

BAB II

KAJIAN TEORI

Penelitian ini membahas tentang pemetaan wilayah provinsi di Indonesia dengan membentuk *cluster* yang diperoleh dari hasil penerapan algoritma pembelajaran *Self Organizing Map*, dimana hasil *output cluster* dari proses algoritma pembelajaran *Self Organizing Map* tersebut divalidasi menggunakan nilai *Davies Bouldin Index* yang kemudian nilai tersebut akan digunakan dalam proses pemilihan model *cluster* terbaik. Sebelum melakukan proses pembentukan *cluster* ada beberapa hal yang perlu diketahui terlebih dahulu, yaitu :

A. Tanah longsor

1. Definisi Tanah Longsor

Tanah longsor secara umum adalah perpindahan material pembentuk lereng berupa batuan, bahan rombakan, tanah atau material campuran tersebut, bergerak ke bawah atau keluar lereng. Secara geologi tanah longsor adalah suatu peristiwa geologi dimana terjadi pergerakan tanah seperti jatuhnya bebatuan atau gumpalan besar tanah (Nandi, 2007: 6).

Ada 6 jenis tanah longsor (BNPB, 2011), yakni: longsoran translasi, longsoran rotasi, pergerakan blok, runtuh batu, rayapan tanah, dan aliran bahan rombakan. Jenis longsoran translasi dan rotasi paling banyak terjadi di Indonesia. Sedangkan longsoran yang paling banyak memakan korban jiwa manusia adalah aliran bahan rombakan.

Pada prinsipnya tanah longsor terjadi jika gaya pendorong lereng lebih besar dari gaya penahan. Gaya penahan umumnya dipengaruhi oleh kekuatan batuan dan

kepadatan tanah, sedangkan gaya pendorong dipengaruhi oleh besarnya sudut lereng, air, beban serta berat jenis tanah batuan.

Gejala umum tanah longsor ditandai dengan munculnya retakan-retakan di lereng yang sejajar dengan arah tebing, munculnya mata air baru secara tiba-tiba dan tebing yang rapuh serta kerikil yang mulai berjatuhan. Faktor penyebab lainnya adalah sebagai berikut (Nandi, 2007: 6-13):

a. Hujan

Pengaruh curah hujan dalam menghasilkan longsor adalah suatu yang jelas, meskipun sangat sulit untuk menjelas secara tepat (Blong dan Dunkerley,1976). Kesulitan ini muncul karena curah hujan hanya mempengaruhi stabilitas lereng secara tidak langsung melalui pengaruhnya terhadap kondisi pori-pori udara di dalam material lereng. Kemudian Caine (1980) menggunakan istilah “pengaruh pemicu” curah hujan terhadap tanah longsor atau aliran debris.

Karakteristik curah hujan yang memicu tanah longsor atau aliran debris telah digunakan untuk membangun hubungan antara curah hujan dan tanah longsor/aliran debris di berbagai belahan dunia. Parameter curah hujan paling sering diselidiki dalam kaitannya dengan inisiasi longsor meliputi curah hujan kumulatif, curah hujan terdahulu, dan durasi curah hujan.

Berbagai upaya telah dilakukan untuk menentukan sistem peringatan pada berbagai tipe bencana. Sistem peringatan ini pertama kali dikembangkan oleh USGS di San Frasisco (Keefer et al.,1987;Wilson & Wieczorek,1995). Sistem peringatan ini didasarkan pada perkiraan kuantitatif curah hujan dari kantor pelayanan cuaca nasional dalam sebuah sistem jaringan alat pengukur hujan *real-*

time lebih dari 40 buah secara terus-menerus dan ambang batas curah hujan yang menginisiasi tanah longsor (Cannon & Ellen,1985).

Ancaman tanah longsor biasanya dimulai pada bulan November karena pada bulan tersebut intensitas curah hujan meningkat. Musim kemarau yang panjang akan menyebabkan terjadinya penguapan air di permukaan tanah dalam jumlah besar. Hal ini mengakibatkan munculnya pori-pori atau rongga tanah hingga terjadi retakan dan merekahnya tanah permukaan. Ketika hujan, air akan menyusup ke bagian yang retak sehingga tanah dengan cepat mengembang kembali. Pada awal musim hujan, intensitas hujan yang tinggi biasanya sering terjadi, sehingga kandungan air pada tanah menjadi jenuh dalam waktu yang singkat. Hujan lebat pada awal musim dapat meimbulkan longsor karena melalui tanah yang merekah air akan masuk dan terakumulasi di bagian dasar lereng, sehingga menimbulkan gerakan lateral. Bila terdapat pepohonan di permukaannya, tanah longsor dapat dicegah karena air akan diserap oleh tumbuhan. Akar tumbuhan juga akan berfungsi untuk mengikat tanah.

