

## KOMPARASI HASIL KLASIFIKASI PENYAKIT DIABETES MELLITUS MENGUNAKAN JARINGAN SYARAF TIRUAN *BACKPROPAGATION* DAN *LEARNING VECTOR QUANTIZATION*

Agus Nurkhozin<sup>1</sup>, Mohammad Isa Irawan<sup>2</sup>, Imam Mukhlas<sup>2</sup>

<sup>1</sup>Mahasiswa Pascasarjana Matematika FMIPA ITS Surabaya

<sup>2</sup>Dosen Pascasarjana Jurusan FMIPA ITS Surabaya

Email: akhozin73@gmail.com

### Abstrak

Diabetes Mellitus merupakan penyakit yang ditandai dengan kenaikan kadar gula dalam darah. Penyebab penyakit ini biasanya disebabkan destruksi sel beta, kebiasaan pola makan, dan olah raga yang kurang teratur. Menurut asuhan keperawatan diabetes mellitus dibedakan dalam empat klasifikasi, tetapi dalam penelitian ini data penyakit diabetes mellitus diklasifikasikan menurut dr. Eko Yulianto diklasifikasikan dalam dua kelas. Jaringan syaraf tiruan merupakan sistem pemroses informasi seperti pemroses pada otak manusia. Jaringan syaraf tiruan telah banyak digunakan dalam banyak aplikasi, salah satunya untuk klasifikasi. Dalam penelitian ini menampilkan dua metode klasifikasi yaitu metode *Backpropagation* dan *Learning Vector Quantization* untuk permasalahan klasifikasi penyakit diabetes mellitus. Kesimpulan yang didapat bahwa dalam klasifikasi penyakit diabetes mellitus metode *Backpropagation* memberikan kinerja lebih baik dibandingkan LVQ

**Kata kunci:** Jaringan Syaraf Tiruan, *Backpropagation*, *Learning Vector Quantization*, *Diabetes Mellitus*

### PENDAHULUAN

Diabetes mellitus merupakan sua-tu keadaan hiperglikemik kronis dan perlahan namun pasti akan merusak jaringan dalam tubuh jika tidak ditangani secara tepat dan serius. Penyakit ini akan mengakibatkan berbagai macam komplikasi kronis terjadi pada mata, ginjal, saraf, dan pembuluh darah. Indonesia menempati urutan ke-4 dengan jumlah penderita diabetes mellitus terbanyak di dunia setelah India, Cina, dan Amerika Serikat.

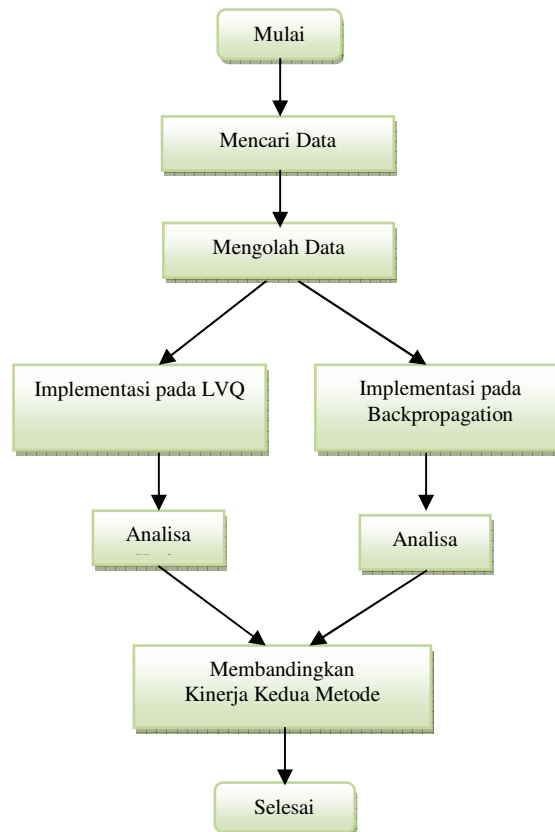
Dalam menyelesaikan Masalah yang kompleks memerlukan metode cepat, tepat dan akurat. Salah satunya adalah Jaringan Syaraf Tiruan. Jaringan ini telah menjadi obyek penelitian yang menarik dan banyak digunakan untuk menyelesaikan permasalahan pada beberapa bidang kehidupan, contohnya klasifikasi. Dalam penelitian ini menggunakan metode jaringan syaraf tiruan *backpropagation* dan *Learning Vector Quantization (LVQ)* dalam kaitannya kasifikasi penyakit diabetes, karena metode ini terkenal bagus, cepat, dan akurat yang mampu melakukan pengelompokan data berdasarkan sifat atribut yang dimiliki oleh sekelompok besar data. Penelitian ini bertujuan untuk mengkomparasi hasil klasifikasi menggunakan Jaringan syaraf tiruan *Backpropagation* dan LVQ dalam meng-klasifikasikan penyakit diabetes mellitus ?

