PERTEMUAN 11

Neural Network

&

Self Organizing Map (SOM)

Copyright © September 2022

Neural Network adalah salah satu algoritma *supervised learning* yang populer dan bisa juga digunakan untuk *semi-supervised* atau *unsupervised learning*

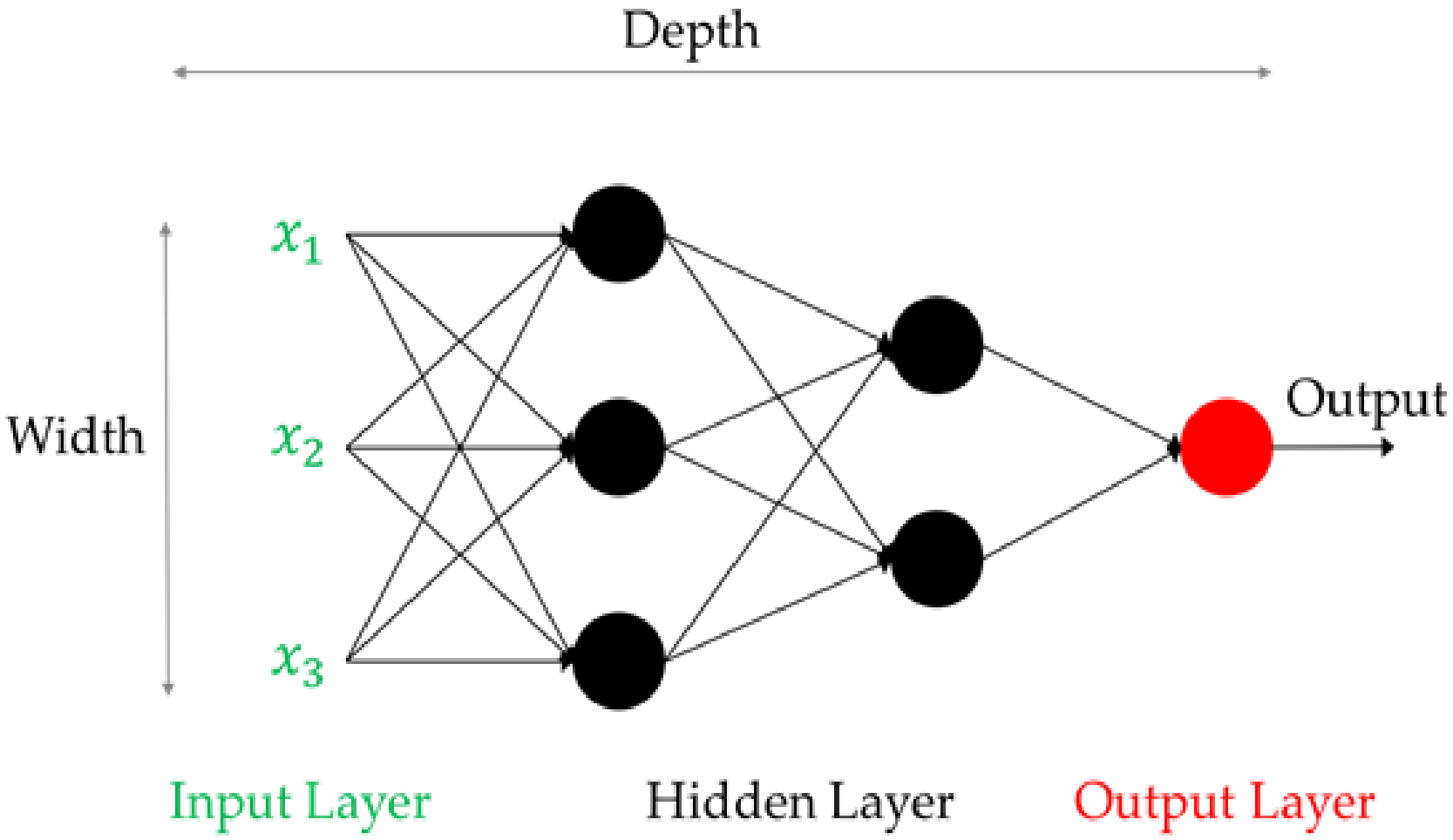
 Copyright © September 2022

Artificial Neural Network (ANN), menghasilkan model yang sulit dibaca dan dimengerti oleh manusia karena memiliki banyak layer (kecuali *single perceptron*) dan sifat **non-linear** (merujuk pada fungsi aktivasi).

