

Penerapan Metode *Self-Organizing Map* (SOM) Untuk Visualisasi Data Geospasial Pada Informasi Sebaran Data Pemilih Tetap (DPT)

Yunus Anis^a, R.Rizal Isnanto^b

^a Universitas Stikubank Semarang

^b Program Studi Sistem Komputer Fakultas Teknik Universitas Diponegoro

Naskah Diterima : 18 Januari 2014; Diterima Publikasi : 21 Maret 2014

Abstract

A Self-Organizing Map (SOM) is a neural network method that has been introduced by Professor Teuvo Kohonen since 1980, as an artificial neural network topology without supervision (Unsupervised ANN), in which the learning process does not require supervision or target output. In this paper, the implementation methods of SOM is used to perform data clustering voters (DPT) in the General Election Governor level. The results of the clustering process then used to perform data classification air-geographical references (geo-referenced) that integrates visualization of the output space through a cartographic representation of the color settings, and explore the use of line width between the boundary-water element geographic reference, calculated according to the distance in the input space is best between locations. Clustering results is then used as the basis for determining the color criteria in the development of the Web-GIS application based on interval number of voters.

Keywords : Self-organizing map; Unsupervised ANN; Clustering; Geo-referenced; Cartographics representation

Abstrak

Metode *Self-Organizing Map* (SOM) merupakan suatu metode Jaringan Syaraf Tiruan yang diperkenalkan oleh Professor Teuvo Kohonen pada tahun 1980-an, sebagai salah satu bentuk topologi dari *Unsupervised Artificial Neural Network* (*Unsupervised ANN*) dimana dalam proses pelatihannya tidak memerlukan pengawasan atau target keluaran. Penerapan metode SOM digunakan untuk mengelompokkan data yang ber-referensi geografis (*geo-referenced*) yang mengintegrasikan visualisasi ruang keluaran dalam representasi kartografi melalui pengaturan warna, dan mengeksplorasi penggunaan lebar garis perbatasan antara unsur yang ber-referensi geografis, dihitung sesuai dengan jarak dalam ruang masukan terbaik antar lokasi. Dalam penelitian ini, metode SOM digunakan untuk melakukan klusterisasi Data Pemilih Tetap (DPT). Hasil klusterisasi kemudian digunakan sebagai dasar untuk menentukan kriteria warna dalam pengembangan aplikasi Web-GIS berdasarkan interval jumlah pemilih.

Kata Kunci : *Self-organizing map; Unsupervised ANN; Klustering; Geo-referenced; Cartographics representation*

1. Pendahuluan

Berbagai layanan web Geospasial seperti Google Maps, sistem navigasi GPS di dalam kendaraan, GPS-ponsel, dan sejumlah layanan berbasis lokasi telah menjadi bagian dari kebutuhan kita sehari-hari. Pada spektrum yang lain seperti aplikasi seperti survei tanah, perencanaan kota, pencegahan bencana alam, aplikasi manajemen sumber daya alam atau lingkungan yang telah ada pada beberapa dekade terakhir ini menggunakan dataset spasial sebagai dasar untuk studi yang kompleks, analisis waktu nyata dan pelaporan (Simion, 2012).

Informasi geospasial secara umum diimplementasikan dalam Sistem Informasi Geografis (SIG) untuk merepresentasikan pemodelan ruang dan objek dalam sebuah lokasi geografis dalam

peta (Papadakis dan Christodoulou, 2010). Peta telah menjadi sumber utama informasi sejak waktu yang lama. SIG menggunakan data digital, model elevasi, citra satelit, sistem pakar dan informasi sumber terbuka (*open source*) terkait dengan perencanaan, deteksi, evaluasi dan pengambilan keputusan. Peta sebagai sebuah SIG pada khususnya menggunakan atribut data spasial. Fitur informasi spasial berasal dari kedua data dan bahasa untuk komunikasi hasil analisis dan interpretasi (Benedict, 2002).

Banyak aplikasi dalam SIG membutuhkan fungsi tidak hanya dalam pengelolaan data dan visualisasi, tetapi juga dalam analisis dan pengambilan keputusan. Misalnya aplikasi perencanaan jalur metro baru di kota, SIG tidak mampu untuk menyelesaikan tugas tanpa adanya kepakaran manusia atau teknologi kecerdasan buatan. Terlepas

dari model untuk menganalisis dengan tema yang berbeda, beberapa aplikasi kasus pemecahan masalah menggunakan kecerdasan buatan (Wei, 2011). Pada umumnya SIG menyediakan kemampuan penanganan basisdata spasial untuk masukan data, penyimpanan, perolehan kembali, manajemen, manipulasi, analisis, dan keluaran (Frehner dan Brandli, 2006).

Penelitian yang dilakukan adalah dengan menganalisis sejauh mana metode *Self-Organizing Map (SOM)* efektif dalam mengimplementasikan suatu aplikasi visualisasi sebaran data dengan klusterisasi. SOM adalah sistem Jaringan Syaraf Tiruan (JST) secara bersamaan melakukan kuantisasi vektor dan proyeksi vektor. Karena karakteristik ini, SOM dapat divisualisasikan melalui ruang keluaran dengan mempertimbangkan perspektif proyeksi vektor, dan melalui ruang masukan data dengan menekankan proses kuantisasi vektor. (Gorricha dan Lobo, 2012).

2. Kerangka Teori

2.1. Definisi Jaringan Saraf Tiruan

Jaringan saraf tiruan (JST) atau *Artificial Neural Network (ANN)* didefinisikan sebagai jaringan dari sekelompok unit pemroses kecil yang dimodelkan berdasarkan jaringan saraf manusia. Jaringan syaraf tiruan merupakan sistem adaptif yang dapat merubah strukturnya untuk memecahkan masalah berdasarkan informasi eksternal maupun internal yang mengalir melalui jaringan tersebut. Secara sederhana, jaringan saraf tiruan adalah alat pemodelan dan statistik non-linier. JST dapat digunakan untuk memodelkan hubungan yang kompleks antara masukan dan keluaran untuk menemukan pola-pola pada data.

