuron ke-j (neuron pemenang), sedangkan ω_x adalah label kelas yang sebenarnya diwakili oleh vektor x. Untuk neuron selain neuron pemenang tidak mengalami pembaharuan apa-apa (Prasetyo, E. 2014).

2.6.1 Proses Pembelajaran LVQ

Pada proses awal pengenalan, vektor input akan mengalami proses pembelajaran yang dilakukan melalui beberapa epoch sampai batas epoch maksimal tercapai. LVQ melakukan pembelajaran pada lapisan kompetitif yang terawasi. 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 input dengan vektor bobot dari masing-masing kelas dan vektor input akan masuk ke dalam kelas yang memiliki jarak terdekat. Algoritma pembelajaran pada LVQ bertujuan mencari nilai bobot yang sesuai untuk mengelompokkan vektor-vektor input ke dalam kelas yang sesuai dengan yang telah diinisialisasi pada saat pembentukan jaringan LVQ.

2.6.2 Algoritma Pelatihan LVQ

Berikut ini adalah langkah-langkah yang perlu dilakukan dalam menerapkan metode LVQ dalam pengolahan data (Prasetyo, E. 2014).

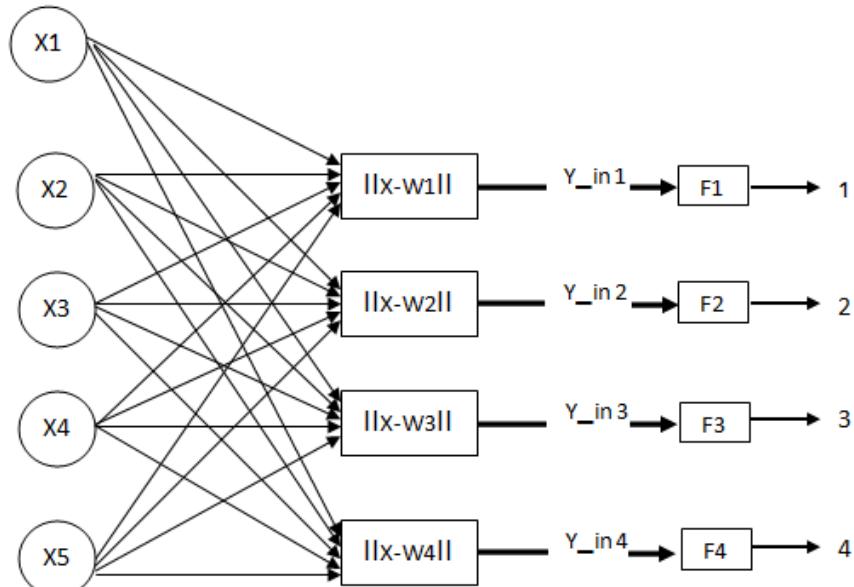
1. Inisialisasi, Tentukan jumlah neuron untuk mengkategorikan semua data latih. Tentukan bobot awal neuron dengan memberikan nilai tengah dari jangkauan nilai setiap fitur. Tentukan laju pembelajaran (η). Tentukan fungsi pembelajaran. Tentukan jumlah iterasi.
2. Lakukan langkah 3 sampai 7 hingga mencapai jumlah iterasi atau nilai laju pembelajaran sudah menjadi sangat kecil.
3. Untuk setiap vector latih masukan, lakukan langkah 4 sampai 5.

4. Hitung kuantisasi vector (d) pada semua neuron menggunakan persamaan 2.2, kemudian pilih yang paling kecil. Neuron dengan kuantisasi paling kecil menjadi pemenang.
5. Perbaharui bobot neuron pemenang menggunakan persamaan 2.3.
6. Perbaharui laju pembelajaran dengan fungsi pembelajaran.
7. Tes kondisi berhenti Jika: (alfa (α) < Mina atau iterasi > Max iterasi)
Maka berhenti, Jika tidak : iterasi = iterasi + 1, ulangi langkah ke 4.