Secara matematis, ANN ibarat sebuah graf, ANN memiliki neuron/*node* (*vertex*), dan sinapsis (*edge*). Topologi ANN akan dibahas lebih detil subbab berikutnya. Karena memiliki struktur seperti graf

operasi pada ANN mudah dijelaskan dalam notasi aljabar linear

Copyright © September 2022

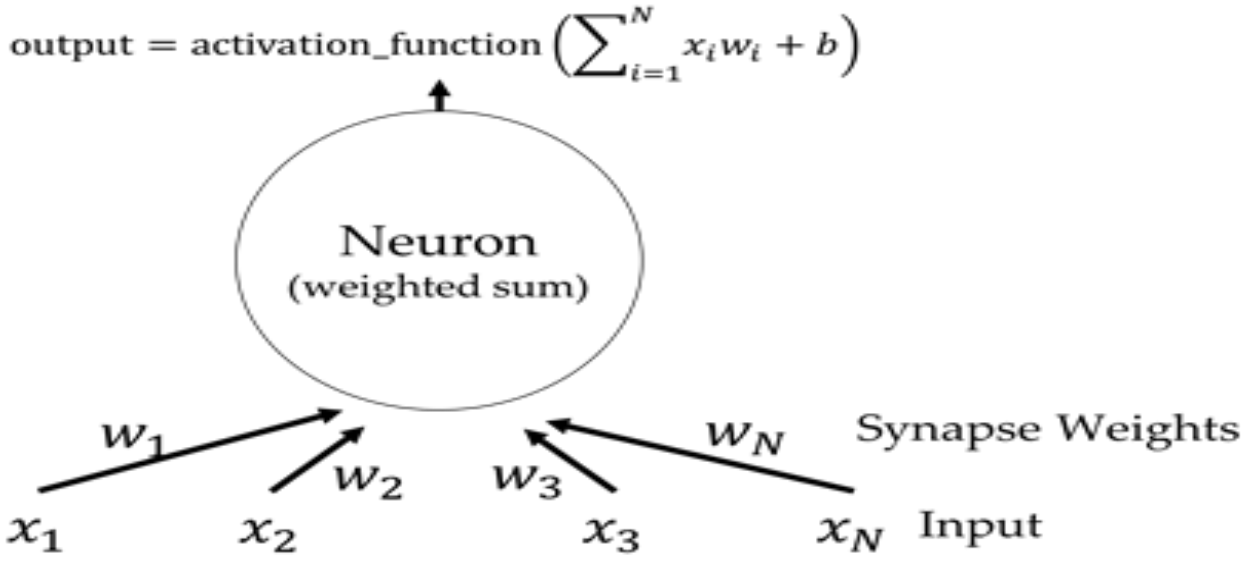


 Copyright © September 2022

Single Perceptron

Bentuk terkecil (minimal) sebuah ANN adalah *single perceptron* yang hanya terdiri dari sebuah neuron

Sebuah neuron diilustrasikan pada gambar dibawah ini :



 Copyright © September 2022

Secara matematis, terdapat *feature vector* **x** yang menjadi *input* bagi neuron tersebut. Ingat kembali, *feature vector* merepresentasikan suatu *data point*, *event* atau instans. Neuron akan memproses *input* **x** melalui perhitungan

Copyright © September 2022

Untuk melakukan pembelajaran *single perceptron*, *training* dilakukan meng- gunakan ***perceptron training rule***. Prosesnya sebagai berikut

1. Inisiasi nilai *synapse weights*, bisa *random* ataupun dengan aturan ter- tentu. Untuk heuristik aturan inisiasi, ada baiknya membaca buku refer- ensi .
2. Lewatkan input pada neuron, kemudian kita akan mendapatkan nilai *out- put*. Kegiatan ini disebut ***feedforward***