b. Lereng terjal

Lereng atau tebing yang terjal akan memperbesar gaya pendorong. Lereng yang terjal terbentuk karena pengikisan air sungai, mata air, air laut, dan angin. Kebanyakan sudut lereng yang menyebabkan longsor adalah 180° apabila ujung lerengnya terjal dan bidang longsornya mendatar.

c. Tanah yang kurang padat dan tebal

Jenis tanah yang kurang padat adalah tanah lempung atau tanah liat dengan ketebalan lebih dari 2,5 m dari sudut lereng lebih dari 220° . Tanah jenis ini

memiliki potensi untuk terjadinya tanah longsor terutama bila terjadi hujan. selain itu tanah ini sangat rentan terhadap pergerakan tanah karena menjadi lembek terkena air dan pecah ketika hawa terlalu panas.

d. Batuan yang kurang kuat

Batuan endapan gunung api dan sedimen berukuran pasir dan campuran antara kerikil, pasir dan lempung umumnya kurang kuat. Batuan tersebut akan mudah menjadi tanah apabila mengalami proses pelapukan dan umumnya rentan terhadap tanah longsor bila terdapat pada lereng yang terjal.

e. Jenis tata lahan

Tanah longsor banyak terjadi di daerah tata lahan persawahan, perladangan dan adanya genangan air di lereng yang terjal. Pada lahan persawahan akarnya kurang kuat untuk mengikat butir tanah dan membuat tanah menjadi lembek dan jenuh dengan air sehingga mudah terjadi longsor. Sedangkan untuk daerah perladangan penyebabnya adalah karena akar pohonnya tidak dapat menembus bidang longsor yang dalam dan umumnya terjadi di daerah longsor lama.

f. Getaran

Getaran yang terjadi biasanya diakibatkan oleh gempa bumi, ledakan, getaran mesin, dan getaran lalu lintas kendaraan. Akibat yang ditimbulkan adalah tanah, badan jalan, lantai, dan dinding rumah menjadi retak.

g. Susut muka air danau atau bendungan

Akibat susutnya muka air yang cepat di danau maka gaya penahan lereng menjadi hilang. Dengan sudut kemiringan waduk, mudah terjadi longsor dan penurunan tanah yang biasanya diikuti oleh retakan.

h. Adanya beban tambahan

Adanya beban tambahan seperti beban bangunan pada lereng, dan di sekitar tikungan jalan pada daerah lembah. Akibatnya adalah terjadi penurunan tanah dan retakan yang arahnya ke arah lembah.

i. Pengikisan/ Erosi

Pengikisan banyak terjadi di air sungai yang mengarah ke tebing. Selain itu akibat penggundulan hutan di sekitar tikungan sungai, tebing akan menjadi terjal.

j. Adanya material timbunan pada tebing

Untuk mengembangkan dan memperluas lahan pemukiman umumnya dilakukan pemotongan tebing dan penimbunan lembah. Tanah timbunan pada lembah tersebut belum terdapatkan sempurna seperti tanah asli yang berada di bawahnya. Sehingga apabila hujan akan terjadi penurunan tanah yang diikuti dengan retakan tanah.

k. Longsoran lama

Longsoran lama umumnya terjadi selama dan setelah terjadinya pengendapan material gunung api pada lereng yang relatif terjal atau pada saat atau sesudah terjadi patahan kulit bumi. Bekas longsoran lama memiliki ciri: adanya tebing terjal yang panjang melengkung membentuk tapal kuda, umumnya dijumpai mata air, pepohonan yang relatif tebal karena tanahnya gembur dan subur, daerah badan longsor bagian atas umumnya relatif landai, dijumpai longsor kecil terutama pada tebing lembah, dijumpai longsoran lama, dijumpai alur lembah dan pada tebingnya dijumpai retakan dan longsoran kecil, longsoran ini cukup luas.

1) Adanya bidang diskontinuitas (bidang tidak sinambung)

Bidang tidak sinambung ini memiliki ciri : bidang pelapisan batuan, bidang kontak antara tanah penutup dengan batuan dasar, bidang kontak antara batuan yang retak dengan batuan yang kuat, bidang kontak antara batuan yang dapat melewatkan air dengan batuan yang tidak melewatkan air, dan bidang kontak antara tanah yang lembek dengan tanah yang padat. Bidang-bidang tersebut merupakan bidang bidang lemah dan dapat berfungsi sebagai bidang luncuran tanah longsor.

2) Penggundulan hutan

Tanah longsor umumnya banyak terjadi di daerah yang relatif gundul dimana pengikatan air tanah sangat kurang.

3) Daerah pembuangan sampah

Penggunaan lapisan tanah yang rendah untuk pembuangan sampah dalam jumlah banyak dapat mengakibatkan tanah longsor ditambah dengan guyuran hujan, seperti yang terjadi di TPAS Leuwigajah di Cimahi. Bencana ini menyebabkan sekitar 120 orang lebih meninggal.

