



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB II

LANDASAN TEORI

2.1 Jaringan Syaraf Tiruan

2.1.1 Definisi

Jaringan syaraf tiruan merupakan simulasi dari otak biologis yang bertujuan untuk belajar mengenali pola data dan mensimulasikan proses belajar adaptif biologis, walau dalam skala yang sangat sederhana (Fu, 1994). Jaringan syaraf tiruan merupakan salah satu representasi buatan dari otak manusia yang selalu untuk mensimulasikan proses pembelajaran pada otak manusia tersebut. Istilah buatan di sini digunakan karena jaringan syaraf ini diimplementasikan dengan menggunakan program komputer yang mampu menyelesaikan sejumlah proses perhitungan selama proses pembelajaran (Kusumadewi, 2003).

JST dapat menangkap pengetahuan dari contoh domain yang dengan mudah baik dari data diskrit maupun kontinu. Serta memiliki kemampuan untuk mengeneralisasi yang baik seperti sistem fuzzy. JST adalah model komputasi dari otak manusia. JST berasumsi perhitungan yang didistribusikan melalui beberapa unit sederhana yang disebut neuron, yang saling berhubungan dan beroperasi secara paralel sehingga dikenal sebagai sistem pemrosesan paralel terdistribusi atau sistem koneksi (Agarkar dan Ghatol, 2010).

Pengetahuan implisit dibangun dan dilatihkan ke jaringan syaraf. Beberapa JST dapat dilatih oleh pola masukan yang khas dan pola keluaran yang sesuai dengan yang diharapkan. Kesalahan antara *output* dengan aktual diharapkan dapat digunakan untuk memperkuat bobot dari koneksi antara neuron. Jenis pelatihan ini dikenal dengan pelatihan terbimbing. Beberapa JST yang lain dilatih dalam model tanpa pengawasan, dimana hanya pola masukan yang diberikan selama pelatihan dan jaringan belajar secara otomatis untuk cluster mereka dalam kelompok dengan fitur yang serupa (Agarkar dan Ghatol, 2010).

Jaringan syaraf tiruan (JST) adalah sistem pemroses informasi yang memiliki karakteristik mirip dengan jaringan syaraf biologi. JST dibentuk sebagai generalisasi model matematika dari jaringan syaraf biologi (Siang, 2005), dengan asumsi bahwa :

1. Pemrosesan informasi terjadi pada banyak elemen sederhana (neuron)
2. Sinyal dikirimkan diantara neuron-neuron melalui penghubung-penghubung
3. Penghubung antar neuron memiliki bobot yang akan memperkuat atau memperlemah sinyal
4. Untuk menentukan output, setiap neuron menggunakan fungsi aktivasi (biasanya bukan fungsi linier) yang dikenakan pada jumlahan input yang diterima. Besarnya output ini selanjutnya dibandingkan dengan suatu batas ambang.

JST ditentukan oleh 3 hal :

1. Pola hubungan antar neuron (disebut arsitektur jaringan)
2. Metode untuk menentukan bobot penghubung (disebut metode *training/learning/algorithm*)
3. Fungsi aktivasi

JST merupakan sebuah tiruan dari jaringan syaraf manusia, oleh karena itu JST juga memiliki sifat yang serupa dengan manusia diantaranya :

1. Pengetahuan yang diperoleh dari proses pelatihan/belajar.
2. Koneksi antar neuron dalam otak (sinapsis). Analogi tersebut dalam JST disebut sebagai bobot sinapsis yang berfungsi sebagai penyimpan pengetahuan (memori).
3. Cara belajar otak dianalogikan dengan prosedur algoritma yang memakai algoritma belajar tertentu yang merupakan suatu fungsi untuk mengubah bobot sinapsis jaringan.

Algoritma JST beroperasi secara langsung dengan angka sehingga data yang tidak numerik harus diubah menjadi data numerik. JST tidak dapat menerangkan bagaimana suatu hasil didapatkan. Hal inilah yang membuat JST mampu digunakan untuk menyelesaikan persoalan yang tidak terstruktur dan sulit didefinisikan. JST memiliki banyak kelebihan (Hermawan, 2006), antara lain:

1. Kemampuan mengakuisisi pengetahuan walaupun dalam kondisi ada gangguan dan ketidakpastian. Hal ini karena JST mampu melakukan

generalisasi, abstraksi, dan ekstraksi terhadap properti statistik dari data.

2. Kemampuan merepresentasikan pengetahuan secara fleksibel. JST dapat menciptakan sendiri representasi melalui pengaturan diri sendiri atau kemampuan belajar.
3. Kemampuan untuk memberikan toleransi atas suatu distorsi (*error / fault*), di mana gangguan kecil pada data dapat dianggap hanya sebagai *noise* belaka.
4. Kemampuan memproses pengetahuan secara efisien karena memakai system parallel, sehingga waktu yang diperlukan untuk mengoperasikannya lebih singkat.

Meskipun demikian JST juga memiliki sejumlah keterbatasan, antara lain kekurangampuannya dalam melakukan operasi – operasi numerik dengan presisi tinggi, operasi algoritma aritmatik, operasi logika, dan operasi simbolis serta lamanya proses pelatihan yang terkadang membutuhkan waktu berhari – hari untuk jumlah data yang banyak.

2.1.2 Aplikasi Jaringan Syaraf Tiruan

Beberapa aplikasi jaringan syaraf tiruan adalah sebagai berikut (Siang, 2005):

1. Pengenalan Pola (*Pattern Recognition*)

Jaringan syaraf tiruan dapat dipakai untuk mengenali pola (missal huruf, angka, suara atau tanda tangan) yang sudah sedikit berubah. Hal ini mirip dengan otak manusia yang masih mampu

mengenal orang yang sudah beberapa waktu tidak dijumpainya (mungkin wajah/bentuk tubuhnya sudah sedikit berubah).

2. *Signal Processing*

Jaringan syaraf tiruan (model ADALINE) dapat dipakai untuk menekan *noise* dalam saluran telpon.