 Copyright © September 2022

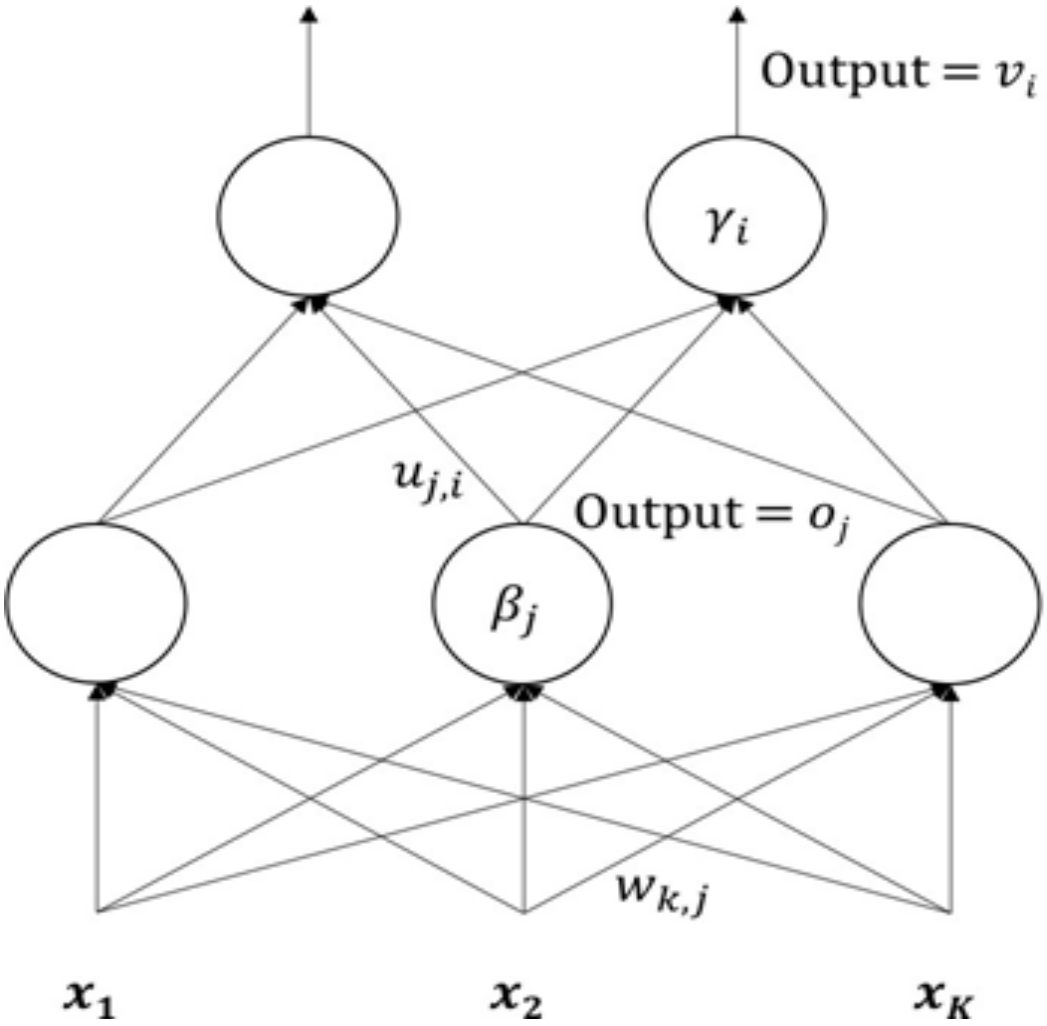
3. Nilai output (actual output) tersebut dibandingkan dengan desired output.
4. Apabila nilai output sesuai dengan desired output, tidak perlu mengubah apa-apa.
5. Apabila nilai output tidak sesuai dengan desired output, hitung nilai error (loss) kemudian lakukan perubahan terhadap learning parameter (synapse weight).
6. Ulangi langkah-langkah ini sampai tidak ada perubahan nilai error, nilai error kurang dari sama dengan suatu threshold (biasanya mendekati 0), atau sudah mengulangi proses latihan sebanyak T kali (threshold).