Menurut Siang (2005) JST merupakan sistem pemroses informasi yang memiliki karakteristik mirip dengan jaringan saraf biologi. JST dibentuk sebagai generalisasi model matematika dari jaringan saraf biologi dengan asumsi bahwa:

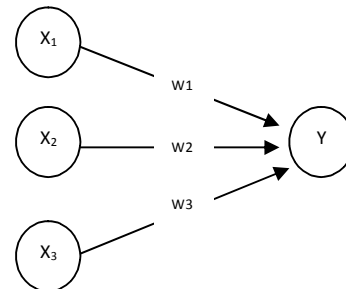
- Pemrosesan informasi terjadi pada banyak elemen sederhana.
- Sinyal dikirimkan diantara *neuron-neuron* melalui penghubung-penghubung.
- Penghubung antara *neuron* memiliki bobot yang akan memperkuat atau memperlemah sinyal.
- Untuk menentukan keluaran, setiap *neuron* menggunakan fungsi aktivasi (biasanya bukan fungsi linier) yang dikenakan pada jumlah masukan yang diterima. Besarnya keluaran ini selanjutnya dibandingkan dengan suatu batas ambang.

JST ditentukan oleh tiga hal sebagai berikut :

- Pola hubungan antar *neuron* (disebut arsitektur jaringan)
- Metode untuk menentukan bobot penghubung (disebut metode *training/learning/algorithm*).
- Fungsi aktivasi (fungsi transfer).

Neuron dalam JST sering diganti dengan istilah simpul. Setiap simpul tersebut untuk menerima atau mengirim sinyal dari atau ke simpul-simpul lainnya. Pengiriman sinyal disampaikan melalui penghubung. Kekuatan hubungan yang terjadi antara setiap simpul yang saling berhubung dikenal dengan nama bobot.

Arsitektur jaringan dan algoritma pelatihan sangat menentukan model-model JST. Arsitektur tersebut gunanya untuk menjelaskan arah perjalanan sinyal atau data di dalam jaringan. Sedangkan algoritma pelatihan menjelaskan bagaimana bobot koneksi harus diubah agar pasangan masukan-keluaran yang diinginkan dapat tercapai. Dalam setiap perubahan harga 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 masukan-keluaran akan meningkat.



Gambar 1. Sebuah sel saraf tiruan

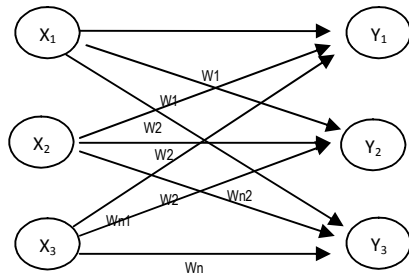
Y menerima masukan dari *neuron* x_1 , x_2 dan x_3 dengan bobot hubungan masing-masing adalah w_1 , w_2 dan w_3 . Ketiga impuls *neuron* yang akan dijumlahkan $net = x_1w_1 + x_2w_2 + x_3w_3$. Besarnya impuls yang diterima oleh Y mengikuti fungsi aktivasi $y = f(net)$. Apabila nilai fungsi aktivasi cukup kuat, maka sinyal akan diteruskan. Nilai fungsi aktivasi (keluaran model jaringan) juga dapat dipakai sebagai dasar untuk merubah bobot.

2.2. Arsitektur Jaringan Saraf Tiruan

Beberapa arsitektur jaringan yang sering dipakai dalam jaringan saraf tiruan digolongkan menjadi tiga model yaitu :

1. Jaringan lapis tunggal

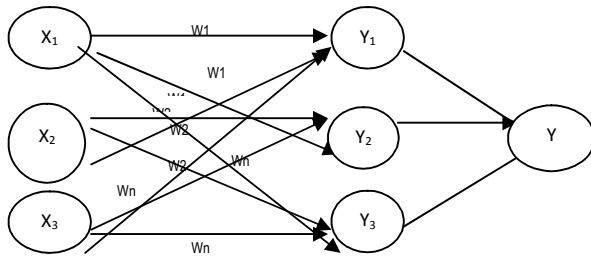
Merupakan sekumpulan *neuron* dihubungkan langsung dengan sekumpulan keluaran seperti tampak pada Gambar 2.2, yang memperlihatkan bahwa arsitektur jaringan lapis tunggal dengan n buah masukan (x_1, x_2, \dots, x_n) dan m buah keluaran (Y_1, Y_2, \dots, Y_m). Jaringan ini semua unit masukan dihubungkan dengan semua unit keluaran. Tidak ada unit masukan yang dihubungkan dengan unit masukan lainnya dan unit keluaran demikian.



Gambar 2. Jaringan Lapis Tunggal

2. Jaringan lapis jamak

Merupakan perluasan dari lapis tunggal. Dalam jaringan ini selain unit masukan dan keluaran, ada unit-unit lain yang sering disebut lapis tersembunyi. Lapis tersembunyi ini bisa saja lebih dari satu.



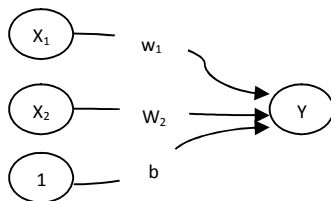
Gambar 3. Jaringan Lapis Jamak

3. Jaringan Recurrent

Model jaringan ini mirip dengan jaringan lapis tunggal ataupun jamak, hanya saja ada *neuron* keluaran yang memberikan sinyal pada unit masukan (sering disebut *feedback loop*), dengan kata lain sinyal mengalir dua arah, yaitu maju dan mundur.

2.3. Bias

Kadang-kadang dalam jaringan ditambahkan sebuah unit masukan yang nilainya selalu = 1. Unit yang sedemikian itu disebut bias (gambar 2.4). Bias dapat dipandang sebagai sebuah input yang nilainya= 1 (Siang, 2005).