### METODE PENELITIAN

Prosedur kerja pada penelitian ini dilakukan dalam beberapa tahapan penelitian. Tahap 1 adalah mengidentifikasi permasalahan dengan disertai pencarian topik yang akan diuji, mencari literature yang sesuai dengan permasalahan penelitian, dan proses pembuatan proposal penelitian tesis. Tahap 2 adalah Mengambil data di mana data yang diambil adalah data pasien pengidap penyakit diabetes mellitus. Data diambil dari RSUD Lamongan Jalan Kusuma Bangsa No. 7 Lamongan. Data yang didapat adalah data pasien dari Bulan Januari sampai Desember 2010 yang tercatat sebanyak 1.386 pasien, dan diambil sampel sebanyak 431 pasien. Selanjutnya data yang

diperoleh didiskusikan dengan dokter spesialis penyakit dalam yaitu dr. Eko Yulianto, Spa. Di mana data tersebut diklasifikasikan dalam dua kelas. Tahap 3 mengimplementasikan pada metode klasifikasi yang didahului oleh pengolahan data sesuai kebutuhan jaringan, yang dijadikan inputan pada saat ditraining dan ditesting. Metode klasifikasi yang penulis gunakan adalah metode *Backpropagation* dan *Learning Vector Quantization (LVQ)* dengan algoritma yang berbeda. Setelah data ditraining dan ditesting dengan menggunakan metode *Backpropagation* dan LVQ hasilnya dianalisis keakurasiannya. Selanjutnya ditarik kesimpulan metode mana dari kedua metode tersebut yang lebih baik.

Diagram Proses Penelitian



Gambar 1 : Denah alur penelitian

## PEMBAHASAN

Data yang diperoleh kemudian peneliti bagi menjadi enam variable input dan target sebanyak dua kelas. Variabel input dinormalisasikan dan diubah menjadi bentuk biner sesuai kebutuhan jaringan. Menurut dr. Eko Yulianto, Spa data dijabarkan sebagai berikut :

1. Kelas 1. Usia muda termasuk anak-anak (<40 tahun), badan kurus, terjadi destruksi sel beta, rata-rata tidak ada riwayat
2. Kelas 2. Usia Dewasa dan Usia Lanjut ( $\geq 40$  tahun), badan kurus, gemuk dan obesitas, ada riwayat dan terjadi diabetagonik

Sebagai Variabel input :

$x_1$  = Usia  $x_2$  = Berat badan  $x_3$  = Tinggi badan  $x_4$  = Riwayat keluarga

$x_5$  = Terdapat gangguan destruksi sel beta  $x_6$  = Pengaruh pola makan

Sedangkan sebagai unit output atau target adalah :

$t_1$  = Kelas 1  $t_2$  = Kelas 2

Untuk  $x_4$  sampai  $x_6$  data sudah berbentuk biner, sedangkan untuk  $x_1$  sampai  $x_3$  data dinormalisasi dengan rumus :

$$x_i = \frac{x_{data} - x_{min}}{x_{max} - x_{min}}$$

kemudian data diubah menjadi bentuk biner [0,1] dengan ketentuan sebagai berikut :

$$x_i(\text{biner}) = \begin{cases} 0 & \text{jika data } x_i \text{ bernilai } \leq 0,5 \\ 1 & \text{jika data } x_i \text{ bernilai } > 0,5 \end{cases}, \text{ untuk } i = 1, 2, 3$$

Untuk data output atau target :