2. Bencana Tanah Longsor di Indonesia

Bencana tanah longsor telah banyak terjadi di Indonesia. Di daerah-daerah tertentu, frekuensi terjadinya bencana tanah longsor dari tahun ke tahun semakin meningkat. Bahkan, di tingkat nasional, jumlah kejadian dan korban yang ditimbulkan cenderung meningkat.

Kondisi iklim indonesia yang berupa tropis basah dan letak geografis Indonesia yang berada di jalur gunung api yang sering menimbulkan gempa bumi,

sehingga menyebabkan potensi tanah longsor menjadi tinggi. Hal ini ditunjang dengan adanya degradasi perubahan tataguna lahan akhir-akhir ini, menyebabkan bencana tanah longsor yang semakin meningkat.

Selama ini dalam pembuatan Rencana Tata Ruang Wilayah serta perencanaan pembangunan daerah jarang yang memperhatikan adanya faktor ancaman longsor. Konsekuensinya adalah dampak yang terjadi akan terus berjatuhan apabila tidak dilakukan tindakan nyata mengurangi risiko bencana tanah longsor.

Salah satu kasus bencana longsor besar yang pernah terjadi di Indonesia adalah peristiwa bencana tanah longsor yang terjadi di Karanganyar, yaitu pada tanggal 26 Desember 2007. Bencana tanah longsor tersebut terjadi di 14 kecamatan di Kabupaten Karanganyar yang menelan korban jiwa 62 meninggal.

B. *Neural Network* (NN)

Model untuk menjelaskan hubungan non linear telah berkembang pesat hingga kini. Salah satu model tersebut adalah *Neural Network* (NN). Model NN adalah model yang didesain untuk memodelkan bentuk arsitektur syaraf pada otak manusia. Telah banyak dilakukan penelitian dengan menggunakan model NN. Hal ini didorong oleh adanya kemungkinan untuk menggunakan NN sebagai instrumen untuk menyelesaikan berbagai permasalahan aplikasi seperti *pattern recognition*, *signal processing*, *processing control* dan peramalan. NN terdiri dari elemen sederhana yang beroperasi secara paralel dan terinspirasi oleh sistem saraf biologis. Seperti di alam, fungsi jaringan ditentukan terutama oleh hubungan antar elemen, dalam NN elemen ini disebut neuron. Umumnya NN disesuaikan, atau dilatih, sehingga masukan tertentu mengarah ke target *output* tertentu. NN berawal

dari memodelkan otak manusia dengan cara berbeda dari *computer digital* konvensional.

Neuron merupakan bagian penting dalam terbentuknya suatu jaringan syaraf. Neuron sendiri terdiri dari tiga bagian, yaitu fungsi penjumlahan (*summing function*), fungsi aktivasi (*activation function*), dan keluaran (*output*).

Informasi (*input*) akan dikirim ke *neuron* dengan bobot tertentu. *Input* ini akan diproses oleh suatu fungsi yang akan menjumlahkan nilai bobot yang ada. Hasil penjumlahan kemudian akan dibandingkan dengan suatu nilai ambang (*threshold*) tertentu melalui fungsi aktivasi setiap *neuron*. Apabila *input* tersebut melewati suatu nilai ambang tertentu, maka *neuron* tersebut akan diaktifkan. Namun apabila *input* tersebut tidak melewati nilai ambang batas tertentu, maka *neuron* tidak akan diaktifkan. Apabila *neuron* tersebut diaktifkan, maka *neuron* tersebut akan mengirimkan *output* melalui bobot-bobot outputnya ke semua *neuron* yang berhubungan dengannya.

Sehingga dapat disimpulkan bahwa *neuron* terdiri dari 3 elemen pembentuk yaitu:

- 1) Himpunan unit-unit yang dihubungkan dengan jalur koneksi. Jalur-jalur tersebut memiliki bobot yang berbeda-beda. Bobot yang bernilai positif akan memperkuat sinyal yang dibawa. Jumlah, struktur, pola hubungan antar unit-unit tersebut akan menentukan arsitektur jaringan.
- 2) Suatu unit penjumlahan yang akan menjumlahkan masukan-masukan sinyal yang sudah dikalikan dengan bobotnya.

- 3) Fungsi aktivasi yang akan menentukan apakah sinyal dari *input* neuron akan diteruskan ke neuron lain atau tidak.

Umumnya, jika menggunakan *Neural Network*, hubungan *input* dan *output* harus diketahui secara pasti dan jika hubungan tersebut telah diketahui maka dapat dibuat suatu model. Proses pembelajaran (*learning*) NN dimulai dengan memasukkan informasi yang sebelumnya telah diketahui hasil kebenarannya. Pemasukan informasi ini dilakukan melalui unit-unit *input*. Bobot-bobot antar koneksi dalam suatu arsitektur diberi nilai awal dan kemudian NN dijalankan. Bagi jaringan sendiri, bobot-bobot ini digunakan untuk pembelajaran dan mengingat suatu informasi yang telah ada. Pengaturan bobot dilakukan secara terus menerus dan dengan menggunakan kriteria tertentu sampai diperoleh hasil yang sesuai dengan kriteria yang telah ditentukan.