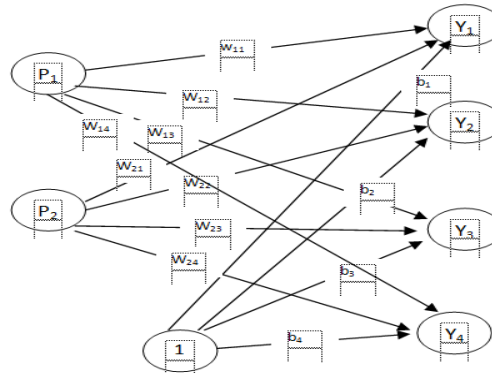


Gambar 4. Jaringan Lapis Jamak

2.4. Metode Self-Organizing Map (SOM)

Algoritma *Self Organizing Map* (SOM) merupakan suatu metode JST yang diperkenalkan

oleh Professor Teuvo Kohonen pada tahun 1980an. SOM merupakan salah satu bentuk topologi dari *Unsupervised Artificial Neural Network* (*Unsupervised ANN*) dimana dalam proses pelatihannya tidak memerlukan pengawasan (target keluaran). SOM digunakan untuk mengelompokkan (*clustering*) data berdasarkan karakteristik/fitur-fitur data. (Shieh dan Liao, 2012). Adapun arsitektur SOM dapat dilihat pada Gambar 5.



Gambar 5. Arsitektur SOM

Self Organizing Map (SOM) menyediakan suatu teknik visualisasi data yang membantu memahami data yang memiliki dimensi yang kompleks dengan mengurangi dimensi data kedalam peta. SOM juga merupakan konsep *clustering* dengan mengelompokkan data yang memiliki kemiripan tertentu. Oleh karena itu dapat dikatakan bahwa SOM mengurangi dimensi data dan menampilkan kesamaan antar data. Penggunaan SOM dalam penganalisaan data dapat mengakomodir data yang beragam mulai dari data dengan *single-dimensional* maupun *multi-dimensional* atau data dengan *single-variati* maupun data yang bersifat *multi-variati*. Dalam data mining SOM lazim digunakan untuk teknik data mining yang berkaitan dengan pengelompokan data (*clustering*) dengan demikian, maka seringkali SOM yang digunakan untuk proses *clustering* disebut dengan metode *SOM clustering*.

2.5. Algoritma dan Notasi SOM

Pada jaringan ini suatu lapisan yang berisi neuron-neuron akan menyusun dirinya sendiri berdasarkan masukan nilai tertentu dalam suatu kelompok yang dikenal dengan istilah *cluster*. Selama proses penyusunan diri, *cluster* yang memiliki *vector* bobot paling cocok dengan pola masukan (memiliki jarak yang paling dekat) akan terpilih sebagai pemenang.

Neuron yang menjadi pemenang beserta neuron-neuron tetangganya akan memperbaiki bobot-bobotnya. Apabila ingin dibagi data-data menjadi K *cluster*, maka lapisan kompetitif akan terdiri atas K buah neuron. Gambar 4 menunjukkan salah satu contoh arsitektur jaringan SOM dengan 2 unit pada

lapisan masukan (P_1 dan P_2), serta 3 unit (neuron) pada lapisan-lapisan keluaran (Y_1 , Y_2 dan Y_3). Sebagai catatan, bobot (W_{ij}) disini mengandung pengertian bobot yang menghubungkan neuron ke-j pada lapisan masukan ke neuron ke-i pada lapisan keluaran.

Pembelajaran kompetitif dengan metode ini diawali dengan memilih secara acak suatu *vector* masukan, misalnya *vector* masukan terpilih adalah *vector* masukan ke-z ($p(z)$). Kemudian dihitung jarak antara *vector* masukan $p(z)$ terhadap bobot ke neuron ke-I adalah $Dist(i)$. Selanjutnya nilai setiap jarak ini dinegatifkan dan ditambah dengan bobot biasnya. Misalkan hasilnya adalah $a(i)$, sehingga formulanya adalah sebagai berikut :

$$a(i) = -Dist(i) + b(i) \tag{2.1}$$

Kemudian dicari nilai $a(i)$ terbesar, misalkan terletak pada idx , dengan nilai terbesar adalah $MaxA$, maka neuron ke- idx akan menjadi pemenang. Set keluaran neuron ke- idx sama dengan 1, dan keluaran neuron lainnya sama dengan 0, sehingga formulanya adalah sebagai berikut :

$$y(i) = 1; \text{ jika } i = idx \tag{2.2}$$

$$y(i) = 0; \text{ jika } i \neq idx \tag{2.3}$$

Bobot yang menuju ke neuron idx ($w(idx,j)$) dengan $j = 1, 2, \dots$ jumlah variabel masukan akan di update :

$$w(idx,j) = w(idx,j) + \alpha(p(z,j) - w(idx,j)) \tag{2.4}$$

Dengan α adalah *learning rate*, yaitu faktor pengali pada perubahan bobot yang berubah terhadap perubahan error. Perubahan α ini sesuai dengan banyaknya input yang masuk. Faktor pengali α /learning rate ini akan selalu berkurang bila tidak ada perubahan error.

Bobot bias juga akan diupdate. Bobot bias yang menuju *neuron* pemenang akan dikurangi sehingga nilainya akan mendekati 0, sedangkan bobot bias selain *neuron* pemenang, nilainya akan bertambah. Formulanya sebagai berikut :

$$c(i) = (1 - \alpha)e^{(1 - \ln(b(i)))} + \alpha y(i) \tag{2.5}$$

$$b(i) = e^{(1 - \ln(c(i)))} \tag{2.6}$$

Proses pembelajaran ini akan berlangsung terus-menerus sehingga epoch mencapai maksimum epoch. Urutan algoritmanya adalah sebagai berikut :

- a. Inisialisasi bobot masukan (w_{ij}) :

$$\text{-----} \tag{2.7}$$

Dengan w_{ij} adalah bobot antara variabel masukan ke-j dengan *sneuron* pada kelas ke-i ($j=1, 2, \dots, m$; $i=1, 2, \dots, K$); $MinP_{id}$ dan $MaxP_i$ masing-masing adalah nilai terkecil pada variabel masukan ke-I, dan nilai terbesar dari variabel masukan ke-i.