2.6.3 Arsitektur LVQ

Arsitektur LVQ terdiri dari lapisan input (*input layer*), lapisan kompetitif (terjadi kompetisi pada input untuk masuk ke dalam suatu kelas berdasarkan kedekatan jaraknya) dan lapisan output (*output layer*). Lapisan input dihubungkan dengan lapisan kompetitif oleh bobot. Dalam lapisan kompetitif, proses pembelajaran dilakukan secara terawasi. Input akan bersaing untuk dapat masuk ke dalam suatu kelas. Hasil dari lapisan kompetitif ini berupa kelas, yang kemudian akan dihubungkan dengan lapisan output oleh fungsi aktivasi. Fungsi aktivasi yang digunakan adalah fungsi linear dengan tujuan kelas yang diperoleh pada lapisan output sesuai dengan kelas yang dimasukkan ke lapisan output.

Misalkan ada lima variable dari vektor input pada contoh jaringan LVQ berikut, $X = (X_1, X_2, X_3, X_4, X_5)$ dengan neuron keluaran Y_1, Y_2, Y_3 , dan Y_4 dengan empat vektor bobot yaitu $W_{1j} = (W_{11}, W_{12}, W_{13}, W_{14}, W_{15}, W_{16})$, $W_{2j} = (W_{21}, W_{22}, W_{23}, W_{24}, W_{25}, W_{26})$, $W_{3j} = (W_{31}, W_{32}, W_{33}, W_{34}, W_{35}, W_{36})$, $W_{4j} = (W_{41}, W_{42}, W_{43}, W_{44}, W_{45}, W_{46})$ maka arsitektur jaringan LVQ dapat dilihat pada Gambar 2.1.



Gambar 2.1 Arsitektur LVQ

Keterangan:

X = vektor masukan (X_1, \dots, X_n)

W = vektor bobot atau vektor pewakil

$|X-W|$ = selisih nilai jarak *Euclidian* antara vektor *input* dengan vektor bobot

F = lapisan kompetitif (fungsi aktivasi)

Y = keluaran (*output*)

Ketika hasil pemrosesan jaringan memberikan hasil klasifikasi yang sama dengan informasi kelas yang diberikan di awal, maka vektor pewakil akan disesuaikan agar lebih dekat dengan vektor masukan. Sebaliknya ketika hasil klasifikasi tidak sama dengan informasi kelas yang diberikan di awal, maka vektor pewakil akan disesuaikan agar menjauhi vektor masukan. Dalam pengenalan metode LVQ terdapat dua proses, yaitu proses pembelajaran dan proses pengujian.

2.6.4 Contoh ANN Learning Vector Quantization pada operasi logika XOR

Pada contoh ini akan dilakukan pengelompokan lima buah data dengan jumlah dimensi dua. Set data yang akan digunakan adalah data pada tabel 2.1 (Prasetyo, E. 2014) :

Tabel 2.1 Set data pada contoh metode LVQ.

No	X	Y	Kelas
1	1	1	0
2	1	0	1
3	0	1	1
4	0	0	0

Fitur yang digunakan adalah X dan Y. Pada contoh ini akan dilakukan proses pengelompokan dengan metode LVQ. Parameter yang digunakan untuk nilai laju pembelajaran (η) yang digunakan adalah 0.5, fungsi pembelajaran adalah 0.6 dari η . jumlah iterasi 6 kali, sedangkan jumlah neuron keluaran yang digunakan adalah 3, dengan rincian, neuron 1 dan 2 mewakili kelas 1, sedangkan neuron 3 mewakili kelas 0. Inisialisasi bobot awal:

$$w = \begin{bmatrix} 0.5 & 0.5 & 0.5 \\ 0.5 & 0.5 & 0.5 \end{bmatrix}$$

Iterasi 1 Hitung kuantitasi vektor 1 [1 1] ke setiap neuron:

$$d_1 = \sum_{i=1}^r (x_i - w_{i1})^2 = (x_1 - w_{11})^2 + (x_2 - w_{21})^2 \\ = (1 - 0.5)^2 + (1 - 0.5)^2 = 0.5^2 + 0.5^2 = 0.25 + 0.25 = 0.5$$

$$d_2 = \sum_{i=1}^r (x_i - w_{i2})^2 = (x_1 - w_{12})^2 + (x_2 - w_{22})^2 \\ = (1 - 0.5)^2 + (1 - 0.5)^2 = 0.5^2 + 0.5^2 = 0.25 + 0.25 = 0.5$$

$$d_3 = \sum_{i=1}^r (x_i - w_{i3})^2 = (x_1 - w_{13})^2 + (x_2 - w_{23})^2 \\ = (1 - 0.5)^2 + (1 - 0.5)^2 = 0.5^2 + 0.5^2 = 0.25 + 0.25 = 0.5$$