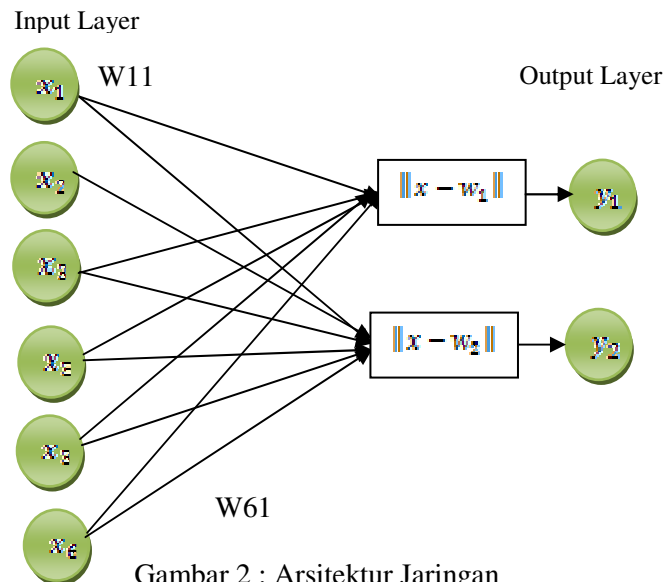
$$t_1 = 1 \text{ dan } t_2 = 2$$

## Implementasi pada Jaringan

### a. Jaringan LVQ

LVQ adalah jaringan *single layer* yang terdiri dari dua lapisan yaitu lapisan input dan output. Lapisan input terdiri dari enam unit input dan unit output terdiri dari dua unit output yakni diambil dari banyaknya kelas klasifikasi.

Arsitektur jaringan LVQ :



Gambar 2 : Arsitektur Jaringan LVQ Permasalahan Diabetes Mellitus

Deskripsi dari gambar di atas adalah sebagai berikut :

$x$  adalah vektor training sebanyak  $(x_1, x_2, \dots, x_6)$

$T$  adalah target untuk vektor training sebanyak 2 target yaitu  $t_1, t_2$  yang ditentukan dalam dua kelas.

$w_j$  adalah vektor bobot untuk unit output ke-j yaitu  $(w_{1j}, w_{2j}, \dots, w_{6j})$

$C_j$  adalah kelas hasil komputasi oleh unit output ke-j, terdiri dari dua kelas yaitu  $t_1, t_2$

$\|x - w_j\|$  adalah jarak Euclidean antara vektor input dengan vektor bobot untuk unit output ke-j.

Algoritma LVQ :

Langkah 0 inialisasi vektor referensi, inialisasi learning rate  $\alpha(0)$

Langkah 1

Bila kondisi STOP belum terpenuhi, kerjakan langkah 2-6

Langkah 2

Untuk setiap vector input training  $x$ , kerjakan langkah 3- 4

Langkah 3

Dapatkan nilai  $J$  sedemikian hingga  $\|x - w_j\|$  minimum

Langkah 4

Update  $w_j$  sebagai berikut :

Jika  $C_j = T$  maka

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

Jika  $C_j \neq T$  maka

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

Langkah 5

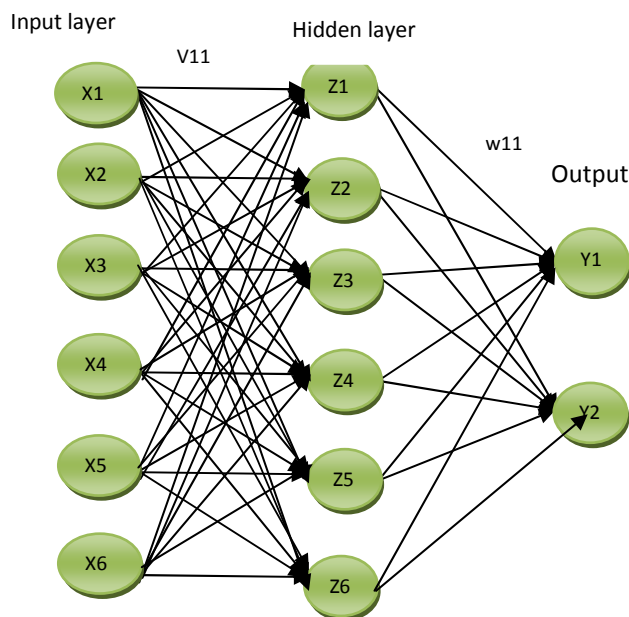
Reduksi learning rate

Langkah 6

Tes kondisi STOP

#### b. Jaringan Backpropagation

Backpropagation adalah jaringan multilayer, yang terdiri dari 3 lapisan yaitu lapisan input, hidden dan output. Lapisan input terdiri dari enam unit. Lapisan hidden terdiri dari 6 yang dipilih dari 10 node yang dipilih berdasarkan dari training dengan error terkecil. Sedangkan unit output terdiri dari dua kelas. Arsitektur jaringan Backpropagation :