3. Peramalan

Jaringan syaraf tiruan juga dapat dipakai untuk meramalkan apa yang akan terjadi di masa yang akan datang berdasarkan pola kejadian yang ada di masa yang lampau. Ini dapat dilakukan mengingat kemampuan jaringan syaraf tiruan untuk mengingat dan membuat generalisasi dari apa yang sudah ada sebelumnya.

Disamping area-area tersebut, jaringan syaraf tiruan juga dilaporkan dapat menyelesaikan masalah dalam bidang kontrol, kedokteran, dan lain-lain. Meskipun banyak aplikasi menjanjikan yang dapat dilakukan oleh jaringan syaraf tiruan, namun jaringan syaraf tiruan juga memiliki beberapa keterbatasan umum. Pertama adalah ketidak akuratan hasil yang diperoleh. Jaringan syaraf tiruan bekerja berdasarkan pola yang terbentuk pada inputnya.

2.1.3 Pemodelan Neuron

Jaringan syaraf terdiri atas beberapa neuron dan ada hubungan antara neuron–neuron tersebut. Neuron adalah sebuah unit pemroses informasi yang menjadi dasar pengoperasian jaringan syaraf tiruan. Syaraf adalah sebuah unit pemroses informasi dengan tiga elemen dasar yaitu (Siang, 2005) :

1. Satu set *link* yang terhubung.
2. Sebuah penjumlahan untuk menghitung besarnya penambahan pada sinyal masukan.
3. Sebuah fungsi aktivasi untuk membatasi banyaknya keluaran pada syaraf.

Sebagian besar jaringan syaraf melakukan penyesuaian bobot – bobotnya selama menjalani pelatihan. Pelatihan dapat berupa pelatihan terbimbing (*supervised training*) di mana diperlukan pasangan masukan – sasaran untuk tiap pola yang dilatihkan. Jenis kedua adalah pelatihan tak terbimbing (*unsupervised training*) (Hermawan, 2006).

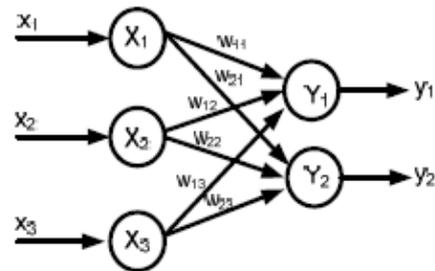
2.1.4 Arsitektur Jaringan

Neuron-neuron dalam suatu sistem jaringan syaraf dikelompokkan dalam lapisan-lapisan. Faktor yang akan menentukan keadaan dari suatu neuron adalah fungsi aktivasi dan pola bobotnya. Pada setiap lapisan yang sama, neuron-neuron akan memiliki fungsi aktivasi yang sama. Setiap neuron pada satu lapisan harus dipetakan tepat ke setiap neuron pada lapisan berikutnya (misalkan lapisan *input* ke lapisan *output* pada *single layer*) (Siang, 2005). Adapun arsitektur dari JST antara lain:

1. Jaringan Satu Lapis (*single layer network*)

Pada jaringan satu lapis, hanya memiliki satu lapisan *input* dan lapisan *output*. Pada jaringan ini sinyal masukan langsung diolah menjadi sinyal keluaran, tanpa melalui *hidden layer* seperti contoh yang ditunjukkan pada gambar 2.1. Pada jaringan satu lapis dapat

ditambah dengan bias yang merupakan bobot koneksi dari unit yang aktivasinya selalu 1.



Gambar 2.1. Jaringan satu lapis

Keterangan:

x_1, x_2, x_3 = nilai *input*

X_1, X_2, X_3 = lapisan *input*

$W_{11}, W_{12}, W_{13}, W_{21}, W_{22}, W_{23}$ = matriks bobot antara *input* dan *output*.

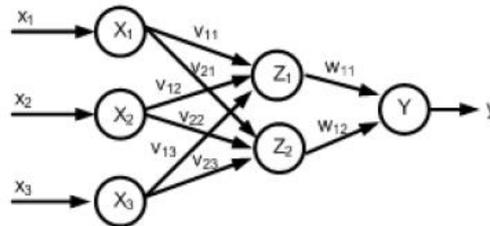
Y_1, Y_2 = lapisan *output*

y_1, y_2 = nilai *output*

2. Jaringan Lapis Banyak (multi layer net)

Jaringan lapis banyak terdiri atas satu lapisan *input*, satu lapisan *output* dan satu atau lebih *hidden layer* yang terletak diantara lapisan *input* dan lapisan *output* seperti contoh yang ditunjukkan pada Gambar 2.2. Jaringan dengan banyak lapisan dapat dimanfaatkan untuk menyelesaikan permasalahan yang lebih sulit dan kompleks daripada jaringan dengan satu lapis, akan tetapi tentu saja dengan

pembelajaran yang lebih rumit dan lama. Pada jaringan lapis banyak juga dapat ditambah dengan satu buah neuron bias pada tiap lapisan.



Gambar 2.2. Jaringan lapis banyak

Keterangan:

x_1, x_2, x_3 = nilai *input*

X_1, X_2, X_3 = lapisan *input*

$v_{11}, v_{12}, v_{13}, v_{21}, v_{22}, v_{23}$ = matriks bobot antara *input* dan *hidden layer*

Z_1, Z_2 = *hidden layer*

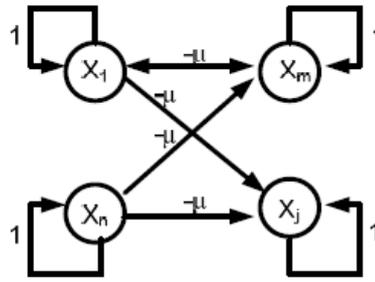
w_{11}, w_{12} = matriks bobot antara *hidden layer* dan *output*

Y = lapisan *output*

y = nilai *output*

3. Jaringan dengan lapisan kompetitif

Hubungan antar neuron pada lapisan kompetitif pada umumnya tidak diperlihatkan pada diagram arsitektur. Salah satu contoh dari arsitektur jaringan dengan lapisan kompetitif diperlihatkan pada Gambar 2.3 yang memiliki bobot $-\mu$.