Karena ketiga neuron memberikan keluaran yang sama, maka pemenangnya bisa dipilih sembarang. Misalkan dipilih neuron 1 mewakili kelas 1, sedangkan vektor 1 mempunyai label kelas 0 (label kelas tidak sama), maka neuron 1 akan mengalami pembaharuan bobot:

$$W_{i1}(\text{baru}) = W_{i1}(\text{lama}) - \eta((X_i - W_{i1}(\text{lama})))$$

$$= \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix} - 0.5 \left(\begin{bmatrix} 1 \\ 1 \end{bmatrix} - \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix} \right) = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix} - \begin{bmatrix} 0.25 \\ 0.25 \end{bmatrix} = \begin{bmatrix} 0.25 \\ 0.25 \end{bmatrix}$$

$$\text{Bobot menjadi: } W = \begin{bmatrix} 0.25 & 0.5 & 0.5 \\ 0.25 & 0.5 & 0.5 \end{bmatrix}$$

Untuk vektor 2 [1 0], hitung kuantisasi ke setiap neuron:

$$\begin{aligned} d_1 &= \sum_{i=1}^r (x_i - w_{i1})^2 = (x_1 - w_{11})^2 + (x_2 - w_{21})^2 \\ &= (1 - 0.25)^2 + (0 - 0.25)^2 = 0.75^2 + (-0.25)^2 = 0.5625 + 0.0625 = 0.625 \end{aligned}$$

$$\begin{aligned} d_2 &= \sum_{i=1}^r (x_i - w_{i2})^2 = (x_1 - w_{12})^2 + (x_2 - w_{22})^2 \\ &= (1 - 0.5)^2 + (0 - 0.5)^2 = 0.5^2 + 0.5^2 = 0.25 + 0.25 = 0.5 \end{aligned}$$

$$\begin{aligned} d_3 &= \sum_{i=1}^r (x_i - w_{i3})^2 = (x_1 - w_{13})^2 + (x_2 - w_{23})^2 \\ &= (1 - 0.5)^2 + (0 - 0.5)^2 = 0.5^2 + 0.5^2 = 0.25 + 0.25 = 0.5 \end{aligned}$$

Neuron pemenang ada 2 yaitu 2 dan 3, di pilih salah satu sebagai pemenang, misalnya neuron 2. Neuron 2 mewakili kelas 1, sedangkan vektor 2 mempunyai label 1 (label kelas sama). Maka neuron 2 akan mengalami pembaharuan bobot:

$$W_{i2}(\text{baru}) = W_{i2}(\text{lama}) - \eta((X_i - W_{i2}(\text{lama})))$$

$$= \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix} + 0.5 \left(\begin{bmatrix} 1 \\ 0 \end{bmatrix} - \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix} \right) = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix} + \begin{bmatrix} 0.25 \\ -0.25 \end{bmatrix} = \begin{bmatrix} 0.75 \\ 0.25 \end{bmatrix}$$

$$\text{Bobot menjadi: } W = \begin{bmatrix} 0.25 & 0.75 & 0.5 \\ 0.25 & 0.25 & 0.5 \end{bmatrix}$$

Untuk vektor 3 [0 1], hitung kuantisasi ke setiap neuron:

$$\begin{aligned} d_1 &= \sum_{i=1}^r (x_i - w_{i1})^2 = (x_1 - w_{11})^2 + (x_2 - w_{21})^2 \\ &= (0 - 0.25)^2 + (1 - 0.25)^2 = (-0.25)^2 + 0.75^2 = 0.0625 + 0.5625 = \\ &0.625 \end{aligned}$$

$$\begin{aligned} d_2 &= \sum_{i=1}^r (x_i - w_{i2})^2 = (x_1 - w_{12})^2 + (x_2 - w_{22})^2 \\ &= (0 - 0.75)^2 + (1 - 0.25)^2 = -0.75^2 + 0.75^2 = 0.5625 + 0.5625 = \\ &1.125 \end{aligned}$$

$$\begin{aligned} d_3 &= \sum_{i=1}^r (x_i - w_{i3})^2 = (x_1 - w_{13})^2 + (x_2 - w_{23})^2 \\ &= (0 - 0.5)^2 + (1 - 0.5)^2 = (-0.5)^2 + 0.5^2 = 0.25 + 0.25 = 0.5 \end{aligned}$$