Gambar 3 : Arsitektur backpropagation permasalahan Diabetes Mellitus

#### Algoritma backpropagation

Dengan menggunakan satu *hidden layer*, algoritma *backpropagation* sebagai berikut :

*Step 0 :*

Inialisasi nilai bobot dengan nilai acak yang kecil.

*Step 1 :*

Selama kondisi berhenti masih tidak terpenuhi, laksanakan *step 2 - 9*.

*Step 2 :* Untuk tiap pasangan pelatihan, kerjakan *step 3 - 8*.

***Feedforward :***

Step 3 :

Untuk tiap unit *input* ( $X_i, i=1, \dots, n$ ) menerima sinyal *input*  $x_i$  dan menyebarkan sinyal itu ke seluruh unit pada *layer* berikutnya (*hidden layer*)

Step 4:

Untuk tiap unit dalam ( $Z_j, j=1, \dots, p$ ) dihitung nilai *input* dengan menggunakan nilai bobotnya

$$z\_in_j = v_{0j} + \sum_{i=1}^n x_i v_{ij}$$

Kemudian dihitung nilai *output* dengan menggunakan fungsi aktivasi yang dipilih

$$z_j = f(z\_in_j)$$

Hasil fungsi tersebut dikirim ke semua unit pada *layer* berikutnya (unit *output*).

Step 5:

Untuk tiap unit *output* ( $Y_k, k=1, \dots, m$ ) dihitung nilai *input* dengan menggunakan nilai bobotnya

$$y\_in_k = w_{0k} + \sum_{j=1}^p z_j w_{jk}$$

Kemudian dihitung nilai *output* dengan menggunakan fungsi aktivasi

$$y_k = f(y\_in_k)$$

### **Backpropagation Of Error**

Step 6 :

Untuk tiap unit *output* ( $Y_k, k=1, \dots, m$ ) menerima pola target yang bersesuaian dengan pola *input* dan kemudian dihitung informasi kesalahan

$$\delta_k = (t_k - y_k) f'(y\_in_k)$$

Kemudian dihitung koreksi nilai bobot yang akan digunakan untuk memperbaharui nilai bobot  $w_{jk}$  :

$$\Delta w_{jk} = \alpha \delta_k z_j$$

Hitung koreksi nilai bias yang kemudian akan digunakan untuk memperbaharui nilai  $w_{0k}$  :

$$\Delta w_{0k} = \alpha \delta_k$$

dan kemudian nilai  $\delta_k$  dikirim ke unit pada *layer* sebelumnya.

Step 7:

Untuk tiap unit dalam ( $Z_j, j=1, \dots, p$ ) dihitung delta *input* yang berasal dari unit pada *layer* di atasnya :

$$\delta\_in_j = \sum_{k=1}^m \delta_k w_{jk}$$

Kemudian nilai tersebut dikalikan dengan nilai turunan dari fungsi aktivasi untuk menghitung informasi kesalahan :  $\delta_j = \delta\_in_j f'(z\_in_j)$

Hitung koreksi nilai bobot yang kemudian digunakan untuk memperbaharui nilai  $v_{ij}$  :

$$\Delta v_{ij} = \alpha \delta_j x_i$$

dan hitung nilai koreksi bias yang kemudian digunakan untuk memperbaharui  $v_{0j}$  :

$$\Delta v_{0j} = \alpha \delta_j$$

### **Update nilai bobot dan bias :**

Step 8 :

Setiap unit *output* ( $Y_k, k=1, \dots, m$ ) mengupdate bias dan bobotnya ( $j=0, \dots, p$ )

$$w_{jk}(new) = w_{jk}(old) + \Delta w_{jk}$$

Tiap unit *hidden* ( $Z_j, j=1, \dots, p$ ) mengupdate bias dan bobotnya ( $i=0, \dots, n$ )

$$v_{ij}(new) = v_{ij}(old) + \Delta v_{ij}$$

Step 9 :  
Menguji apakah kondisi berhenti sudah terpenuhi.