Copyright © September 2022

Multilayer Perceptron

multilayer perceptron secara lit- eral memiliki beberapa *layers*. Pada *lecture note* ini, secara umum ada tiga *layers*: *input*, *hidden*, dan *output layer*. *Input layer* menerima *input* (tanpa melakukan operasi apapun), kemudian nilai *input* (tanpa dilewatkan ke fungsi aktivasi) diberikan ke *hidden units* Pada *hidden units*, *in- put* diproses dan dilakukan perhitungan hasil fungsi aktivasi untuk tiap-tiap neuron, lalu hasilnya diberikan ke *layer* berikutnya *Out- put* dari *input layer* akan diterima sebagai input bagi *hidden layer*. Begit- upula seterusnya *hidden layer* akan mengirimkan hasilnya untuk *output layer*. Kegiatan ini dinamakan ***feed forward***

Copyright © September 2022

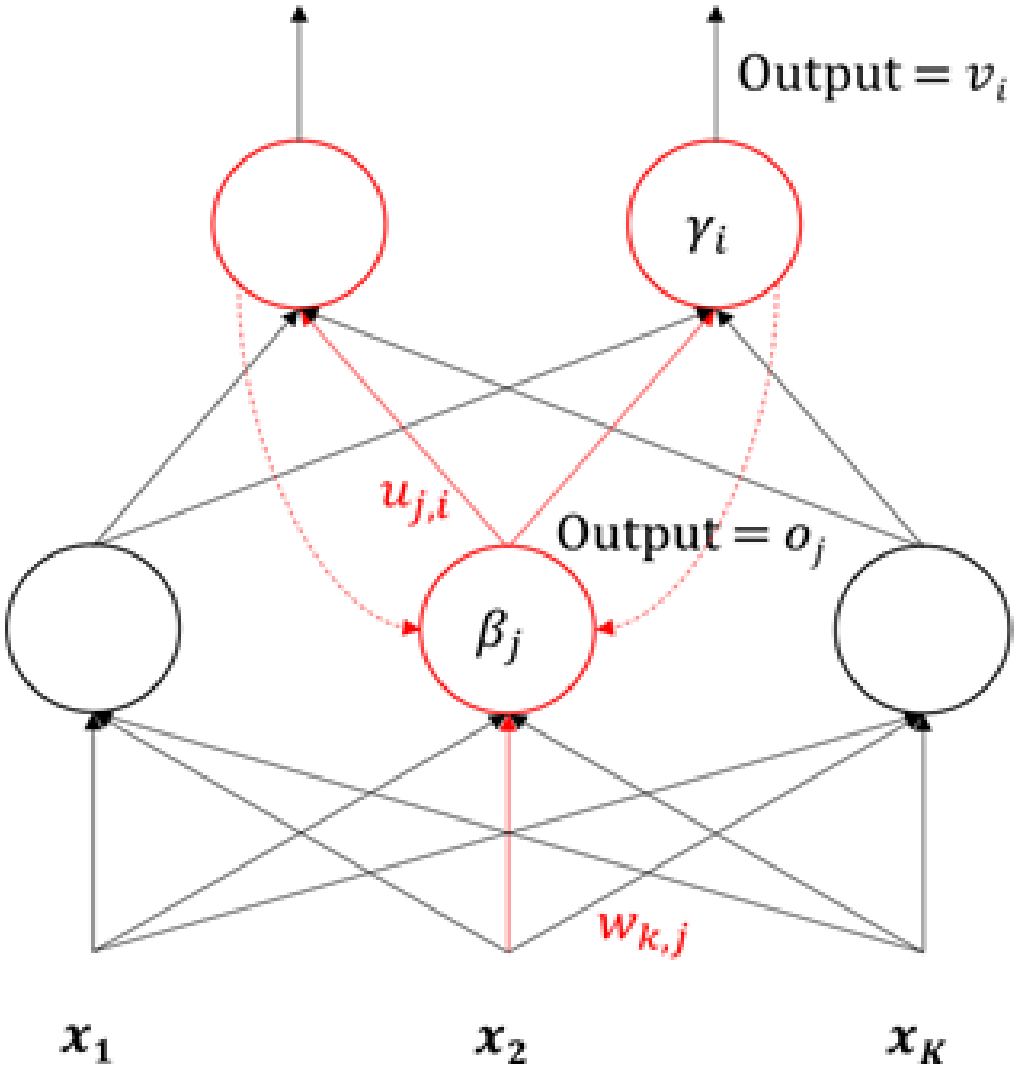


Output Units (I units)