Gambar 2.3. Jaringan dengan lapisan kompetitif

2.1.5 Konsep Pembelajaran Jaringan Syaraf Tiruan

Ciri utama dari sebuah sistem JST adalah kemampuan untuk belajar dari contoh ataupun pengalaman terdahulu. Agar sistem dapat berfungsi dengan baik, jaringan tidak diprogram seperti yang dilakukan pada sistem komputer konvensional akan tetapi harus dengan pembelajaran ataupun pelatihan.

Fungsi JST ditentukan oleh bobot-bobot penghubungnya. Hal ini didasarkan pada fungsi masukan keluarannya. Bobot-bobot koreksi pada beberapa kasus dapat berupa variabel yang telah ditentukan pada saat perancangan, tetapi ada juga jaringan yang harus mencari sendiri besarnya bobot-bobot yang sesuai.

Proses penyesuaian parameter secara berurutan dilakukan dengan tujuan untuk mendekati fungsi yang diinginkan. Proses penyesuaian parameter (bobot) inilah dalam JST disebut proses belajar atau pelatihan.

Proses belajar dikategorikan dalam dua jenis (Hermawan, 2006):

1. Proses belajar Terbimbing (*supervised learning*)

Algoritma pembelajaran terbimbing memerlukan keluaran/target yang telah diketahui sebelumnya, target sebagai dasar perubahan bobot. Pada proses pelatihan, satu sinyal masukan diberikan kepada satu neuron yang selanjutnya akan diolah dan akan menghasilkan satu keluaran. Nilai keluaran ini akan dibandingkan dengan nilai targetnya. Jika terjadi perbedaan antara *output* pembelajaran dengan *output* target, maka disini akan timbul *error*. Untuk mendapatkan nilai keluaran yang sesuai dengan target (*error* yang terjadi kecil), maka harus dilakukan pelatihan secara berkala. Pada algoritma pembelajaran terbimbing jaringan dibimbing agar dapat menyelesaikan persoalan-persoalan yang terdapat dalam paket pembelajarannya.

2. Proses belajar tidak terbimbing (*unsupervised learning*)

Pada algoritma pembelajaran tak terbimbing, jaringan akan mengubah bobot-bobotnya, sehingga tanggapan terhadap masukan tanpa memerlukan keluaran acuan/ target. Tujuan pembelajaran ini adalah mengelompokkan unit-unit yang hampir sama dalam suatu area tertentu. Pembelajaran ini cocok untuk pengelompokan pola.

2.1.6 Normalisasi Data

Di dalam jaringan syaraf tiruan, akan lebih baik apabila semua data masukan dinormalisasi terlebih dahulu. Normalisasi data dapat membuat proses pelatihan lebih cepat, efisiensi memori, dan meningkatkan tingkat

akurasi. JST bekerja dengan baik dengan data yang memiliki jarak tertentu, misalnya 1 ke 1 atau 0 ke 1, sehingga data perlu dinormalisasi. Normalisasi data dapat memastikan data yang terdistribusi di antara lapisan masukan sampai ke lapisan keluaran kurang lebih selalu seragam. (Mendelssohn, 1993.)

Normalisasi data dapat dilakukan dengan berbagai cara, salah satunya adalah dengan menggunakan metode *Gaussian*. Normalisasi data dengan metode *Gaussian* dapat diperoleh dengan cara :

$$\frac{x - \text{mean}}{\sqrt{\frac{\sum_{i=1}^n (x_i - \text{mean})^2}{n}}}$$

Keterangan :

X = data yang dinormalisasi

Mean = rata – rata seluruh data x

N = banyaknya data.

2.1.7 Fungsi Aktivasi

Fungsi aktivasi merupakan bagian penting dalam tahapan perhitungan keluaran dari suatu algoritma. Fungsi aktivasi merupakan fungsi matematis yang berfungsi membatasi dan menentukan jangkauan *output* suatu neuron. Dengan adanya fungsi aktivasi ini membuat setiap neuron yang dipakai pada neuron menjadi aktif. Beberapa fungsi aktivasi yang digunakan dalam JST adalah (Hermawan, 2006):

1. Fungsi Sigmoid

Fungsi sigmoid ada dua, yaitu pertama fungsi sigmoid biner, memiliki nilai pada range 0 s.d 1. Fungsi sigmoid biner dirumuskan sebagai:

$$f(x) = \frac{1}{1 + e^{-\alpha x}}$$

dengan turunan dari $f(x)$ adalah:

$$f'(x) = \sigma f(x)[1-f(x)]$$

Kedua adalah fungsi sigmoid bipolar, yang memiliki nilai antara -1 s.d 1. Fungsi sigmoid bipolar dirumuskan sebagai:

$$f(x) = \frac{1 - e^{-\alpha x}}{1 + e^{-\alpha x}}$$

dengan turunan dari $f(x)$ adalah:

$$f'(x) = \frac{\sigma}{2} [1+f(x)][1-f(x)]$$

2. Fungsi *Threshold* (batas ambang)

Fungsi *threshold* ada dua, yaitu:

a. Fungsi *threshold* biner, yang dirumuskan:

$$F(x) = \begin{cases} 1, & \text{jika } x \geq a \\ 0, & \text{jika } x < a \end{cases}$$

b. Fungsi *threshold* bipolar, yang dirumuskan:

$$F(x) = \begin{cases} 1, & \text{jika } x \geq a \\ -1 & \text{jika } x < a \end{cases}$$

dengan a adalah batas ambang. Jika *threshold* a bernilai 0, maka fungsi ini disebut sebagai fungsi undak.