Lapisan lapisan penyusun NN dibagi menjadi 3 yaitu (Siang, 2005: 24):

- 1) Lapisan *input* (*Input Layer*)

Node-node di dalam lapisan *input* disebut neuron-neuron *input*. Neuron-neuron *input* menerima *input* dari luar, *input* yang diberikan merupakan penggambaran suatu permasalahan.

- 2) Lapisan tersembunyi (*Hidden Layer*)

Node di dalam lapisan tersembunyi disebut neuron tersembunyi. *Output* dari lapisan ini tidak dapat diamati secara langsung.

- 3) Lapisan *output* (*Output Layer*)

Node-node di dalam lapisan *output* disebut neuron-neuron *output*. *Output* dari lapisan ini merupakan hasil dari NN terhadap suatu permasalahan.

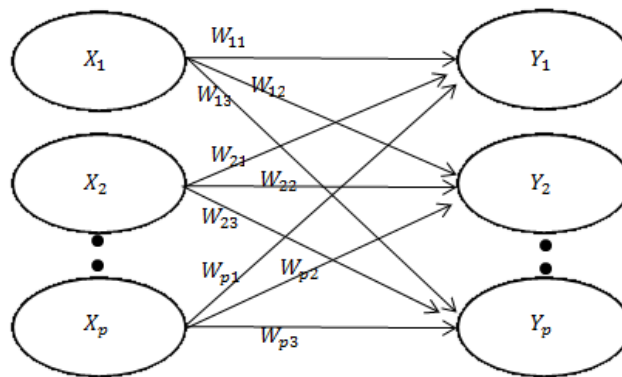
1. Arsitektur Jaringan

Model-model NN ditentukan oleh arsitektur jaringan serta algoritma pelatihan. Arsitektur biasanya menjelaskan arah perjalanan sinyal atau data di dalam jaringan, sedangkan algoritma belajar menjelaskan bagaimana bobot koneksi harus diubah agar pasangan *input-output* yang diinginkan dapat tercapai. Perubahan bobot koneksi dapat dilakukan dengan berbagai cara, tergantung pada jenis algoritma pelatihan yang digunakan. Dengan mengatur besarnya nilai bobot ini diharapkan bahwa kinerja jaringan dalam mempelajari berbagai macam pola yang dinyatakan oleh setiap pasangan *input-output* akan meningkat.

Beberapa arsitektur jaringan yang sering dipakai dalam NN antara lain (Fausett, 1994: 12-15) :

a. Jaringan Layar Tunggal (*Single-Layer Network*)

Jaringan layar tunggal (*single-layer network*) adalah jaringan yang menghubungkan langsung neuron pada input layer dengan neuron pada output layer dengan bobot yang berbeda-beda. Gambar 2.1 adalah contoh jaringan syaraf dengan lapisan tunggal.

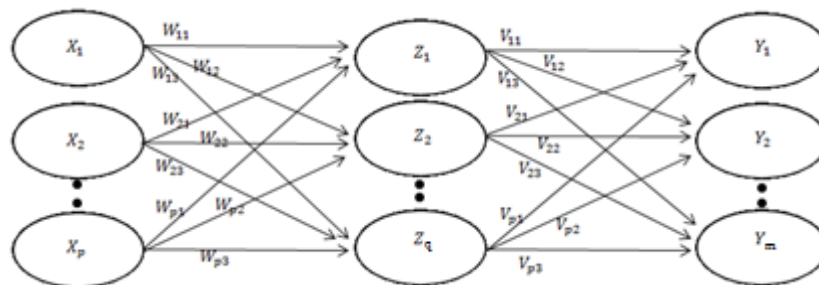


Gambar 2.1 Arsitektur Jaringan Syaraf dengan Layer Tunggal

Pada gambar 2.1, terlihat bahwa jaringan tersebut lapisan input memiliki p neuron yang ditunjukkan oleh notasi x_1, x_2 , sampai dengan x_p . Sedangkan pada lapisan output memiliki p neuron yang ditunjukkan oleh notasi y_1, y_2 , sampai dengan y_p . Setiap neuron pada lapisan input dan lapisan output dihubungkan oleh bobot yang bersesuaian (w).

b. Jaringan Lapisan Jamak (*Multi-Layer Network*)

Jaringan lapisan jamak (*multi-layer network*) adalah jaringan yang lebih kompleks yang terdiri dari *input layer*, beberapa *hidden layer* dan *output layer*. Gambar 2.2 merupakan contoh arsitektur jaringan dengan banyak lapisan (*Multi-layer Net*).