Hidden Units (J units)

Input Units (K units)

Copyright © September 2022



Output Units (I units)

Hidden Units (J units)

Input Units (K units)

 Copyright © September 2022

Self Organizing Map (SOM)

 Copyright © September 2022

Pengertian

SOM merupakan salah satu teknik dalam Neural Network yang bertujuan untuk melakukan visualisasi data dengan cara mengurangi dimensi data melalui penggunaan self-organizing neural networks sehingga manusia dapat mengerti high-dimensional data yang dipetakan dalam bentuk low-dimensional data

Self-Organizing Map (SOM) atau sering disebut topology-preserving map pertama kali diperkenalkan oleh Teuvo Kohonen pada tahun 1996

Copyright © September 2022

Metode pembelajaran yang digunakan SOM adalah tanpa bimbingan dari suatu data input-target atau unsupervised learning yang mengasumsikan sebuah topologi yang terstruktur menjadian unit-unit kelas/cluster (Kohonen, 1989 dan Fausett, 1993).

Copyright © September 2022

Pada algoritma SOM, vektor bobot untuk setiap unit cluster berfungsi sebagai contoh dari input pola yang terkait dengan cluster itu. Selama proses self-organizing, cluster satuan yang bobotnya sesuai dengan pola vektor input yang paling dekat (biasanya, kuadrat dari jarak Euclidean minimum) dipilih sebagai pemenang

Copyright © September 2022

Unit pemenang dan unit tetangganya (dalam pengertian topologi dari unit cluster) terus memperbarui bobot mereka (Fausett, 1993). Setiap output akan bereaksi terhadap pola input tertentu sehingga hasil Kohonen SOM akan menunjukkan adanya kesamaan ciri antar anggota dalam cluster yang sama.

Copyright © September 2022

Dalam jaringan SOM, neuron target tidak diletakkan dalam sebuah baris seperti layaknya model JST yang lain. Neuron target diletakkan dalam dua dimensi yang bentuk/topologinya dapat diatur. Topologi yang berbeda akan menghasilkan neuron sekitar neuron pemenang yang berbeda sehingga bobot yang dihaslkan juga akan berbeda