3. Fungsi Identitas (linear)

Fungsi identitas digunakan untuk jaringan yang memiliki keluaran sembarang nilai riil (bukan hanya pada range [0 1] atau [0

-1]). Fungsi identitas memiliki nilai keluaran yang sama dengan nilai masukannya. $y = f(x) = x$ (2.7)

4. Fungsi Saturating Linier

Fungsi saturating linier dirumuskan sebagai:

$$f(x) = \begin{cases} 1 & \text{jika } x \geq 0,5 \\ x+0,5 & \text{jika } -0,5 < x < 0,5 \\ 0 & \text{jika } x \leq -0,5 \end{cases}$$

5. Fungsi Simetrik Saturating Linier

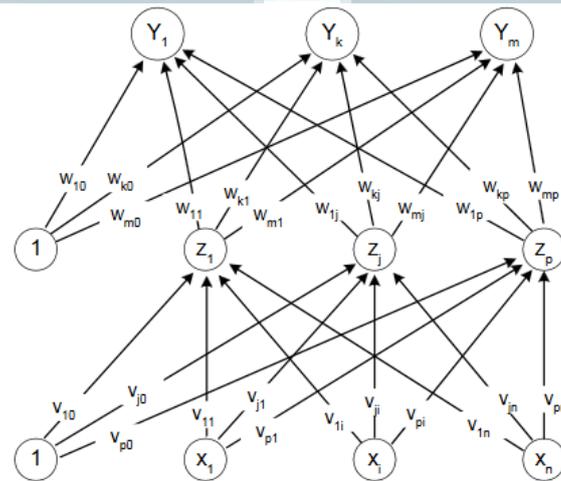
Fungsi simetrik saturating linier dirumuskan sebagai:

$$f(x) = \begin{cases} -1; & \text{jika } x < -1 \\ x; & \text{jika } -1 \leq x \leq 1 \\ 1; & \text{jika } x > 1 \end{cases}$$

2.1.8 Arsitektur Jaringan Syaraf Tiruan *Backpropagation*

Perambatan galat mundur (*backpropagation*) merupakan sebuah metode untuk pelatihan jaringan syaraf tiruan agar mampu mendapatkan keseimbangan antara kemampuan jaringan untuk mengenali pola yang digunakan selama pelatihan serta kemampuan jaringan untuk memberikan respon yang benar terhadap pola masukan yang serupa (tapi tidak sama) dengan pola yang dipakai selama pelatihan (Siang, 2005).

Backpropagation memiliki beberapa unit yang ada dalam satu atau lebih layar tersembunyi. Gambar 2.4 adalah arsitektur backpropagation dengan n buah masukan (ditambah sebuah bias), sebuah layar tersembunyi yang terdiri dari p unit (ditambah sebuah bias), serta m buah unit keluaran.



Gambar 2.4. Arsitektur *backpropagation*

v_{ji} merupakan bobot garis dari unit masukan x_i ke unit layar tersembunyi z_j (v_{j0} merupakan bobot garis yang menghubungkan bias di unit masukan ke unit layar tersembunyi z_j). w_{kj} merupakan bobot dari unit layar tersembunyi z_j ke unit keluaran y_k (w_{k0} merupakan bobot dari bias di layar ters ke unit keluaran z_k).

2.1.8 Atribut Jaringan Syaraf Tiruan Backpropagation

Agar pelatihan dapat berlangsung dengan lancar dan mampu menghasilkan nilai yang akurat, maka ada beberapa atribut yang perlu di atur sedemikian rupa yang akan berdampak pada jalannya pelatihan JST.

1. *Max. epoch*

Epoch merupakan satuan menyatakan berapa banyak algoritma JST yang telah di jalankan untuk memeriksa seluruh data masukan. *Max. epoch* ditentukan untuk menentukan batasan maksimal iterasi dari *backpropation*.

2. *Hidden Layer*

Banyaknya *hidden layer* mampu turut serta mempengaruhi hasil dari perhitungan JST. Tidak ada aturan pasti dalam menentukan banyaknya lapisan tersembunyi pada JST. Salah satu cara menentukan *hidden layer* adalah dengan *trial and error*.

3. Bobot dan bias

Bobot digunakan untuk menerjemahkan data masukan sehingga dapat digunakan untuk lapisan selanjutnya untuk menentukan hasil. Bobot biasa diisi dengan bilangan acak kecil.

4. Laju Pembelajaran

Laju pembelajaran sangat berpengaruh terhadap proses pelatihan. Laju pembelajaran merupakan intensitas pelatihan yang dilakukan, semakin besar laju pembelajaran maka pelatihan akan semakin cepat.

5. Momentum

Momentum merupakan perubahan yang terjadi terhadap bobot yang didasarkan pada gradien pola terakhir dan pola sebelum – sebelumnya. Penambahan momentum dimaksudkan untuk

menghindari perubahan bobot yang mencolok akibat adanya data yang sangat berbeda.

2.1.8 Algoritma Pelatihan *Backpropagation*

Backpropagation melatih jaringan untuk mendapatkan keseimbangan antara kemampuan jaringan untuk mengenali pola yang digunakan selama pelatihan serta kemampuan jaringan untuk memberikan respon yang benar terhadap pola masukan yang serupa (tapi tidak sama) dengan pola yang dipakai selama pelatihan (Siang, 2005). Algoritma pelatihan menggunakan *backpropagation* terdiri dari beberapa langkah yaitu:

1. Inisialisasi semua bobot dengan bilangan acak yang terkecil
2. Selama kondisi masih bernilai *false*, lakukan langkah – langkah berikut:
3. Phase 1 Feedforward :
 - a. Tiap unit masukan ($X_i; i=1,..,n$) menerima sinyal dan meneruskannya ke lapisan tersembunyi. Data *input* merupakan dataset pelatihan yang sebelumnya telah dipilih.
 - b. Pada setiap lapisan tersembunyi ($Z_j; j=1,..,p$), akan dilakukan perhitungan terhadap seluruh input dan bobotnya untuk mendapatkan nilai $Z_{net j}$.

$$z_{net j} = v_{j0} + \sum_{i=1}^n x_i v_{ji}$$

Hasil dari perhitungan tersebut lalu akan di aktivasi menjadi sinyal keluaran (Z_j) dengan menggunakan sigmoid

$$z_j = f(z_{net_j}) = \frac{1}{1 + e^{-z_{net_j}}}$$

Hasil perhitungan akan dikirim ke seluruh unit pada unit output.