Inisialisasi bobot bias (b_i)

$$b_i = e^{[1 - \ln(1/K)]} \tag{2.8}$$

dengan b_i adalah bobot bias ke neuron ke-I, dan K adalah jumlah klas.

Set parameter *learning rate*.

Set maksimum epoch (MaxEpoch).

- b. Set epoch = 0;
- c. Kerjakan jika epoch < MaxEpoch;
 - a. Epoch = epoch + 1
 - b. Pilih data secara acak, misalkan data terpilih adalah data ke-z.
 - c. Cari jarak antara data ke-z dengan setiap bobot masukan ke-i (D_i) :

$$D = \sum (w - P) \tag{2.9}$$

- d. Hitung a: $a_i = -D + b_i$ (2.10)

e. Cari a_i terbesar :

- i. $MaxA = \max(a_i)$, dengan $i=1, 2, \dots, K$
- ii. $Idx = I$, sedemikian hingga $a_i = MaxA$.

f. Set keluaran *neuron* ke-I (y_i) :

$$y_i = \begin{matrix} 1, & = \\ 0, & \neq \end{matrix} \tag{2.11}$$

g. Update bobot yang menuju ke neuron Idx :

$$w_{Idx-j} = w_{Idx-j} + \alpha(P_{zj} - w_{Idx-j}) \tag{2.12}$$

h. Update bobot bias :

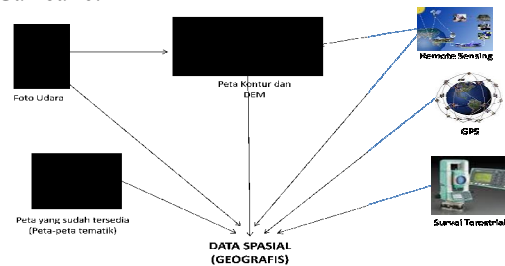
$$c(i) = (1 - \alpha)e^{(1 - \ln(b(i)))} + \alpha y(i) \tag{2.13}$$

$$b(i) = e^{(1 - \ln(c(i)))}$$

3. Metodologi

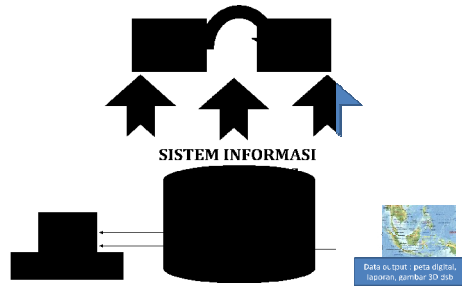
3.1. Bahan dan Alat Penelitian

Sumber-sumber data geografis (geospasial) diperoleh melalui beberapa cara, seperti terlihat pada Gambar 6.



Gambar 6. Sumber Data Geospasial (Nath, 2000).

Dari sumber-sumber data di atas, maka dapat digambarkan kerangka kerja dan tahapan pembuatan SIG seperti terlihat pada Gambar 7.



Gambar 7. Kerangka Kerja SIG (Nath *et al.*, 2000)

Dalam penelitian ini data yang dikumpulkan adalah data atribut dan data spasial, data yang telah dikumpulkan adalah sebagai berikut :

1. Peta Rupa Bumi Indonesia (RBI) kota Semarang dalam bentuk digital dengan skala 1:25.000, sistem proyeksi UTM, datum WGS 84 zone 49 S diperoleh dari BPN kota Semarang.
2. Data Pemilih Tetap (DPT) pada pemilihan Gubernur dan Wakil Gubernur Jawa Tengah tahun 2013 yang diperoleh dari kantor Komisi Pemilihan Umum (KPU) kota Semarang.

3.2. Alat Penelitian

Alat yang digunakan dalam menunjang pelaksanaan penelitian dan pengumpulan data adalah sebagai berikut :

1. Perangkat keras (*hardware*) terdiri dari :
 - a. Komputer : Acer Aspire 4315, Intel Celeron 550 (2.0 GHz), 1.99 GB of RAM
 - b. Printer : HP Deskjet D2566
2. Perangkat lunak (*software*) terdiri dari :
 - a. Sistem : *Microsoft Windows XP Profesional Version 2002 Service Pack 3*
 - b. *Microsoft Excel 2003*, digunakan untuk memasukkan data survei lapangan yang dapat dimasukkan dalam perangkat lunak *Mapserver* dan juga sebagai basisdata dalam penelitian ini.
 - c. *Autocad Land Desktop 2004*, digunakan untuk membuka berkas peta RBI sehingga dapat dimasukkan ke dalam perangkat lunak *Mapserver*.
 - d. *Matlab 7.01*, digunakan untuk melakukan analisis klusterisasi data DPT dan membuat GUI penerapan metode SOM.
 - e. *Mapserver* dengan *framework PMapper*, digunakan untuk pengolahan peta tematik yang memiliki data-data yang ada, data tersebut merupakan penggabungan dari *excel* dan *autocad*.
 - f. *ArcGIS 9.0* yang dikembangkan oleh *Environmental System Research Institute (ESRI)*, digunakan untuk memanipulasi data geografis, menggambar peta, query (pencarian), seleksi dan editing peta.

4. Hasil Dan Pembahasan

4.1. Data Geospasial

Informasi geospasial merupakan data geospasial yang sudah diolah sehingga dapat digunakan sebagai alat bantu dalam perumusan kebijakan, pengambilan keputusan, dan atau pelaksanaan kegiatan yang berhubungan dengan ruang kebumihian.

