

**PERBANDINGAN METODE *K-MEANS*
DENGAN METODE *FUZZY C-MEANS*
UNTUK MENENTUKAN PUSAT *CLUSTER*
PADA JARINGAN *RADIAL BASIS FUNCTION* (RBF)
(Studi Kasus pada Data Saham Harian PT.Astra, Tbk)**

SKRIPSI

**Oleh:
PUTRA ADI
0410940046-94**



**PROGRAM STUDI MATEMATIKA
JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS BRAWIJAYA
MALANG
2009**

**PERBANDINGAN METODE *K-MEANS*
DENGAN METODE *FUZZY C-MEANS*
UNTUK MENENTUKAN PUSAT *CLUSTER*
PADA JARINGAN *RADIAL BASIS FUNCTION* (RBF)
(Studi Kasus pada Data Saham Harian PT.Astra, Tbk)**

SKRIPSI

Sebagai salah satu syarat untuk memperoleh gelar
Sarjana Sains dalam bidang Matematika

Oleh:
PUTRA ADI
0410940046-94



**PROGRAM STUDI MATEMATIKA
JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS BRAWIJAYA
MALANG
2009**

LEMBAR PENGESAHAN SKRIPSI

**PERBANDINGAN METODE *K-MEANS* DENGAN
METODE *FUZZY C-MEANS*
UNTUK MENENTUKAN PUSAT *CLUSTER*
PADA JARINGAN *RADIAL BASIS FUNCTION* (RBF)
(Studi Kasus pada Data Saham Harian PT.Astra, Tbk)**

Oleh:
PUTRA ADI
0410940046-94

**Setelah dipertahankan di depan majelis penguji
pada tanggal 11 Februari 2009
dan dinyatakan memenuhi syarat untuk memperoleh gelar
Sarjana Sains dalam Bidang Matematika**

Dosen Pembimbing I

Dosen Pembimbing II

Syaiful Anam, S.Si, M.T
NIP. 132 300 237

Drs. Hery Subagyo, M.Kes
NIP. 131 281 896

**Mengetahui,
Ketua Jurusan Matematika
Fakultas MIPA Universitas Brawijaya**

Dr. Agus Suryanto, MSc
NIP. 132 126 049

LEMBAR PERNYATAAN

Saya yang bertanda tangan di bawah ini :

Nama : PUTRA ADI
NIM : 0410940046 - 94
Jurusan : Matematika
Penulis Skripsi berjudul : PERBANDINGAN METODE *K-MEANS* DENGAN METODE *FUZZY C-MEANS* UNTUK MENENTUKAN PUSAT *CLUSTER* PADA JARINGAN *RADIAL BASIS FUNCTION* (RBF)
 (Studi Kasus pada Data Saham Harian PT.Astra, Tbk)

Dengan ini menyatakan bahwa :

1. Isi dari Skripsi yang saya buat benar-benar karya sendiri dan tidak menjiplak karya orang lain, selain nama-nama yang termaktub di isi dan tertulis di daftar pustaka Skripsi ini.
2. Apabila di kemudian hari ternyata Skripsi yang saya tulis terbukti hasil jiplakan, maka saya bersedia menanggung segala resiko yang akan saya terima.

Demikian pernyataan ini dibuat dengan segala kesadaran.

Malang, 11 Februari 2009
 Yang menyatakan,

(Putra Adi)
 NIM. 0410940046-94



**PERBANDINGAN METODE *K-MEANS*
DENGAN METODE *FUZZY C-MEANS*
UNTUK MENENTUKAN PUSAT *CLUSTER*
PADA JARINGAN *RADIAL BASIS FUNCTION* (RBF)
(Studi Kasus pada Data Saham Harian PT.Astra, Tbk)**