- c. Setiap unit output Y_k ($k = 1, 2, \dots, m$), akan dilakukan perhitungan terhadap sinyal keluaran sebelumnya dan bobotnya untuk mendapatkan nilai Y_{net_k} dengan W_{kj} sebagai bias pada

$$y_{net_k} = w_{k0} + \sum_{j=1}^p z_j w_{kj}$$

unit keluaran j

Hasil juga akan diaktivasi dengan menggunakan sigmoid untuk mendapatkan sinyal keluaran (Y_k)

$$y_k = f(y_{net_k}) = \frac{1}{1 + e^{-y_{net_k}}}$$

4. Phase 2 Backpropagation

- a. Setiap unit output ($Y_k, k = 1, \dots, m$) menerima sebuah sinyal dari perhitungan sebelumnya dan kemudian akan menghitung error (δ_k) antara target dan *output* yang dihasilkan.

$$\delta_k = (t_k - y_k) f'(y_{net_k}) = (t_k - y_k) y_k (1 - y_k)$$

δ_k kemudian akan digunakan untuk menghitung koreksi error (ΔW_{jk}) yang nantinya akan dipakai untuk memperbaharui W_{jk} ,

dimana :

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

Selain itu juga dihitung koreksi bias ΔW_{k0} yang nantinya akan dipakai untuk memperbaharui W_{k0} , dimana :

$$\Delta W_{k0} = \alpha \delta_0$$

- b. Setelah itu dilakukan perhitungan faktor δ berdasarkan kesalahan di setiap unit tersembunyi $Z_j (j = 1, 2, \dots, p)$

$$\delta_{net_j} = \sum_{k=1}^m \delta_k w_{kj}$$

Kemudian hasilnya dikalikan dengan turunan dari fungsi aktivasi yang digunakan jaringan untuk menghasilkan faktor koreksi error δ_j , dimana :

$$\delta_j = \delta_{net_j} f'(z_{net_j}) = \delta_{net_j} z_j (1 - z_j)$$

- c. Kemudian, dilakukan perhitungan terhadap perubahan bobot V_{ji} (yang akan dipakai nanti untuk merubah bobot V_{ji})

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

Selain itu juga dihitung koreksi bias ΔV_{j0} yang nantinya akan dipakai untuk memperbaharui V_{j0} , dimana :

$$\Delta V_{j0} = \alpha \delta_j$$

5. Phase 3 Perubahan Bobot dan Bias

Setiap unit output ($Y_k, k = 1, \dots, m$) akan memperbaharui bias dan bobotnya dari setiap hidden unit ($j = 0, \dots, p$), dimana

- a. Perubahan bobot garis yang menuju ke unit keluaran :

$$w_{kj}(\text{baru}) = w_{kj}(\text{lama}) + \Delta w_{kj}$$

- b. Perubahan bobot garis yang menuju ke unit tersembunyi :

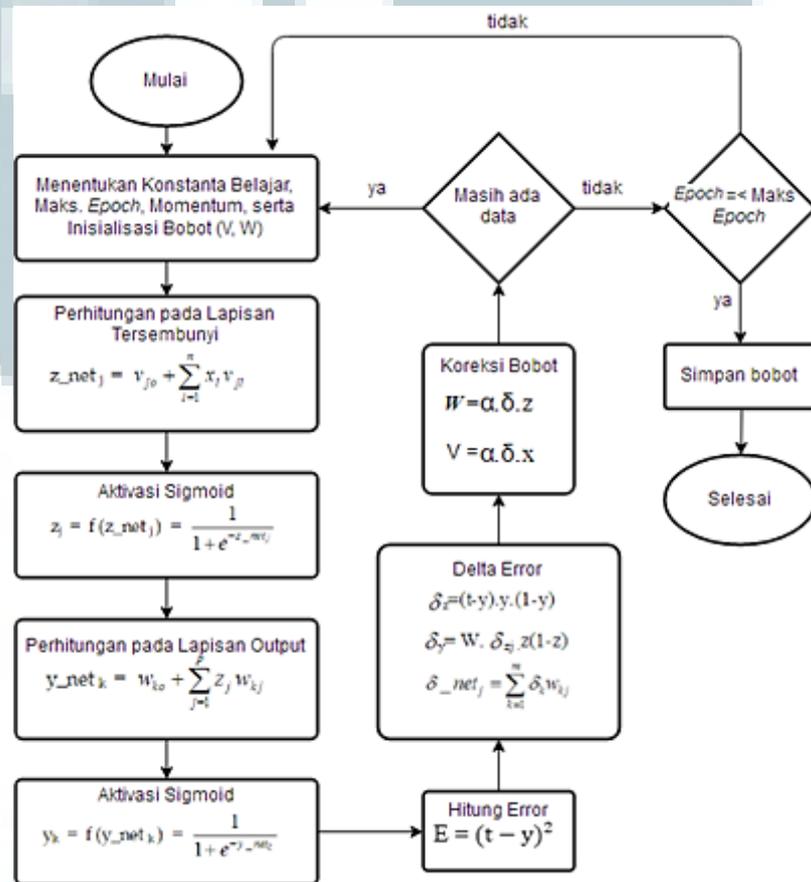
$$v_{ji}(\text{baru}) = v_{ji}(\text{lama}) + \Delta v_{ji} \quad (j = 1, 2, \dots, p ; i = 0, 1, \dots, n)$$

6. Memeriksa kondisi iterasi

Dalam propagasi mundur, Batasan iterasi dihitung dengan menggunakan *Mean Square Error* untuk menghitung error yang didapat antara hasil perhitungan dengan target, dimana :

$$\text{Mean Square Error} = E = t * \{ (t_{k1} - y_{k1})^2 + (t_{k2} - y_{k2})^2 + \dots + (t_{km} - y_{km})^2 \}$$

Gambar 2.5 berikut adalah *flowchart* berdasarkan penjelasan seperti yang disebutkan di atas.