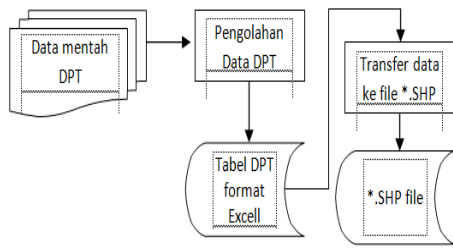
Penerapan metode *Self-Organizing Map (SOM)* pada penelitian ini sebagaimana tercantum dalam arsitektur SOM (gambar 2.6), dapat digambarkan sebagai berikut :

1. Sumber data yang digunakan adalah data pemilih tetap (DPT) pada pemilihan Gubernur dan Wakil Gubernur Jawa Tengah pada Pemilu tahun 2013.
2. Model SOM yang digunakan untuk proses klusterisasi data adalah sebagai berikut :
 - a. Data masukan sebagai pola digunakan yaitu jumlah pemilih laki-laki (x_1) dan jumlah pemilih perempuan (x_2) masing-masing sebanyak 16 masukan.
 - b. Data masukan bias yang kadang-kadang diperlukan dalam jaringan yang selalu bernilai 1.
 - c. Penentuan pembelajaran dengan melakukan analisis jumlah iterasi untuk mencapai maksimum *epoch* yang terbaik, dengan sebuah indikasi bahwa maksimum *epoch* akan tercapai apabila pada iterasi berikutnya sudah tidak terjadi perubahan hasil klusterisasi.
 - d. Pilih data secara acak, misalkan data terpilih adalah data ke-z, kemudian cari jarak antara data ke-z dengan setiap bobot input ke-i (D_i) dan mencari nilai a_i terbesar sebagaimana notasi 2.9 dan 2.10 untuk menentukan *neuron* keluaran.
 - e. *Update* bobot *neuron* ke- i dan sekaligus *update* bias untuk mendapatkan hasil klustering.

Pengolahan data DPT di kota Semarang pada pemilu Gubernur dan Wagub Jawa Tengah 2013 yang baru lalu, sebagaimana terlihat dalam berikut ini:

- a) Jumlah pemilih : 1.125.696 pemilih tetap, terdiri dari 547.192 pemilih laki-laki dan 578.504 pemilih perempuan.
- b) Di wilayah kota Semarang terdiri dari 177 Kelurahan dan 16 kecamatan dengan jumlah TPS sebanyak 2807 TPS.

Berdasarkan data statistik di atas, maka dapat digali sebaran data DPT sekaligus dapat diolah sebagai bahan studi kasus penelitian sebaran DPT. Adapun gambaran alur proses pengolahan datanya dapat dilihat sebagai berikut :



Gambar 8. Alur proses pengolahan data DPT

4.2. Proses Klasterisasi

Proses klasterisasi Tabel 4.1 di bawah ini menunjukkan Data Pemilih Tetap (DPT) pada Pemilihan Umum (Pemilu) Gubernur dan Wakil Gubernur Jawa Tengah 2013 kota Semarang yang akan digunakan sebagai data non spasial dalam penelitian ini.

Tabel 1. Data Pemilih Tetap (DPT) Pilgub Jateng 2013

No.	Kecamatan	Pemilih Pria	Pemilih Wanita
1	Semarang Tengah	23.974	26.831
2	Semarang Utara	43.720	46.822
3	Semarang Timur	28.077	30.640
4	Gayamsari	24.879	26.020
5	Genuk	33.396	33.755
6	Pedurungan	62.209	64.426
7	Semarang Selatan	24.714	27.496
8	Candisari	27.444	29.949
9	Gajahmungkur	20.214	22.026
10	Tembalang	53.178	55.675
11	Banyumanik	44.302	48.041
12	Gunungpati	28.737	29.302
13	Semarang Barat	55.125	58.692
14	Mijen	20.589	21.311
15	Ngaliyan	45.484	45.949
16	Tugu	11.150	11.569
	Jumlah	547.192	578.504

Sumber : KPU Kota Semarang

Dari data di atas dapat digambarkan dalam suatu susunan vektor sebagai berikut :

Vektor x(1)=(23,974 43,720 28,077 24,879 33,396 62,209 24,714 27,444 20,214 53,178 44,302 28,737 5 5,125 20,589 45,484 11,150)

Vektor x(2)=(26,831 46,822 30,640 26,020 33,755 64,426 27,496 29,949 22,026 55,675 48,041 29,302 58,692 21,311 45,949 11,569)

Dengan menggunakan algoritma SOM maka akan dikelompokkan 2 buah vektor tersebut ke dalam

maksimum 2 kelompok, dengan tahapan sebagai berikut :

1. Inisialisasi bobot

Kolom matriks bobot menyatakan jumlah komponen dalam sebuah vektor (dalam kasus ini=16) dan baris menyatakan jumlah maksimum kelompok yang akan dibentuk (dalam kasus ini=2). Misalkan secara acak didapat bobot awal :

$$w = \begin{pmatrix} 0.2 & 0.6 & 0.5 & 0.9 & 0.3 & 0.5 & 0.4 & 0.6 & 0.3 & 0.8 & 0.2 & 0.6 & 0.5 & 0.7 & 0.6 & 0.3 \\ 0.8 & 0.4 & 0.7 & 0.3 & 0.5 & 0.7 & 0.4 & 0.8 & 0.2 & 0.5 & 0.8 & 0.6 & 0.2 & 0.6 & 0.5 & 0.7 \end{pmatrix}$$

Jari-jari sekitar yang dipakai=0 dan laju pemahaman awal=0.6

2. Pelatihan (Training)

Pelatihan dilakukan untuk setiap vector sebagaimana berikut ini :