Algoritma Testing

Setelah *training*, jaringan syaraf *backpropagation* diaplikasikan dengan menggunakan fase yang diberikan sebelumnya dari algoritma *training*. Prosedur aplikasinya adalah sebagai berikut :

Step 0 :

Inisialisasi bobot (dari algoritma *training*)

Step 1 :

Untuk setiap vektor *input*,  
kerjakan *step* 2-4

Step 2 :

Untuk  $i = 1, \dots, n$ : set aktifasi dari unit *input*  $x_i$ ;

Step 3 :

Untuk  $j = 1, \dots, p$ :

$$z\_in_j = v_{0j} + \sum_{i=1}^n x_i v_{ij} ;$$

$$z_j = f(z\_in_j)$$

Step 4 :

$$\text{Untuk } k = 1, \dots, m: y\_in_k = w_{0k} + \sum_{j=1}^p z_j w_{jk} ;$$

$$y_k = f(y\_in_k)$$

Penelitian ini mencari perbandingan keakurasian hasil dari dua metode jaringan yang digunakan. Data ditraining dengan menggunakan kedua metode klasifikasi yaitu jaringan LVQ dan Backpropagation dengan masing-masing algoritma berbeda. Setelah data ditraining dan ditesting dengan kedua metode tersebut, selanjutnya hasil training keduanya dibandingkan. Di bawah ini akan diberikan hasil klasifikasi dari kedua metode klasifikasi. Hasil klasifikasi yang diperoleh setelah proses training dan testing dapat dilihat pada tabel berikut :

Tabel 1 : Hasil klasifikasi dengan menggunakan  $\alpha = 0.5$  dengan jumlah epoch = 100, Jumlah training 345 dan jumlah data testing 86 pasien

Metode	Error	Persentasi keakurasian	Waktu (detik)
LVQ	27.441861	82.558139	1.00
Backpropagation	26.744187	73.255813	1.53

**KESIMPULAN**

Berdasarkan hasil dan pembahasan, dapat disimpulkan bahwa untuk proses training, metode LVQ lebih akurat dibandingkan metode Backpropagation. Hal ini terlihat dari besarnya error klasifikasi dan waktu yang digunakan klasifikasinya. Sehingga klasifikasi data diabetes menggunakan LVQ memberikan tingkat akurasi lebih tinggi atau akurat dalam membaca pola jika dibandingkan klasifikasi data menggunakan jaringan Backpropagation.

## **SARAN**

Dalam rangka pengembangan penelitian perlu dilakukan pada klasifikasi kasus yang lain karena jaringan LVQ dan Backpropagation masih dapat memung-kinkan untuk dikaji dan dikembangkan.

## **DAFTAR PUSTAKA**

- Arief, M. (2001). *Diabetes Mellitus dan Komplikasinya*. Puslitbang Kesehatan. Jakarta
- Fausett, L. (1994). *Fundamentals of Neural Networks: Architecture, Algorithms, and Applications*, Prentice Hall
- Jorgens, V. dkk. (1994). *Bagaimana Mengobati Diabetes secara Mandiri untuk Pasien yang tidak Menyuntikkan Insulin*. Fakultas Kedokteran Universitas Indonesia.
- Kusumadewi, S. (2003). *Artificial Intelligence*. Graha Ilmu, Yogyakarta.
- Kuzuya, T. Dkk. (2002). "Report of Committee on the Classification and Diagnostic Criteria of Diabetes Mellitus". *Diabetes Research and Clinical Practice*. 55. Hal 65-85.
- Laksana, Teddy. (1996). *Perancangan Alat Pradiagnosis untuk Diabetes*. Surabaya
- Rafflesia, U. (2010). *Perbandingan Performansi Jaringan LVQ dan RBF untuk Permasalahan Klasifikasi pada Penyakit Karies Gigi*. Tesis Magister. ITS. Surabaya.
- Siang JJ. (2005). *Jaringan Syaraf Tiruan & Pemrogramannya Menggunakan Matlab*, Andi Yogyakarta
- Subianto, T. (2009). *Diabetes Mellitus*. Asuhan keperawatan.  
<http://teguhsuanto.blogspot.com/2009/06/asuhan-keperawatan-diabetesmellitus.html>.  
Diakses pada Tanggal 7 Januari 2011.