Gambar 2.100. Pelatihan *backpropagation*

2.2 Diabetes Melitus

2.2.1 Definisi

Diabetes melitus merupakan salah satu kelompok penyakit metabolik yang ditandai oleh hiperglikemia karena gangguan sekresi insulin, kerja insulin, atau keduanya. Keadaan hiperglikemia kronis dari diabetes berhubungan dengan kerusakan jangka panjang, gangguan fungsi dan kegagalan berbagai organ, terutama mata, ginjal, saraf, jantung, dan pembuluh darah (*American Diabetes Association, 2012*). *Hiperglikemia* sendiri adalah istilah medis untuk keadaan di mana kadar gula dalam darah lebih tinggi dari nilai normal. Dalam keadaan normal, gula darah berkisar antara 70 – 100 mg/dL. Kadar gula biasanya sedikit meningkat dari nilai normal sesaat sesudah makan, tapi keadaan ini tidak dianggap hiperglikemia.

Diabetes melitus adalah sindrom klinis yang ditandai dengan hiperglikemia karena defisiensi insulin yang absolut maupun relatif. Kurangnya hormon insulin dalam tubuh yang dikeluarkan dari sel B pankreas mempengaruhi metabolisme karbohidrat, protein, dan lemak menyebabkan gangguan signifikan. Kadar glukosa darah erat diatur oleh insulin sebagai regulator utama perantara metabolisme. Hati sebagai organ utama dalam transport glukosa yang menyimpan glukosa sebagai glikogen dan kemudian dirilis ke jaringan perifer ketika dibutuhkan (*Animesh, 2006*).

2.2.2 Epidemiologi

Berdasarkan data dari *World Health Organization* (2016), diperkirakan terdapat 422 juta orang dewasa di seluruh dunia yang menderita diabetes pada

tahun 2014 dimana jumlah tersebut meningkat 4 kali lipat dibandingkan dengan 108 juta orang pada tahun 1980. Peningkatan drastis ini disebabkan oleh meningkatnya faktor dari diabetes tipe 2 yakni obesitas.

Sepuluh besar negara dengan dengan penderita DM tertinggi di dunia menurut Federasi Diabetes International pada tahun 2015 adalah Cina, India, Amerika , Brazil, Rusia, Mexico, Indonesia, Egypt, Jepang, dan Bangladesh. Pada tahun 2040 Cina, India, dan Amerika diprediksikan tetap menduduki posisi tiga teratas negara dengan penderita DM tertinggi. Sementara, Indonesia diprediksikan akan tetap berada dalam sepuluh besar.

2.2.3. Klasifikasi Diabetes Mellitus

Menurut American Diabetes Association (2012), klasifikasi diabetes meliputi empat kelas klinis :

a. Diabetes Melitus tipe 1

Hasil dari kehancuran sel β pankreas, biasanya menyebabkan defisiensi insulin yang absolut.

b. Diabetes Melitus tipe 2

Hasil dari gangguan sekresi insulin yang progresif yang menjadi latar belakang terjadinya resistensi insulin.

c. Diabetes tipe spesifik lain

Misalnya : gangguan genetik pada fungsi sel β , gangguan genetik pada kerja insulin, penyakit eksokrin pankreas (seperti *cystic fibrosis*), dan

yang dipicu oleh obat atau bahan kimia (seperti dalam pengobatan HIV/AIDS atau setelah transplantasi organ).

d. *Gestational* Diabetes Melitus

Diabetes yang terjadi dalam masa kehamilan.

2.2.4. Patogenesis Diabetes Melitus

2.2.4.1 Diabetes Melitus Tipe 1

DM tipe 1 adalah hasil dari interaksi genetik, lingkungan, dan faktor imunologi yang pada akhirnya mengarah pada kerusakan sel β pankreas dan defisiensi insulin. DM tipe 1 adalah hasil dari interaksi genetik, lingkungan, dan faktor imunologi yang pada akhirnya mengarah terhadap kerusakan sel β pankreas dan insulin defisiensi. Massa sel β kemudian menurun dan sekresi insulin menjadi semakin terganggu, meskipun toleransi glukosa normal dipertahankan (Franz *et al.*, 2010).

DM tipe 1 disebut juga diabetes yang diperantarai imun. Diabetes yang tipe ini hanya 5-10% dari penderita diabetes. Tanda dari penghancuran imun sel β termasuk autoantibodi sel islet, autoantibodi terhadap insulin, autoantibodi untuk GAD (GAD65), dan autoantibodi terhadap tirosin fosfatase IA-2 dan IA-2b. DM tipe 1 ini, tingkat kehancuran sel β cukup bervariasi, menjadi cepat pada beberapa individu (terutama bayi dan anak-anak) dan lambat pada orang lain (terutama dewasa). Beberapa pasien, terutama anak-anak dan remaja,

dapat hadir dengan ketoasidosis sebagai manifestasi pertama penyakit. Namun orang lain, terutama orang dewasa, dapat mempertahankan fungsi sel β sisa yang cukup untuk mencegah ketoasidosis selama bertahun-tahun, orang tersebut akhirnya menjadi tergantung pada insulin untuk bertahan hidup dan beresiko untuk ketoasidosis. Pada tahap selanjutnya dari penyakit, ada sedikit atau tidak ada sekresi insulin sebagai manifestasi dari rendah atau tidak terdeteksi C-peptida di dalam plasma. DM tipe 1 umumnya terjadi pada masa kanak-kanak dan remaja, tetapi bisa terjadi pada usia berapapun, bahkan dalam dekade 8 dan 9 kehidupan. Kehancuran autoimun sel β memiliki beberapa kecenderungan genetik dan juga terkait dengan faktor lingkungan yang masih buruk. Walaupun pasien jarang obesitas ketika mereka hadir dengan diabetes tipe ini, kehadiran obesitas tidak bertentangan dengan diagnosis. Pasien-pasien ini juga rentan terhadap gangguan autoimun lainnya seperti penyakit Graves, tiroiditis Hashimoto, penyakit Addison, vitiligo, celiac sprue, hepatitis autoimun, myasthenia gravis, dan anemia pernisiiosa (*American Diabetes Association, 2012*).