Neuron pemenang adalah neuron 3. Neuron 3 mewakili kelas 0, sedangkan vektor 3 mempunyai label kelas 1 (label kelas tidak sama), maka neuron 3 akan mengalami pembaharuan bobot:

$$W_{i3}(\text{baru}) = W_{i3}(\text{lama}) - \eta ((X_i - W_{i3}(\text{lama})))$$

$$= \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix} - 0.5 \left(\begin{bmatrix} 0 \\ 1 \end{bmatrix} - \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix} \right) = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix} - \begin{bmatrix} -0.25 \\ 0.25 \end{bmatrix} = \begin{bmatrix} 0.75 \\ 0.25 \end{bmatrix}$$

$$\text{Bobot menjadi: } W = \begin{bmatrix} 0.25 & 0.75 & 0.75 \\ 0.25 & 0.25 & 0.25 \end{bmatrix}$$

Untuk vektor 4 [0 0], hitung kuantisasi ke setiap neuron:

$$\begin{aligned}
 d_1 &= \sum_{i=1}^r (x_i - w_{i1})^2 = (x_1 - w_{11})^2 + (x_2 - w_{21})^2 \\
 &= (0 - 0.25)^2 + (0 - 0.25)^2 = (-0.25)^2 + (-0.25)^2 = 0.0625 + 0.0625 = \\
 &\quad 0.125
 \end{aligned}$$

$$\begin{aligned}
 d_2 &= \sum_{i=1}^r (x_i - w_{i2})^2 = (x_1 - w_{12})^2 + (x_2 - w_{22})^2 \\
 &= (0 - 0.75)^2 + (0 - 0.25)^2 = (-0.75)^2 + (-0.25)^2 = 0.5625 + 0.0625 = \\
 &\quad 0.625
 \end{aligned}$$

$$\begin{aligned}
 d_3 &= \sum_{i=1}^r (x_i - w_{i3})^2 = (x_1 - w_{13})^2 + (x_2 - w_{23})^2 \\
 &= (0 - 0.75)^2 + (0 - 0.25)^2 = (-0.75)^2 + (-0.25)^2 = 0.5625 + 0.0625 = \\
 &\quad 0.625
 \end{aligned}$$

Neuron pemenang adalah neuron 1. Neuron 1 mewakili kelas 1, sedangkan vektor 4 mempunyai label 0 (label kelas tidak sama), maka neuron 1 akan mengalami pembaharuan bobot:

$$\begin{aligned}
 W_{i1}(\text{baru}) &= W_{i1}(\text{lama}) - \eta ((X_i - W_{i1}(\text{lama})) \\
 &= \begin{bmatrix} 0.25 \\ 0.25 \end{bmatrix} + 0.5 \left(\begin{bmatrix} 0 \\ 0 \end{bmatrix} - \begin{bmatrix} 0.25 \\ 0.25 \end{bmatrix} \right) = \begin{bmatrix} 0.25 \\ 0.25 \end{bmatrix} + \begin{bmatrix} -0.125 \\ -0.125 \end{bmatrix} = \begin{bmatrix} 0.375 \\ 0.375 \end{bmatrix}
 \end{aligned}$$

$$\text{Bobot menjadi: } W = \begin{bmatrix} 0.375 & 0.75 & 0.75 \\ 0.375 & 0.25 & 0.25 \end{bmatrix}$$

Nilai keluaran yang di dapat dari semua vektor adalah = [1 1 0 1]

Iterasi 1 sudah berakhir. Laju pembelajaran di perbaharui:

$$\eta(1) = 0.6 \times \eta(0) = 0.6 \times 0.5 = 0.3$$

Iterasi 2

Dengan cara yang sama, bobot akhir iterasi 2 menjadi:

$$W = \begin{bmatrix} 0.3413 & 0.7725 & 0.75 \\ 0.7312 & 0.0175 & 0.25 \end{bmatrix}$$

Nilai keluaran yang di dapat dari semua vektor adalah = [1 1 1]

Laju pembelajaran di perbaharui:

$$\eta(2) = 0.6 \times \eta(1) = 0.6 \times 0.3 = 0.18$$

Iterasi 3

Dengan cara yang sama, bobot akhir iterasi 3 menjadi:

$$W = \begin{bmatrix} 0.2155 & 0.8135 & 0.75 \\ 0.8731 & 0.0144 & 0.25 \end{bmatrix}$$

Nilai keluaran yang di dapat dari semua vektor adalah = [1 1 1]

Laju pembelajaran di perbaharui:

$$\eta(3) = 0.6 \times \eta(2) = 0.6 \times 0.18 = 0.108$$

Iterasi 4

Dengan cara yang sama, bobot akhir iterasi 4 menjadi:

$$W = \begin{bmatrix} 0.1922 & 0.9236 & 0.7770 \\ 0.8868 & 0.0142 & 0.3310 \end{bmatrix}$$

Nilai keluaran yang di dapat dari semua vektor adalah = [0 1 1].