Untuk vector x(1) =

(23,974 43,720 28,077 24,879 33,396 62,209 24,714 27,444 20,214 53,178 44,302 28,737 55,125 20,589 45,484 11,150):

$$D(1) = (0.2 - 23,974)^2 + (0.6 - 43,720)^2 + (0.5 - 28,077)^2 + (0.9 - 24,879)^2 + (0.3 - 33,396)^2 + (0.5 - 62,209)^2 + (0.4 - 24,714)^2 + (0.6 - 27,444)^2 + (0.3 - 20,214)^2 + (0.8 - 53,178)^2 + (0.2 - 44,302)^2 + (0.6 - 28,737)^2 + (0.5 - 55,125)^2 + (0.7 - 20,589)^2 + (0.6 - 45,484)^2 + (0.3 - 11,150)^2 = 21.364$$

$$D(2) = (0.8 - 23,974)^2 + (0.4 - 43,720)^2 + (0.7 - 28,077)^2 + (0.3 - 24,879)^2 + (0.5 - 33,396)^2 + (0.7 - 62,209)^2 + (0.4 - 24,714)^2 + (0.8 - 27,444)^2 + (0.2 - 20,214)^2 + (0.5 - 53,178)^2 + (0.8 - 44,302)^2 + (0.6 - 28,737)^2 + (0.2 - 55,125)^2 + (0.6 - 20,589)^2 + (0.5 - 45,484)^2 + (0.7 - 11,150)^2 = 21.343$$

D(j) minimum untuk j = 2, maka vector bobot di baris 2 dimodifikasi menurut aturan :

$$= \quad + 0.6 (\quad - \quad)$$

$$W_{21} = 0.2 + 0.6 (0.2 - 0.2) = 0.2$$

$$W_{22} = 0.5 + 0.6 (0.6 - 0.5) = 0.56$$

$$W_{23} = 0.8 + 0.6 (0.5 - 0.8) = 0.62$$

$$W_{24} = 0.6 + 0.6 (0.9 - 0.6) = 0.78$$

$$W_{25} = 0.5 + 0.6 (0.3 - 0.5) = 0.38$$

$$W_{26} = 0.8 + 0.6 (0.5 - 0.8) = 0.62$$

$$W_{27} = 0.6 + 0.6 (0.4 - 0.6) = 0.48$$

$$W_{28} = 0.2 + 0.6 (0.6 - 0.2) = 0.44$$

$$W_{29} = 0.6 + 0.6 (0.3 - 0.6) = 0.42$$

$$W_{210} = 0.5 + 0.6 (0.8 - 0.5) = 0.68$$

$$W_{211} = 0.7 + 0.6 (0.2 - 0.7) = 0.40$$

$$W_{212} = 0.7 + 0.6 (0.6 - 0.7) = 0.64$$

$$W_{213} = 0.8 + 0.6 (0.5 - 0.8) = 0.62$$

$$W_{214} = 0.2 + 0.6 (0.7 - 0.2) = 0.50$$

$$W_{215} = 0.5 + 0.6 (0.6 - 0.5) = 0.56$$

$$W_{216} = 0.8 + 0.6 (0.3 - 0.8) = 0.50$$

Diperoleh vector bobot baru :

$$w = \begin{pmatrix} 0.20 & 0.60 & 0.50 & 0.90 & 0.30 & 0.50 & 0.40 & 0.60 & 0.30 & 0.80 & 0.20 & 0.60 & 0.50 & 0.70 & 0.60 & 0.30 \\ 0.20 & 0.56 & 0.62 & 0.78 & 0.38 & 0.62 & 0.48 & 0.44 & 0.42 & 0.68 & 0.40 & 0.64 & 0.62 & 0.50 & 0.56 & 0.50 \end{pmatrix}$$

3. Analisis Data

Untuk melakukan analisis data DPT dengan klusterisasi SOM, maka digunakan bantuan perangkat lunak Matlab 7.01. Fungsi Matlab yang digunakan untuk mendefinisikan perintah SOM adalah sebagai berikut :

Net = newsom (PR, [D1, D2, ...], TFCN, DFCN, OLR, OSTEPS, TLR, TND)

Dimana :

PR : matriks Rx2 yang berisi nilai minimum dan maksimum R buah elemen masukan

Di : ukuran topologi layar (*default* = [5 8])

TFCN : fungsi topologi (*default* = 'hextop')

DFCN : fungsi jarak (*default* = 'linkdist')

OLR : parameter laju pembelajaran awal (*learning rate, default* = 0,9)

OSTEPS: jumlah iterasi (*default* = 1000)

TLR : parameter *learning rate* berikutnya (*default* = 0,02)

TND : jarak di sekitar *neuron* pemenang

Fungsi topologi dapat diganti dengan 'gridtop' atau 'randtop', sedangkan fungsi jarak dapat diganti dengan 'dist' atau 'mandist'. Selanjutnya proses pengelompokan *neuron* dan melihat hasil pengelompokan dapat dilakukan dengan perintah-perintah di bawah ini :

Net = train(net, p)

b = sim(net, p)

ac = vec2ind (b)

Dari data DPT dapat dibuat struktur tabel dengan variabel seperti pada table 2.

Tabel 2. Data Pemilih Tetap (DPT) Pilgub Jateng 2013

No.	Kecamatan	X1	X2
1	Semarang Tengah	23,974	26,831
2	Semarang Utara	43,720	46,822
3	Semarang Timur	28,077	30,640
4	Gayamsari	24,879	26,020
5	Genuk	33,396	33,755
6	Pedurungan	62,209	64,426
7	Semarang Selatan	24,714	27,496
8	Candisari	27,444	29,949
9	Gajahmungkur	20,214	22,026
10	Tembalang	53,178	55,675
11	Banyumanik	44,302	48,041
12	Gunungpati	28,737	29,302
13	Semarang Barat	55,125	58,692
14	Mijen	20,589	21,311
15	Ngaliyan	45,484	45,949
16	Tugu	11,150	11,569

Keterangan :

X1 : Jumlah pemilih laki-laki

X2 : Jumlah pemilih perempuan

Perintah Matlab yang digunakan secara lengkap adalah sebagai berikut :

% Memasukkan matriks dpt (X1 - X4)

```
>> dpt=[23.974 43.72 28.077 24.879 33.396 62.209
24.714 27.444 20.214 53.178 44.302 28.737 55.125
20.589 45.484 11.15;
26.831 46.822 30.64 26.02 33.755 64.426 27.496
29.949 22.026 55.675 48.041 29.302 58.692 21.311
45.949 11.569];
```