Beberapa bentuk DM tipe 1 tidak memiliki etiologi yang dikenal, disebut dengan idiopatik diabetes. Beberapa pasien dengan diabetes ini memiliki insulinopenia dan rentan terhadap ketoasidosis, tetapi tidak memiliki bukti autoimun (*American Diabetes Association, 2012*).

2.2.4.2 Diabetes Mellitus Tipe 2

DM tipe 2 ditandai dengan gangguan sekresi insulin, resistensi insulin, produksi glukosa hepatic yang berlebihan, dan abnormal metabolisme lemak. Obesitas, khususnya visceral atau pusat (yang dibuktikan dengan rasio pinggul/pinggang), sangat umum pada penderita DM tipe 2. Pada tahap awal gangguan, toleransi glukosa tetap mendekati normal, meskipun resistensi insulin, karena sel-sel β pankreas mengkompensasi dengan meningkatkan produksi insulin. Resistensi insulin dan kompensasi hiperinsulinemia, pankreas pada individu tertentu tidak dapat mempertahankan keadaan hiperinsulinemia. IGT, ditandai dengan peningkatan glukosa postprandial, kemudian berkembang. Lebih lanjut, penurunan sekresi insulin dan peningkatan produksi glukosa hepatic menyebabkan diabetes dengan hiperglikemia puasa. Akhirnya, kegagalan sel β mungkin terjadi (Franz *et al.*, 2010).

a) Gangguan metabolisme otot dan lemak

Resistensi insulin, penurunan kemampuan insulin untuk bertindak efektif pada jaringan target (terutama otot, hati, dan lemak), adalah fitur yang menonjol dari DM tipe 2 dan hasil dari kombinasi kerentanan genetik dan obesitas. Resistensi insulin adalah relatif, tingkat supernormal insulin yang beredar akan menormalkan glukosa plasma. Kurva insulin dosis-respon menunjukkan pergeseran ke kanan, menunjukkan sensitivitas

berkurang, menunjukkan penurunan secara keseluruhan dalam penggunaan glukosa maksimum (30-60% lebih rendah dibandingkan orang normal). Resistensi insulin menyebabkan kegagalan penggunaan glukosa oleh jaringan yang sensitif terhadap insulin dan output glukosa hepatic meningkat, kedua efek ini berkontribusi untuk hiperglikemia.

Peningkatan output glukosa hepatic terutama menyumbang peningkatan tingkat FPG, sedangkan hasil penggunaan glukosa perifer menurun menyebabkan postprandial hiperglikemia. Dalam otot rangka, ada yang lebih besar penurunan dalam penggunaan glukosa non-oxidatif (pembentukan glikogen) dibandingkan metabolisme glukosa oksidatif melalui glikolisis. Mekanisme molekuler yang tepat mengarah ke resistensi insulin pada DM tipe 2 belum dijelaskan. Tingkat insulin reseptor dan aktivitas tirosin kinase di otot berkurang, tetapi perubahan ini kemungkinan sekunder untuk hiperinsulinemia. Kemudian, gangguan postreseptor pada regulasi insulin fosforilasi/defosforilasi dapat menjadi peran dominan dalam resistensi insulin. Misalnya, gangguan PI-3-kinase dapat mengurangi translokasi GLUT 4 ke membran plasma. Ketidaknormalan lainnya termasuk akumulasi lipid dalam miosit skeletal, yang dapat merusak fosforilasi oksidatif mitokondria dan mengurangi stimulasi insulin mitokondria memproduksi ATP. Gangguan oksidasi lemak dan akumulasi lipid

dalam miosit dapat menghasilkan oksigen reaktif seperti lipid peroksida (Franz *et al.*, 2010).

Obesitas menyertai DM tipe 2, dianggap bagian dari proses patogenik. Peningkatan massa adiposit mengarah ke peningkatan tingkat sirkulasi asam lemak bebas dan lemak lainnya. Selain mengatur berat badan, nafsu makan, dan pengeluaran energi, adipokines juga memodulasi sensitivitas insulin dan menyebabkan resistensi insulin pada otot rangka dan juga hati. Adiposit dan adipokin juga memproduksi keadaan inflamasi dan mungkin menjelaskan mengapa tanda peradangan seperti IL-6 dan protein C-reaktif sering meningkat pada DM tipe 2 (Franz *et al.*, 2010).

b) Gangguan Sekresi Insulin

Sekresi insulin dan sensitivitas insulin adalah saling terkait. Pada DM tipe 2, sekresi insulin meningkat pada awalnya mengkompensasi resistensi insulin untuk menjaga toleransi glukosa normal. Awalnya, sekresi insulin mengalami defek ringan dan selektif melibatkan stimulasi glukosa untuk sekresi insulin. Menanggapi sekretagogues non-glukosa lain, seperti arginin masih dipertahankan. Akhirnya defek sekresi insulin berkembang menjadi keadaan sekresi insulin sangat tidak memadai. Alasan untuk penurunan kapasitas sekresi insulin di DM tipe 2 tidak jelas. Asumsinya adalah defek genetik pada resistensi insulin

menyebabkan kegagalan sel β . Terbentuk amiloid polipeptida pada pulau langerhans, sehingga berdampak negatif terhadap fungsi pulau langerhans. Tingginya kadar asam lemak bebas dan lemak makanan juga dapat memperburuk fungsi sel β (Franz *et al.*, 2010).

c) **Peningkatan Produksi Glukosa Hepar**

Pada DM tipe 2, resistensi insulin pada hepar merefleksikan kegagalan hiperinsulinemia untuk menekan glukoneogenesis, yang menyebabkan kondisi hiperglikemia dan penurunan simpanan glikogen oleh hepar pada masa pascaprandial. Peningkatan produksi glukosa oleh hepar terjadi pada masa-masa awal diabetes, meskipun sepertinya setelah onset gangguan sekresi insulin dan resistensi insulin pada otot rangka. Sebagai hasil dari resistensi insulin pada jaringan adiposa dan obesitas, asam lemak bebas dari adiposit meningkat, yang menyebabkan peningkatan sintesis lemak (VLDL dan trigliserida) dalam penyimpanan hepatosit lemak. Steatosis dalam hati dapat menyebabkan penyakit lemak hati non alkohol dan tes fungsi hati yang abnormal. Hal ini juga bertanggung jawab untuk dislipidemia yang ditemukan dalam DM tipe 2 (Franz *et al.*, 2010).