Laju pembelajaran di perbaharui:

$$\eta(4) = 0.6 \times \eta(3) = 0.6 \times 0.108 = 0.0648.$$

Iterasi 5

Dengan cara yang sama, bobot akhir iterasi 5 menjadi:

$$W = \begin{bmatrix} 0.1797 & 0.9286 & 0.7402 \\ 0.8942 & 0.0133 & 0.3501 \end{bmatrix}$$

Nilai keluaran yang di dapat dari semua vektor adalah = [0 1 1 0].

Laju pembelajaran di perbaharui:

$$\eta(5) = 0.6 \times \eta(4) = 0.6 \times 0.0648 = 0.0389$$

Iterasi 6

Dengan cara yang sama, bobot akhir iterasi 6 menjadi:

$$W = \begin{bmatrix} 0.1727 & 0.9314 & 0.7411 \\ 0.8983 & 0.0127 & 0.3608 \end{bmatrix}$$

Nilai keluaran yang di dapat dari semua vektor adalah = [0 1 1 0]

Laju pembelajaran di perbaharui:

$$\eta(6) = 0.6 \times \eta(5) = 0.6 \times 0.0389 = 0.0233 .$$

Setelah iterasi 6 kali, didapat bobot akhir adalah

$$W = \begin{bmatrix} 0.1727 & 0.9314 & 0.7411 \\ 0.8983 & 0.0127 & 0.3608 \end{bmatrix}$$

Sedangkan semua vektor sudah terpetakan dengan benar ke kelasnya masing-masing artinya model ANN sudah selesai dan siap digunakan untuk prediksi.

2.7 Evaluasi

Menurut Alfiyanto et al (2014) tujuan evaluasi percobaan pada klasifikasi yaitu untuk mengukur keefektifan apakah sistem mengklasifikasi secara benar. Evaluasi biasanya membutuhkan sebuah matriks yang disebut berupa matriks *confusion*. Matriks *counfusion* ditunjukkan pada tabel 2.2.

Tabel 2.2 Matriks confusion

Ck	<i>Classifier positive label</i>	<i>Classifier negative label</i>
<i>True positive label</i>	A	B
<i>True negative label</i>	C	D

Dari tabel 2.2 menunjukkan bahwa jika diberikan kategori Ck, parameter A adalah jumlah dokumen yang berhasil dikategorikan oleh sistem ke dalam kategori Ck, parameter B adalah jumlah dokumen yang mempunyai kategori Ck namun sistem tidak mengklasifikasikannya ke dalam kategori Ck, parameter C adalah jumlah dokumen yang bukan kategori Ck namun sistem mengklasifikasikannya ke dalam kategori Ck, dan parameter D adalah jumlah dokumen yang tidak termasuk kategori Ck dan sistem juga tidak mengklasifikasikannya ke dalam kategori Ck.

Recall adalah ukuran keberhasilan sistem dalam mengenali dokumen pada setiap kategori tanpa melihat ketepatan klasifikasi yang dilakukan. *Recall* dihitung menggunakan persamaan 2.4.

$$Recall = \frac{A}{A + B} \quad \dots \dots \dots (2.4)$$

Precision adalah ukuran keberhasilan sistem dalam melakukan ketepatan klasifikasi tanpa melihat seberapa banyak dokumen yang berhasil dikenali. *Precision* dihitung menggunakan persamaan 2.5.

$$Precision = \frac{A}{A + C} \quad \dots \dots \dots (2.5)$$

F-measure mewakili pengaruh relatif antara *precision* dan *recall*, yang dihitung dengan persamaan berikut. *F-measure* dihitung menggunakan persamaan 2.6.

$$Fmeasure = \frac{2 \cdot recall \cdot precision}{(recall + precision)} \quad \dots \dots \dots (2.6)$$