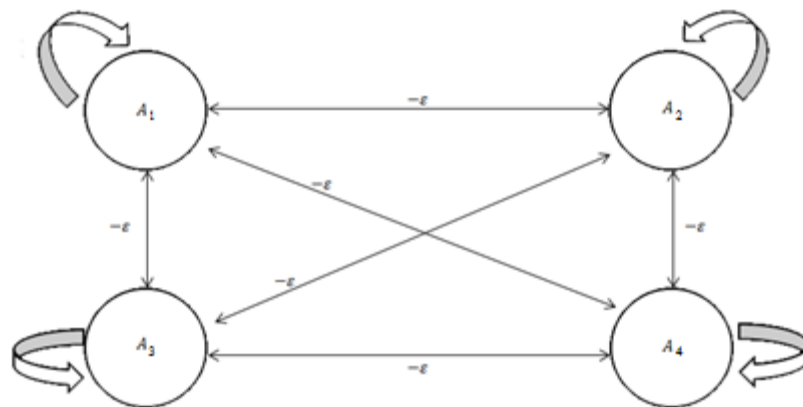
Gambar 2.2 Arsitektur Jaringan Syaraf dengan Layer Jamak

Pada gambar 2.2, terdapat lapisan *input* dengan banyak j neuron *input* ($x_j, j = 1, 2, 3, \dots, p$). Lapisan tersembunyi ada satu dengan banyaknya neuron tersembunyi ($z_k, k = 1, 2, 3, \dots, q$). Dan lapisan *output* dengan banyaknya neuron ($y_l, l = 1, 2, 3, \dots, m$). Bobot-bobot yang menghubungkan neuron *input* ke- j menuju neuron ke- k pada lapisan tersembunyi disimbolkan dengan w_{jk} sedangkan

v_{kl} adalah bobot-bobot dari neuron ke- k pada lapisan tersembunyi yang menuju pada lapisan *output* ke- l .

c. Jaringan Syaraf dengan Lapisan Kompetitif

Bentuk arsitektur jaringan syaraf dengan lapisan kompetitif memiliki bentuk yang berbeda karena antar neuron pada jaringan ini dapat saling dihubungkan. Jaringan ini memetakan pola *input* secara tepat. Namun, apabila terlalu banyak membuat NN hanya mampu mengingat set data yang diberikan pada saat proses pelatihan saja. Sedangkan apabila diberikan data *input* yang baru maka jaringan tidak mampu mengeluarkan input yang benar. Pada jaringan ini sekumpulan neuron akan bersaing untuk dapat menjadi neuron aktif. Umumnya hubungan antara neuron pada lapisan kompetitif tidak diperlihatkan pada diagram arsitektur. Gambar 3 merupakan salah satu contoh arsitektur jaringan syaraf dengan lapisan kompetitif dengan koneksi dari lapisan tersebut memiliki bobot $-\epsilon$.



Gambar 2.3 Arsitektur Jaringan Syaraf dengan Lapisan Kompetitif

Setiap jaringan memuat banyak jalur koneksi yang menghubungkan neuron dalam tiap *layer*. Penghubung ini memiliki bobot (*weight*) yang memfasilitasi

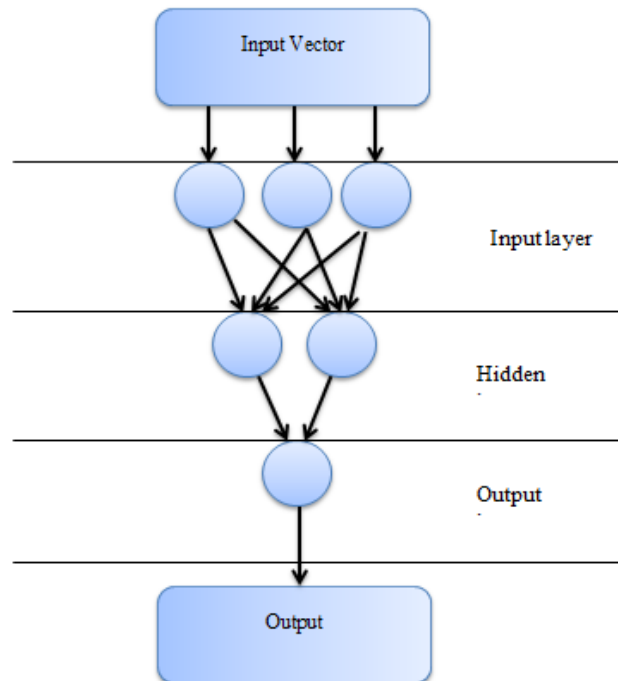
pertukaran informasi antar neuron (A_i). Metode ini digunakan untuk menentukan bobot koneksi tersebut dinamakan algoritma pelatihan (*training*). Setiap neuron mempunyai tingkat aktivasi yang merupakan fungsi dari input yang masuk padanya. Aktivasi yang dikirim suatu neuron ke neuron yang lain berupa sinyal dan hanya dapat dikirim sekali dalam satu waktu, meskipun sinyal tersebut disebarkan pada beberapa neuron yang lain.