2.2.5 Gejala Diabetes Melitus

Manifestasi utama penyakit DM adalah hiperglikemia, yang terjadi akibat berkurangnya jumlah glukosa yang masuk ke dalam sel, berkurangnya

penggunaan glukosa oleh berbagai jaringan, dan peningkatan produksi glukosa (glukoneogenesis) oleh hati.

Gejala utama yang paling umum ada pada penderita diabetes disebut 3P yakni poliuri, polidipsi dan penurunan berat badan sekalipun asupan kalori memadai, merupakan gejala utama defisiensi insulin. Kadar glukosa plasma jarang melampaui 120 mg/dL pada manusia normal, kadar yang jauh lebih tinggi selalu dijumpai pada pasien defisiensi kerja insulin. Setelah kadar tertentu glukosa plasma dicapai (umumnya > 80 mg/dL), taraf maksimal reabsorpsi glukosa pada tubulus renalis akan dilampaui, dan gula akan diekskresikan ke dalam urin (glukosuria). Volume urin meningkat akibat terjadinya diuresis osmotik dan kehilangan air yang bersifat obligatorik pada saat yang bersamaan (poliuria), kejadian ini selanjutnya akan menimbulkan dehidrasi (hiperosmolaritas), bertambahnya rasa haus dan menyebabkan penderita menjadi banyak minum (polidipsia). Glukosuria menyebabkan kehilangan kalori yang cukup besar (4,1 kkal untuk setiap gram karbohidrat yang diekskresikan keluar), kehilangan ini, jika ditambah lagi dengan hilangnya jaringan otot dan adiposa, akan mengakibatkan penurunan berat badan yang hebat meskipun terdapat peningkatan selera makan (polifagia) dan asupan kalori yang normal atau meningkat (Granner, 2003).

2.2.6 Pemeriksaan Gula Darah

Selain mengetahui gejala – gejala dari diabetes, ada cara lain yang juga dapat dilakukan untuk mendiagnosa penyakit diabetes, yaitu pemeriksaan gula darah. Menurut *American Diabetes Association* pemeriksaan gula darah

itu sendiri ada 4 macam yakni A1C, gula darah sewaktu, gula dara puasa, dan gula darah 2 jam setelah puasa / *post prandial* (*American Diabetes Association*, 2014).

a. A1C

A1C merupakan sebuah tes pengukuran gula darah dalam jangka waktu 2 atau 3 bulan. Tingkat A1C normal berkisar kurang dari 5.7%.

b. Gula darah sewaktu (GDS)

GDS merupakan sebuah pengukuran gula darah dalam waktu kapan pun tanpa memperhatikan waktu ataupun kondisi seseorang. GDS normal berkisar kurang dari 200 mg/dl.

c. Gula darah puasa (GDP)

Gula darah puasa merupakan sebuah pengukuran gula darah dimana seseorang diwajibkan untuk puasa selama kurang lebih 8 jam terlebih dahulu. Biasa dilakukan pada pagi hari sebelum sarapan. Kadar GDP normal berkisar kurang dari 100 mg/dl.

d. Gula darah 2 jam setelah puasa (GDPP)

Pengukuran GDPP dilakukan 2 jam setelah makan. Gunanya adalah untuk menilai seberapa besar fungsi pankreas atau insulin yang di keluarkan oleh pankreas untuk menetralsisir gula darah. Pada umumnya setelah makan pasien akan mengalami kenaikan gula darah dan akan berangsur normal setelah kira-kira dua jam setelahnya. GDPP normal berkisar kurang dari 140 mg/dl.

2.3 Bahasa Pemrograman C#

Menurut Deitel (2011), C# merupakan sebuah bahasa pemrograman yang mempunyai akses secara khusus pada *platform* .NET. C# memungkinkan *programmer* untuk mudah dalam bermigrasi ke .NET karena memiliki akar dari C, C++, dan java. C# memiliki akses ke perpustakaan kelas (*class library*) yang kuat dari komponen - komponen yang sudah dibangun sebelumnya, sehingga memungkinkan *programmer* untuk mengembangkan aplikasi dengan cepat.

Bahasa pemrograman C# yang asli distandarisasi oleh Ecma International pada Desember 2002. Sejak itu, Microsoft sudah menawarkan beberapa ekstensi bahasa yang telah diadopsi sebagai bagian dari standar Ecma C# yang telah direvisi.

2.4 MySQL

MySQL merupakan sistem manajemen basis data relasi yang bersifat *open source* dimana pengguna dapat menyimpan, mencari, mengurutkan dan mendapatkan data dengan efisien melalui sebuah perintah. MySQL menggunakan bahasa SQL (*Structured Query Language*), yaitu bahasa *query* pada basis data yang baku bagi seluruh dunia (Welling, 2001).

Berikut merupakan kelebihan yang dimiliki oleh MySQL (Welling, 2001).

a. *Performance*

MySQL begitu cepat dalam pemrosesan data.

b. *Low cost*

MySQL dapat digunakan tanpa biaya karena merupakan sebuah sistem basis data yang bersifat *open source*. Selain *open source*, MySQL juga menyediakan lisensi komersial tetapi masih dengan biaya yang murah.

c. *Ease of use*

MySQL Bisa dijalankan pada spesifikasi hardware yang rendah dan lebih mudah digunakan termasuk pada saat proses persiapan dibandingkan dengan produk *DBMS* lainnya.

d. *Probability*

MySQL dapat digunakan dan diimplementasikan pada berbagai sistem UNIX dan juga pada *Microsoft Windows*

e. *Source code*

Sama seperti bahasa pemrograman seperti PHP dan sebagainya, pengguna dapat mengubah maupun menambahkan *code* pada *MySQL*.

UMMN