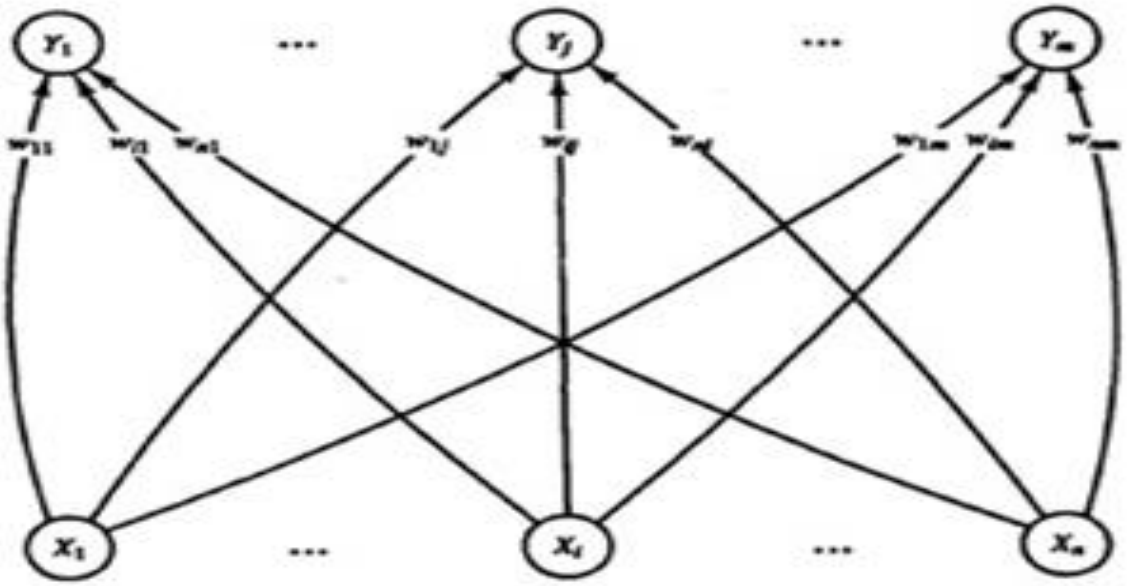
Copyright © September 2022

Pada SOM, perubahan bobot tidak hanya dilakukan pada bobot garis yang terhubung ke neuron pemenang saja, tetapi juga pada bobot garis ke neuron-neuron di sekitarnya. neuron di sekitar neuron pemenang ditentukan berdasarkan jaraknya dari neuron pemenang.

Copyright © September 2022

Arsitektur Topologi SOM

Arsitektur SOM merupakan jaringan yang terdiri dari dua lapisan (layer), yaitu lapisan input dan lapisan output. Setiap neuron dalam lapisan input terhubung dengan setiap neuron pada lapisan output. Setiap neuron dalam lapisan output merepresentasikan kelas (cluster)dari input yang diberikan.



Copyright © September 2022

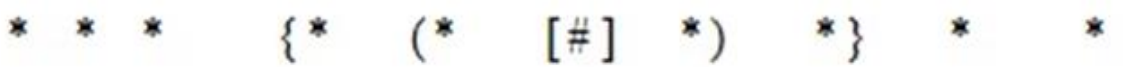
Topologi SOM

topologi, SOM memiliki 3 jenis topologi hubungan ketetanggaan (neighborhood) yaitu linear array, rectangular dan heksagonal grid

Copyright © September 2022

Topologi Linier

Topologi linear aray menunjukkan cluster unit yang tersusun secara linear. Cluster unit yang menjadi pemenang [#] memiliki dua unit tetangga (neighbour) yang berjarak 1 (R = 1), dan mempunyai dua unit tetangga yang berjarak 2 (R = 2).