Berdasarkan dari pola koneksi, *Neural Network* dapat dibagi ke dalam dua kategori:

a. Struktur *feedforward*

Sebuah jaringan yang sederhana mempunyai struktur *feedforward* dimana *signal* bergerak dari *input* kemudian melewati *hidden layer* dan akhirnya mencapai unit *output* (mempunyai struktur perilaku yang stabil).

Tipe jaringan *feedforward* mempunyai sel syaraf yang tersusun dari beberapa lapisan. Lapisan *input* bukan merupakan sel syaraf. Lapisan ini hanya memberi pelayanan dengan mengenalkan suatu nilai dari suatu variabel. *Hidden layer* dan lapisan *output* sel syaraf terhubung satu sama lain dengan lapisan sebelumnya. Kemungkinan yang timbul adalah adanya hubungan dengan beberapa unit dari lapisan sebelumnya atau terhubung semuanya.

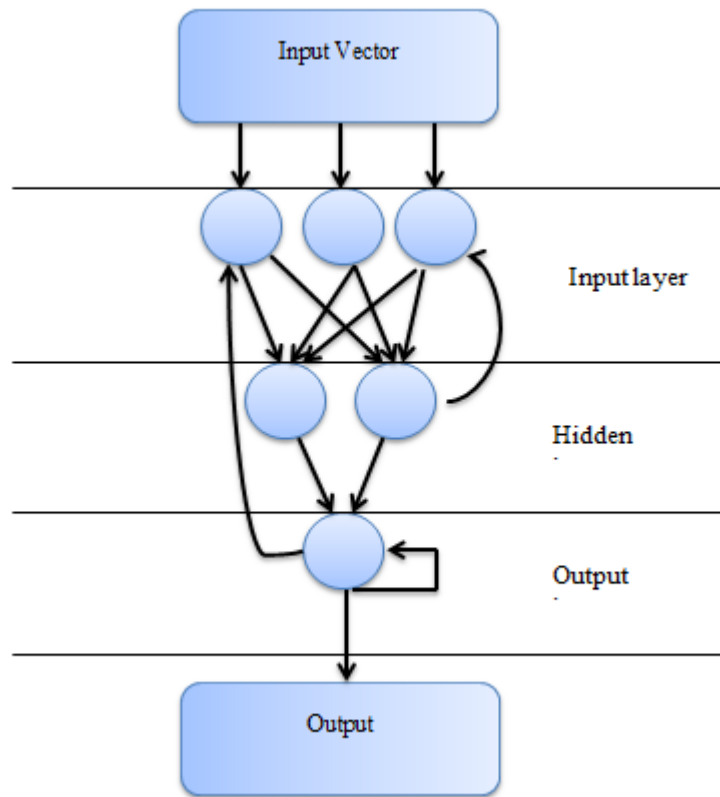


Gambar 2.4 Jaringan Syaraf Tiruan *Feedforward*

Yang termasuk dalam struktur *feedforward* antara lain *single layer perceptron*, *multilayer perceptron*, *radial basis function network*, *higher order network*, dan *polynomial learning networks*.

b. Struktur *recurrent (feedback)*

Suatu jaringan *recurrent* adalah jaringan yang mempunyai koneksi kembali dari *output* ke *input* (dari satu layer ke layer yang lain) seperti pada gambar 2.5, hal ini menimbulkan ketidakstabilan dan dinamika yang sangat kompleks pada jaringan tersebut.



Gambar 2.5 Jaringan Syaraf Tiruan *Recurrent*

Dengan *feedback* pada struktur *recurrent* dapat mempercepat proses iterasi. Adanya proses iterasi yang lebih cepat akan membuat *update* parameter dan kecepatan konvergensi menjadi lebih cepat.

2. Fungsi Aktivasi

Fungsi aktivasi mendefinisikan nilai *output* dari suatu neuron dalam level aktivasi tertentu berdasarkan nilai output pengombinasi linear. Jika

$$net = \sum_{i=1}^n x_i w_{ij} \tag{2.1}$$

maka fungsi aktivasinya adalah

$$y = f(\text{net}) = f\left(\sum_{i=1}^n x_i w_{ij}\right) \quad (2.2)$$

dengan net adalah kombinasi linear, x_i menunjukkan vektor input ke-i, w_{ij} menunjukkan bobot-bobot dari vektor input ke-i menuju output ke-j, dan n menunjukkan dimensi vektor *input*

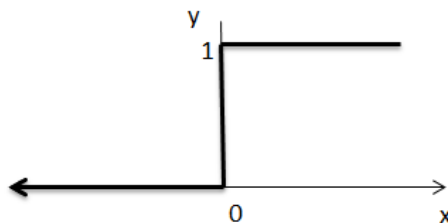
Ada beberapa macam fungsi aktivasi yang digunakan pada jaringan syaraf tiruan, antara lain :

a. Fungsi Undak Biner

Jaringan dengan lapisan tunggal sering menggunakan fungsi undak (*step function*) untuk mengonversikan input dari suatu variabel yang bernilai kontinu ke suatu output biner, yaitu 0 atau 1 (Gambar 2.6).