2.8 Penelitian Sebelumnya

Penelitian yang dilakukan oleh Aji Seto Arifianto, Moechammad Sarosa, Onny Setyawati (2014) dengan judul Klasifikasi Stroke Berdasarkan

Kelainan Patologis dengan *Learning Vector Quantization*, mereka membuat sebuah sistem untuk klasifikasi stroke dalam 2 tipe yaitu: *ischemic stroke* disebut juga *infark* atau *nonhemorrhagic* disebabkan oleh gumpalan atau penyumbatan dalam arteri yang menuju ke otak yang sebelumnya sudah mengalami proses aterosklerosis. Tipe kedua adalah *hemorrhagic stroke* merupakan kerusakan atau "ledakan" dari pembuluh darah di otak, perdarahan dapat disebabkan lamanya tekanan darah tinggi dan aneurisma otak. Data yang digunakan untuk pelatihan metode *Learning Vector Quantization* sebanyak 323 data sedangkan untuk ujicoba 50 data. Sistem yang dibangun berhasil mengklasifikasi stroke berdasarkan kelainan patologis dengan tingkat akurasi 96% dengan nilai *Area Under Curve* 0.952 termasuk dalam katagori *excellent/A*.

Penelitian selanjutnya dilakukan oleh Dhimas Arief Dharmawan (2014) dengan judul Deteksi Kanker Serviks Otomatis Berbasis Jaringan Saraf Tiruan LVQ dan DCT. Pada penelitian ini dirancang perangkat lunak dengan jaringan saraf tiruan Learning Vector Quantizatin (LVQ), sebagai alat bantu deteksi kanker serviks. Sebelum dideteksi, dilakukan pengolahan citra terhadap citra sel serviks, yaitu preprocessing, peregangan kontras, median filter, operasi morfologi, segmentasi, dan ekstraksi fitur dengan Discrete Cosine Transform (DCT). Citra sel serviks yang digunakan berjumlah 73 buah yang terdiri atas lima puluh buah citra sel normal dan 23 buah citra sel kanker. Proses pelatihan LVQ menggunakan 35 buah citra sel normal dan empat belas buah citra sel kanker. Proses pengujian LVQ menggunakan 15 buah citra sel normal dan sembilan buah citra sel kanker. Dari hasil pengujian, didapatkan nilai sensitivitas, spesifisitas, dan akurasi sebesar 88,89%, 100%, dan 95,83%.

Penelitian selanjutnya dilakukan oleh Nurul Hidayati, Budi Warsito (2010) dengan judul Prediksi Terjangkitnya Penyakit Jantung Dengan Metode Learning Vector Quantization. Pada penelitian ini dibahas prediksi penyakit jantung menggunakan LVQ. Menurut berbagai sumber terdapat 10 faktor yang mempengaruhi seseorang berpotensi terjangkit penyakit jantung maka

dari itu pada penelitian ini, menggunakan 10 sample yang memengaruhi seseorang berpotensi terjangkit penyakit jantung berikut ini adalah 10 sample yang digunakan dalam penelitian ini meliputi data, usia, jenis kelamin, angina, tekanan darah saat beristirahat, kolesterol, kecepatan gula darah, denyut nadi maksimal, merokok, keturunan dan olahraga. Jumlah data seluruhnya adalah 270. Parameter yang digunakan dalam penelitian ini adalah: alfa (learning rate) 0.25, deca 0.1, mina 0.001, maxepoch 100. setelah dilakukan training sampai 100 epoch tingkat keberhasilan jaringan LVQ untuk dapat mengenali pola dengan benar sebesar 66.79%.

Penelitian selanjutnya dilakukan oleh Hariyono Hanafi (2015) dengan judul Penentuan Klasifikasi Status Gizi Orang Dewasa Dengan Menggunakan Metode ANN *Learning Vector Quantization* (LVQ). Penelitian yang dilakukan untuk mengklasifikasikan status gizi orang dewasa dengan menggunakan data antropometri meliputi data usia, berat badan, tinggi badan, lingkar pinggang dan lingkar pinggul. Nilai parameter yang digunakan pada algoritma LVQ ini meliputi nilai *learning rate* (α) = 0.3, nilai minimal *learning rate* (Mina) = 0.001, nilai pengurangan α adalah 0.6, dan menggunakan 8 neuron dalam melakukan klasifikasi status gizi orang dewasa dengan nilai akurasi mencapai 86%