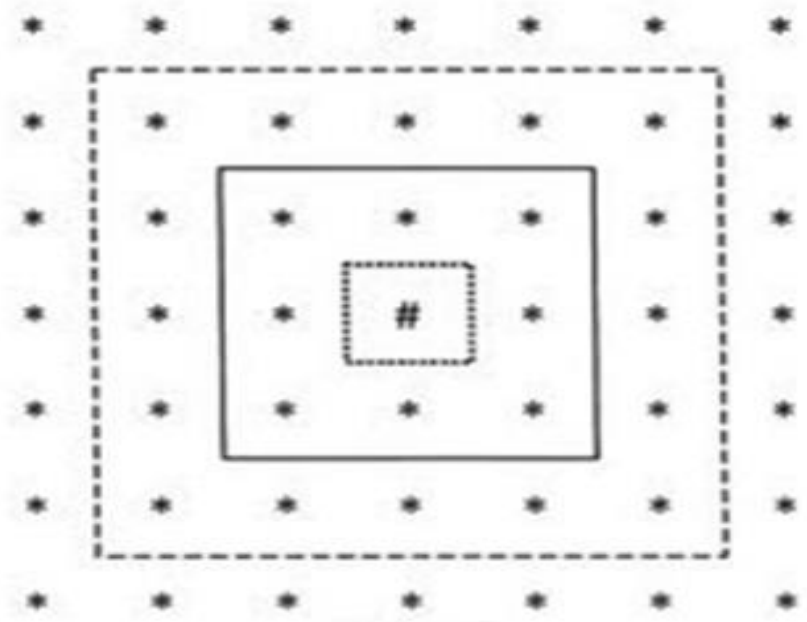


Keterangan : [ ] : R = 0 ; ( ) : R = 1 ; { } : R = 2

 Copyright © September 2022

Rectangular grid

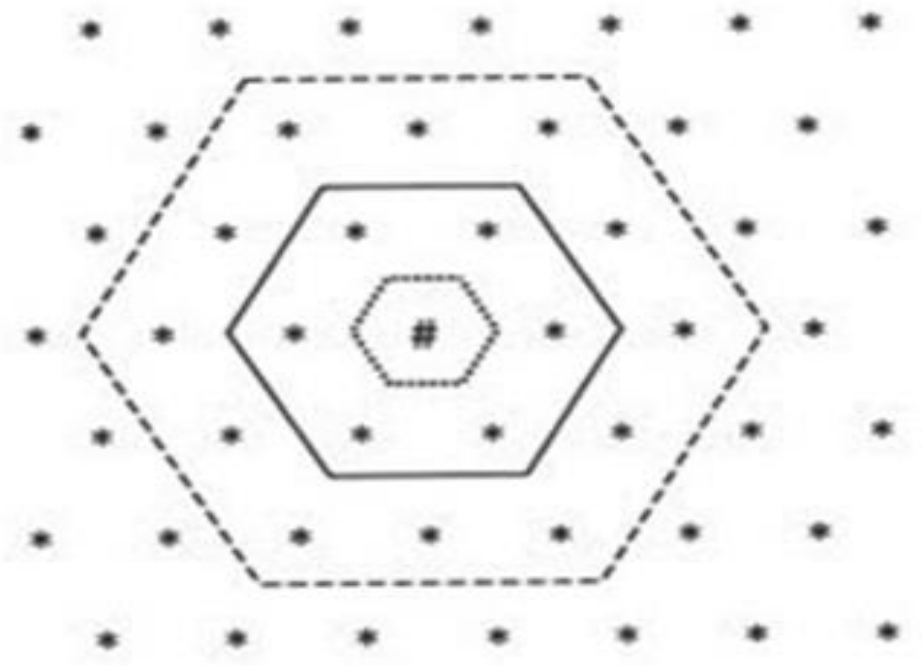
Rectangular grid adalah topologi dari cluster unit dua dimensi. Unit tetangga (neighbour) dari unit pemenang membentuk bujur sangkar. Unit pemenang [#] memiliki 8 neighbour berjarak 1 (R=1) dan 16 neighbour berjarak 2 (R=2).



 Copyright © September 2022

topologi heksagonal

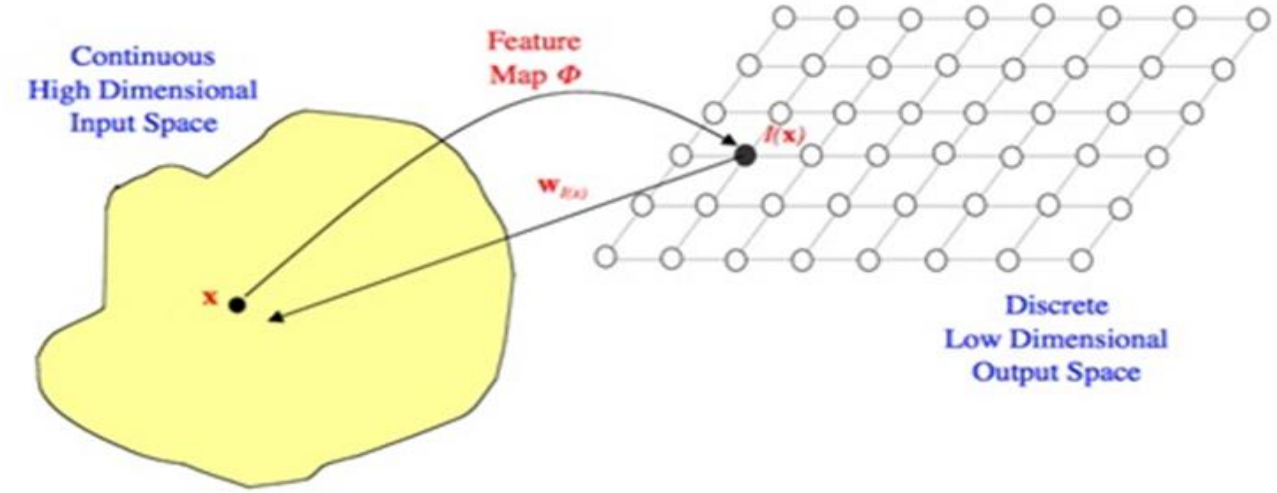
Dalam topologi heksagonal grid, unit tetangga (neighbour) yang berjarak 1 (R=1) dari unit pemenang adalah 6 dan yang berjarak 2 (R=2) adalah 12.