% Menampilkan matriks per kolom >> q=dpt'

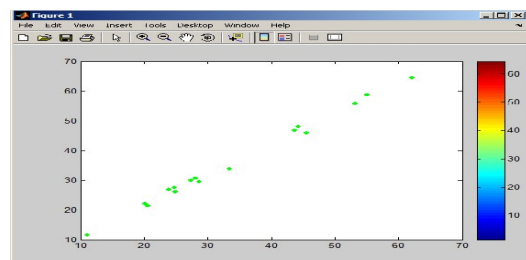
Hasilnya akan nampak sebagai berikut :

```
23.9740 26.8310
43.7200 46.8220
28.0770 30.6400
24.8790 26.0200
33.3960 33.7550
62.2090 64.4260
24.7140 27.4960
27.4440 29.9490
20.2140 22.0260
53.1780 55.6750
44.3020 48.0410
28.7370 29.3020
55.1250 58.6920
20.5890 21.3110
45.4840 45.9490
11.1500 11.5690
```

Matriks data masukan terdiri atas 16 baris yang merupakan Data Pemilih Tetap di 16 kecamatan di kota Semarang. Adapun kolom matriks terdiri dari 2 yang masing-masing merupakan data jumlah pemilih pria dan jumlah pemilih wanita. Untuk menggambarkan matriks data masukan digunakan instruksi plot, secara lengkap ditulis instruksi seperti berikut :

```
>>plot(dpt(1,:),dpt(2:),'green','markersize',15)
```

Selanjutnya diperoleh grafik sebagaimana gambar 20.



Gambar 9. Grafik matriks data masukan

Dari data masukan di atas dapat dilihat nilai minimum dan maksimum masing-masing nilai vektor masukan sebagaimana Tabel 19.

Tabel 3. Nilai Minimum dan Maksimum Masing-masing Nilai Vektor Masukan

Tingkat	Laki-Laki	Perempuan
Minimum	11.1500	11.5690
Maksimum	62.2090	64.4260

Berdasarkan data masukan nilai di atas selanjutnya dibangun jaringan kompetitif dengan jumlah empat klaster, parameter *learning rate* dan *Conscience learning rate* menggunakan *default*. Instruksi yang diketikkan dalam jendela perintah (*command window*) Matlab adalah :

```
>> net=newc([11.1500 62.2090; 11.5690
64.4260],4);
% Fungsi aktivasi keluaran
% Batasan max dan min tiap vektor masukan
% Dibentuk kelas dengan metode hextop 4x1
>>net=newsom(minmax(dpt),[4 1]);
```

% Inisialisasi beban awal

```
>>net.iw{1,1}
```

Hasilnya :

```
ans =
36.6795 37.9975
36.6795 37.9975
36.6795 37.9975
36.6795 37.9975
```

% Inisialisasi bias

```
>>net.b{1}
```

Hasilnya :

```
ans =
[]
```

% Fungsi aktivasi : (net=train (net,dpt) berfungsi untuk melatih jaringan NET sesuai dengan masukan data dtp

```
>>net=train(net,dpt);
```

Hasilnya :

```
TRAINR, Epoch 0/100
TRAINR, Epoch 25/100
TRAINR, Epoch 50/100
TRAINR, Epoch 75/100
TRAINR, Epoch 100/100
TRAINR, Maximum epoch reached.
```

% Perintah simulasi pada jaringan NET data dpt yang akan men-simulasikan model Simulink menggunakan semua pengaturan dialog parameter simulasi termasuk workspace I / O pilihan.

```
>>b=sim(net,dpt)
```

Hasilnya :

```
b =
(1,1) 1
(3,2) 1
(2,3) 1
(1,4) 1
```

```
(2,5) 1
(4,6) 1
(1,7) 1
(2,8) 1
(1,9) 1
(4,10) 1
(3,11) 1
(2,12) 1
(4,13) 1
(1,14) 1
(3,15) 1
(1,16) 1
```

% Perintah untuk melakukan transformasi vektor untuk indeks. IND2VEC dan VEC2IND memungkinkan indeks untuk diwakili baik oleh mereka sendiri atau sebagai vektor yang berisi 1 di deretan indeks yang diwakili.

```
>>ac=vec2ind(b);
>>net=init(net);
% Jumlah iterasi (epoch)
>>net.trainParam.epochs=60;
>>net=train(net,dpt);
```

Hasilnya tampak sebagaimana lampiran 1.

```
>>b=sim(net,dpt);
```

```
>>ac=vec2ind(b)
```

Hasilnya sebagai berikut :

```
ac =
1 3 2 1 2 4 1 2 1 4 3 2 4
1 3 1
```

Hasil proses klusterisasi dapat dilihat pada tabel 4.

Tabel 4. Hasil klusterisasi

No.	Kecamatan	Klaster
1	Semarang Tengah	1
2	Semarang Utara	3
3	Semarang Timur	2
4	Gayamsari	1
5	Genuk	2
6	Pedurungan	4
7	Semarang Selatan	1
8	Candisari	2
9	Gajahmungkur	1
10	Tembalang	4
11	Banyumanik	3
12	Gunungpati	2
13	Semarang Barat	4
14	Mijen	1
15	Ngaliyan	3
16	Tugu	1

Sehingga, secara lengkap hasil klusterisasi menghasilkan empat klaster sebagaimana tabel 5.