Fungsi undak biner (*Hard Limit*) dirumuskan sebagai berikut

$$y = \begin{cases} 0, & \text{jika } x \leq 0 \\ 1, & \text{jika } x > 0 \end{cases}$$

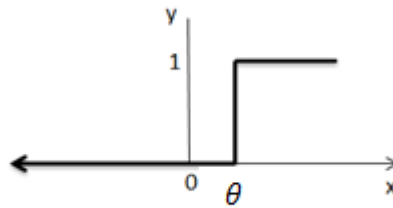


Gambar 2.6 Fungsi Aktivasi: Undak Biner (*Hard Limit*)

b. Fungsi *Threshold*

Fungsi *threshold* merupakan fungsi undak biner yang menggunakan nilai ambang. Fungsi *threshold* dengan nilai ambang θ dirumuskan sebagai :

$$y = \begin{cases} 0, & \text{jika } x \leq \theta \\ 1, & \text{jika } x > \theta \end{cases}$$



Gambar 2.7 Fungsi Aktivasi *Threshold*

3. Algoritma pembelajaran

Pembelajaran dalam NN didefinisikan sebagai suatu proses dimana parameter-parameter bebas NN diadaptasi melalui suatu proses perangsangan berkelanjutan oleh lingkungan dimana jaringan berada. Proses pembelajaran merupakan bagian penting dari konsep NN.

Proses pembelajaran bertujuan untuk melakukan pengaturan terhadap bobot yang ada pada NN, sehingga diperoleh bobot akhir yang tepat sesuai dengan pola data yang dilatih (Sri Kusumadewi & Sri Hartati, 2010:84). Pada proses pembelajaran akan terjadi perbaikan bobot-bobot berdasarkan algoritma tertentu. Nilai bobot akan naik jika informasi yang diberikan ke suatu neuron mampu tersampaikan ke neuron yang lain. Sebaliknya, nilai bobot akan berkurang jika informasi yang diberikan ke neuron tidak tersampaikan ke neuron yang lainnya. Terdapat 2 metode pembelajaran NN, yaitu (Fausett, 1994: 15):

i. Pembelajaran Terawasi (*Supervised Learning*)

Metode pembelajaran pada NN disebut terawasi jika *output* yang telah diketahui sebelumnya. Tujuan pembelajaran terawasi adalah untuk mempredksi satu atau lebih variabel target dari satu atau lebih variabel *input*. Pada proses pembelajaran, satu pola *input* akan diberikan ke stau *neuron* pada lapisan *input*. Selanjutnya pola akan dirambatkan sepanjang NN hingga sampai ke neuron *output*. Lapisan *ouput* akan membangkitkan pola *output* yang akan dicocokkan dengan pola *output* targetnya. *Error* muncul apabila terdapat perbedaan antara pola *output* hasil pembelajaran dengan pola target sehingga diperlukan pembelajaran lagi.

ii. Pembelajaran Tak Terawasi (*Unsupervised Learning*)

Pembelajaran tak terawasi tidak memerlukan target *output* dan jaringan dapat melakukan *training* sendiri untuk mengestrak fitur dari variabel independen. Pada metode ini, tidak dapat ditentukan hasil *output*nya. Selama proses pembelajaran, nilai bobot disusun dalam suatu range tertentu sesai dengan nilai *input* yang diberikan. Tujuan pembelajaran ini adalah untuk mengelompokkan unit-unit yang hampir sama ke suatu area tertentu.

4. Fungsi Transfer

Karakter dari *neural network* tergantung atas bobot dan fungsi transfer yang mempunyai ciri tertentu untuk setiap unit. Fungsi ini terdiri dari 3 kategori yaitu :

- a. Untuk *linear units*, aktivitas *output* adalah sebanding dengan jumlah bobot *output*.

- b. Untuk *threshold units*, *output* diatur satu dari dua tingkatan tergantung dari apakah jumlah *input* adalah lebih besar atau lebih kecil dari nilai ambang.
- c. Untuk *sigmoid units*, *output* terus menerus berubah-ubah tetapi tidak berbentuk linear. Unit ini mengandung kesamaan yang lebih besar dari sel saraf sebenarnya dibandingkan dengan linear dan *threshold* unit, namun ketiganya harus dipertimbangkan dengan perkiraan kasar.