 Copyright © September 2022

cara kerja SOM

Secara umum, cara kerja SOM ditunjukkan oleh Gambar 5 dibawah ini:



Terdapat titik (x) pada ruang input untuk dipetakan ke titik I(x) pada ruang output. Setiap titik (I) dalam ruang output akan memetakan ke titik yang sesuai dalam ruang input melalui bobot wI(x).

Copyright © September 2022

Menurut Haykin (1999) terdapat tiga komponen penting dalam SOM yaitu:

1. Competition: Untuk setiap pola input, neuron menghitung nilai masing-masing fungsi diskriminan yang memberi dasar untuk kompetisi. Neuron tertentu dengan nilai terkecil dari fungsi diskriminan dinyatakan sebagai pemenang.
2. Cooperation: Neuron pemenang menentukan lokasi spasial dari lingkungan topologi excited neuron untuk memberi dasar kerjasama dalam suatu lingkungan neuron.
3. Synaptic Adaption: Excited neuron menurunkan nilai fungsi diskriminan yang berkaitan dengan pola input melalui penyesuaian bobot terkait sehingga respon dari neuron pemenang keaplikasi berikutnya dengan pola input yang sama akan meningkat.

 Copyright © September 2022

Algoritma SOM

Pengelompokkan data menggunakan algoritma SOM yang terdiri dari 4 tahapan yaitu:

1. Kompetisi: Setiap simpul output j, dihitung nilai $D(x, w_i)$ yang merupakan fungsi jarak Euclidian antara x dan w_j. Fungsi ini didefinisikan sebagai berikut:

$$D(x, w_m) = \sqrt{\sum_{i=1}^{n} (x_i - w_{mi})^2} \qquad (1)$$

dimana x adalah vektor dari *node input*

sedangkan w_m adalah vektor bobot dari node neuron ke-m.

 Copyright © September 2022

2. Update Bobot: setelah mendapat nilai jarak dari tiap-tiap vektor *input* kevektor bobot, pilih nilai jarak yang minimum sebagai neuron pemenang. Setiap neuron pemenang beserta tetangganya dilakukan proses adaptasi yaitu memperbaharui nilai bobot dimana h(t) adalah fungsi *node* tetangga (neighborhood function) dan t adalah banyaknya iterasi. Fungsi node tetangga yang digunakan adalah fungsi Gauss (Kohonen et al, 2001) dengan formula:

 Copyright © September 2022

$$h(t) = \alpha(t) \times e^{\left(-\frac{\|r_i - r_c\|^2}{2\delta^2(t)}\right)} \quad (3)$$

Dimana $\alpha(t)$ adalah nilai laju pembelajaran atau biasa disebut nilai alpha. Laju pembelajaran adalah fungsi penurunan tingkat pembelajaran seiring perubahan waktu (Fausett 1993).

$\|r_i - r_c\|^2$ adalah jarak kuadrat antara *neuron* ke-I dengan *neuron* pemenang dalam grid dan $\delta(t)$ adalah lebar tetangga. Nilai laju pembelajaran diperoleh dari:

$$\alpha(t) = \alpha_i\left(1 - \frac{t}{t_{max}}\right) \quad (4)$$

Dimana α_i adalah nilai awal laju pembelajaran dan t_{max} adalah iterasi maksimum.

Perubahan lebar tetangga didapat dari perhitungan berikut ini:

$$\delta(t) = \delta_i \left(\frac{\delta_f}{\delta_i}\right)^{\frac{t}{t_{max}}} \quad (7)$$

 Copyright © September 2022