Tabel 5. Hasil klasterisasi





Klaster	Kecamatan
1	Gayamsari, Semarang Selatan, Semarang Tengah, Gajah Mungkur, Mijen, Tugu
2	Semarang Timur, Genuk, Candisari, Gunungpati
3	Semarang Utara, Banyumanik, Ngaliyan
4	Pedurungan, Tembalang, Semarang Barat

Berdasarkan data DPT yang terlihat dari kedua variabel X1 dan X2, pada 16 kecamatan di kota Semarang, dengan menggunakan metode JST Self Organizing Maps (SOM) diperoleh 4 klaster yang mencerminkan kondisi DPT yang saling berdekatan rata-rata jumlahnya. Kedua variabel tersebut meliputi jumlah pemilih laki-laki dan jumlah pemilih perempuan. Dari tabel di atas, dapat dilihat kecamatan yang sifatnya relatif berdekatan (jumlahnya) berdasarkan kedua variabel tersebut.

4.3. Pemilihan Warna Klaster

Untuk melakukan pemilihan warna klaster dapat diasumsikan dan untuk visualisasi program aplikasinya sebagaimana tabel 6.

Tabel 6. Hasil Perhitungan statistik

Klaster	Kecamatan	Warna
1	Gayamsari, Semarang Selatan, Semarang Tengah, Gajah Mungkur, Mijen, Tugu	
2	Semarang Timur, Genuk, Candisari, Gunungpati	
3	Semarang Utara, Banyumanik, Ngaliyan	
4	Pedurungan, Tembalang, Semarang Barat	

Tabel 6 menunjukkan pemilihan warna dari hasil proses klasterisasi. Warna Hijau, Biru, Merah dan Ungu merupakan warna untuk menentukan interval jumlah pemilih (laki-laki dan perempuan).

Klaster 1 : diberi warna Hijau untuk interval antara jumlah pemilih antara : 22.719 - 52.210 (Kecil)

Klaster 2 : diberi warna Biru untuk interval antara jumlah pemilih antara : 57.393 - 67.151 (Cukup)

Klaster 3 : diberi warna Merah untuk interval antara jumlah pemilih antara : 90.542 - 92.343 (Sedang)

Klaster 4 : diberi warna Ungu untuk interval antara jumlah pemilih antara : 108.853 - 126.635 (Besar)

4.4. Analisis Proses Training

Dari hasil proses training terhadap jumlah iterasi yang dilakukan, maka dapat dilihat hasil klasterisasi sebagaimana tabel 7.

Tabel 7. Hasil proses training pada iterasi ke 10 s/d 100, 300, 500, 1000, 3000, dan 5000.

Jenis Kelamin	Jumlah Data	Iterasi	Hasil Klasterisasi			
Laki-Laki & Perempuan	2 baris, 16 kolom	10	1	4	2	1
			3	4	1	2
			1	4	4	2
		20	4	1	4	1
			1	4	2	1
			3	4	1	2
		30	1	4	4	2
			4	1	4	1
			1	4	1	1
		40	3	4	1	1
			1	4	4	1
			4	1	4	1
		50	2	3	2	2
			2	4	2	2
			1	4	3	2
		60	4	1	3	1
			1	3	2	1
			2	4	2	2
		70	1	4	3	2
			4	1	3	1
			1	3	2	1
		80	2	4	1	2
			1	4	3	2
			4	1	3	1
90	1	3	2	1		
	2	4	1	2		
	1	4	3	2		
100	4	1	3	1		
	1	3	2	1		
	2	4	1	2		
300	1	4	3	2		
	4	1	3	1		
	1	3	2	1		
500	2	4	1	2		
	1	4	3	2		
	4	1	3	1		
1000	1	3	2	1		
	2	4	1	2		
	1	4	3	2		
3000	4	1	3	1		
	1	3	2	1		
	2	4	1	2		
5000	1	4	3	2		
	4	1	3	1		
	1	3	2	1		

Dari tabel 7 di atas maka dapat diambil suatu kesimpulan bahwa pada iterasi ke 60 merupakan max epoch dari proses training data-data di atas, karena pada iterasi berikutnya sudah tidak terjadi perubahan susunan klaster.

5. Kesimpulan

Berdasarkan uraian di atas, maka dapat diambil beberapa kesimpulan sebagai berikut :

1. Perangkat lunak aplikasi yang telah dibuat dapat digunakan untuk menyelesaikan pengambilan keputusan terkait visualisasi data DPT dengan berbagai pertimbangan menyangkut jumlah pemilih, jumlah TPS dan beberapa pertimbangan lainnya.
2. Perangkat lunak aplikasi yang telah dibuat dapat digunakan untuk mempermudah melihat data secara visual terkait dengan DPT di wilayah kota Semarang.
3. Dari hasil pengujian dapat disimpulkan bahwa dengan menggunakan metode *Self-Organizing Map* (SOM) dapat ditetapkan klasterisasi dan pewarnaan yang dipilih sebagai dasar visualisasi sebaran data DPT di kota Semarang.

Daftar Pustaka

Frehner, M. dan Brandli, M., S.D., 2006. Virtual database: spatial analysis in a web-based data management system for distributed ecological data. *International Journal of Environmental Modelling & Software* 21, 1544-1554.

Gorricha, J. dan Lobo, V.S.D., 2012. Improvements on the visualization of clusters in geo-referenced data using Self-Organizing Maps. *International Journal of Computers & Geosciences* 43, 177-186.

Benedict, S.D., 2002. A GIS Application To Enhance Cell-Based Information Modelling, *International Journal of Information System* 142, 151-160.

Nath, S.D., 2000. Applications Of Geographical Information Systems (GIS) For Spatial Decision Support In Aquaculture. *International Journal of Aquacultural Engineering* 23, 233-278.

Shieh, S.L. dan Liao, I.E., S.D., 2012. A new approach for data clustering and visualization using self-organizing maps. *International Journal of Expert Systems with Applications* 39, 1924-11933.

Siang, J.J., 2009. Jaringan Syaraf Tiruan & Pemrogramannya Menggunakan Matlab. Yogyakarta : ANDI.

Simion, S.D., 2012. Speeding Up Spatial Database Query Execution Using Gpus. *Procedia Computer Science* 9, 1870 - 1879.

Wei, S.D., 2011. Knowledge-based problem solving method in SIG application. *Journal International of Knowledge-Based Systems* 24, 542-553.