Fungsi transfer dibutuhkan untuk membuat *neural network* melakukan beberapa kerja khusus. Maka dari itu perlu dipilih bagaimana unit-unit dihubungkan antara satu dengan yang lain dan mengatur bobot dari hubungan tersebut secara cepat. Hubungan tersebut menentukan apakah mungkin suatu unit mempengaruhi unit yang lain. Bobot menentukan kekuatan dari pengaruh tersebut.

Dapat dilakukan pembelajaran terhadap 3 lapisan pada *neural network* untuk melakukan kerja khusus dengan menggunakan prosedur dibawah ini :

1. Memperkenalkan *neural network* dengan contoh pembelajaran yang terdiri dari sebuah pola dari aktifitas untuk unit-unit *input* bersama dengan pola yang diharapkan dari aktifitas untuk unit-unit *output*.
2. Menentukan seberapa dekat output sebenarnya dari *neural network* sesuai dengan *output* yang diharapkan.
3. Mengubah bobot-bobot setiap hubungan agar *neural network* menghasilkan suatu perkiraan yang lebih baik dari *output* yang diharapkan.

C. Clustering

Clustering merupakan salah satu metode dalam *data mining* yaitu teknik pengelompokkan data, pengamatan atau memperhatikan dan membentuk kelas

obyek yang memiliki kemiripan. *Clustering* tidak mempunyai target *output*. Pada metode ini tidak dapat ditentukan hasil output selama proses pembelajaran. Selama proses pembelajaran, nilai bobot disusun dalam suatu *range* tertentu tergantung pada nilai *input* yang diberikan. Tujuan pembelajaran ini adalah mengelompokkan unit-unit yang hampir sama dalam suatu area tertentu. *clustering* berbeda dengan klasifikasi, dalam hal tidak ada variabel target untuk *clustering*. *Clustering* tidak mengklasifikasikan, meramalkan, atau memprediksi nilai dari sebuah variabel target dan digunakan ketika kita tidak mengetahui bagaimana data harus dikelompokkan (Susanto&Suryani, 2010).

D. Euclidian Distance

Euclidian Distance dianggap sebagai *distance matrix* yang mengadopsi prinsip Phytagoras. Hal ini dikarenakan pola perhitungannya yang menggunakan aturan pangkat dan akar kuadrat. Euclidian akan memberikan hasil jarak yang relatif kecil (Davies & Bouldin, 1979:224). Jarak antara nilai random atau bobot dan data dihitung dengan menggunakan rumus persamaan 2.3 berikut :

$$D_{ij} = \sum_{i=1}^n (w_{ij} - x_i)^2 \quad (2.3)$$

dimana D_{ij} adalah jarak Euclidian dari input vektor ke-i dengan neuron ke-j, w_{ij} adalah bobot penghubung input vektor ke-j dengan neuron ke-j (bobot akhir), x_i adalah input vector ke-i, dan n adalah jumlah dimensi vektor input.

E. Davies Bouldin Index

Davies Bouldin Indeks merupakan salah satu metode validasi *cluster* untuk evaluasi kuantitatif dari hasil *clustering*. Pengukuran ini bertujuan meminimumkan jarak *inter-cluster* dan memaksimalkan jarak *intra-cluster* (Saitta&Smith, 2007). Dalam penelitian ini *Davies Bouldin Indeks* akan digunakan untuk mendeteksi *outlier* pada masing-masing *cluster* yang terbentuk .

1) *Variance* dari *cluster*

$$var(x) = \frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})^2 \quad (2.4)$$

2) Jarak antar *cluster*

$$R_{ij} = \frac{var(C_i) + var(C_j)}{\|c_i - c_j\|} \quad (2.5)$$

3) Menentukan jarak *inter-cluster* maksimum

$$R_i = \max R_{ij} \quad (2.6)$$
$$j = 1, \dots, k, i \neq j$$

4) Validasi *Davies Bouldin*

$$DB = \frac{1}{k} \sum_{i=1}^k R_i \quad (2.7)$$

dengan

$var(x)$: variansi dari data dalam satu *cluster*

N : banyaknya data dalam satu *cluster*

x_i : data ke- i dalam satu *cluster*

c_i : *centroid* dari *cluster* ke- i yang dinyatakan dengan rata-rata *cluster*

$var(C_i)$: <i>variance</i> dari <i>cluster-i</i>
\bar{x}	: rata-rata jarak dalam satu <i>cluster</i>
R_{ij}	: jarak <i>intra-cluster</i> antara <i>cluster-i</i> dengan <i>cluster-j</i>
R_i	: jarak <i>intra-cluster</i> maksimum
DB	: nilai <i>Davies Bouldin Index</i>