ABSTRAK

Jaringan saraf tiruan (JST) adalah sistem pemrosesan informasi yang memiliki performansi tertentu seperti pada jaringan saraf biologis. Untuk memperoleh JST yang mampu menyelesaikan permasalahan diperlukan beberapa tahapan diantaranya membangun jaringan (*design*), melatih jaringan (*training*) dan terakhir menguji kemampuan jaringan (*testing*). Dalam skripsi ini JST yang dibahas adalah jaringan *Radial Basis Function* (RBF). Jaringan RBF adalah model JST yang menggunakan metode pembelajaran hibrida. Artinya, jaringan RBF mengalami proses pembelajaran terpisah, yaitu pembelajaran tidak terawasi pada lapisan *input* dan pembelajaran terawasi pada lapisan *output*. Untuk pembelajaran tidak terawasi, metode yang digunakan adalah metode *clustering*. Pada langkah analisis *cluster* digunakan metode *K-Means* untuk menentukan pusat-pusat Fungsi Basis Rasial. Namun metode ini memiliki kelemahan yakni proses pelatihannya sulit untuk mencapai kekonvergenan walaupun waktu yang dibutuhkan relatif singkat. Untuk itu, dalam skripsi ini digunakan metode lain yaitu metode *Fuzzy C-Means* sebagai pembandingnya. Adapun tujuan dari skripsi ini yaitu mengetahui perbandingan dari kedua metode dan membandingkan hasil prediksi harga penutupan dari data saham harian PT.Astra, Tbk. Sebagai kesimpulan, jaringan RBF dengan metode *Fuzzy C-Means* lebih baik dibandingkan metode *C-Means* karena metode ini mampu mencapai kekonvergenan dengan nilai *error* yang sangat kecil.

Kata kunci : Jaringan Saraf Tiruan (JST), Jaringan Fungsi Basis Radial (RBF), metode *K-Means*, metode *Fuzzy C-Means*



**PERBANDINGAN METODE *K-MEANS*
DENGAN METODE *FUZZY C-MEANS*
UNTUK MENENTUKAN PUSAT *CLUSTER*
PADA JARINGAN *RADIAL BASIS FUNCTION* (RBF)
(Studi Kasus pada Data Saham Harian PT.Astra, Tbk)**

ABSTRAK

Jaringan saraf tiruan (JST) adalah sistem pemrosesan informasi yang memiliki performansi tertentu seperti pada jaringan saraf biologis. Untuk memperoleh JST yang mampu menyelesaikan permasalahan diperlukan beberapa tahapan diantaranya membangun jaringan (*design*), melatih jaringan (*training*) dan terakhir menguji kemampuan jaringan (*testing*). Dalam skripsi ini JST yang dibahas adalah jaringan *Radial Basis Function* (RBF). Jaringan RBF adalah model JST yang menggunakan metode pembelajaran hibrida. Artinya, jaringan RBF mengalami proses pembelajaran terpisah, yaitu pembelajaran tidak terawasi pada lapisan *input* dan pembelajaran terawasi pada lapisan *output*. Untuk pembelajaran tidak terawasi, metode yang digunakan adalah metode *clustering*. Pada langkah analisis *cluster* digunakan metode *K-Means* untuk menentukan pusat-pusat Fungsi Basis Rasial. Namun metode ini memiliki kelemahan yakni proses pelatihannya sulit untuk mencapai kekonvergenan walaupun waktu yang dibutuhkan relatif singkat. Untuk itu, dalam skripsi ini digunakan metode lain yaitu metode *Fuzzy C-Means* sebagai pembandingnya. Adapun tujuan dari skripsi ini yaitu mengetahui perbandingan dari kedua metode dan membandingkan hasil prediksi harga penutupan dari data saham harian PT.Astra, Tbk. Sebagai kesimpulan, jaringan RBF dengan metode *Fuzzy C-Means* lebih baik dibandingkan metode *C-Means* karena metode ini mampu mencapai kekonvergenan dengan nilai *error* yang sangat kecil.

Kata kunci : Jaringan Saraf Tiruan (JST), Jaringan Fungsi Basis Radial (RBF), metode *K-Means*, metode *Fuzzy C-Means*



**THE COMPARISON OF K-MEANS METHOD
AND FUZZY C-MEANS METHOD
TO DETERMINE CLUSTER CENTER
ON RADIAL BASIS FUNCTION NEURAL NETWORK
(A Case Study On Daily Stocks Data Of PT.Astra, Tbk)**

ABSTRACT

Artificial Neural Network (ANN) is an information processing system that has performances certainly related to biological neural network. To obtain neural network which is capable to solved problems, we need several stages to design, training and testing the network. In this final project, we used Radial Basis Function (RBF). RBF is an ANN which using hybrid learning. It means that, the process have been separated, unsupervised learning on input layer and supervised learning on output layer. In cluster analysis, we used K-Means method to determine the center of RBF. Although this method needs a short time to reach a convergence but is difficult to do. Therefore, in this final project, we used Fuzzy C-Means method. The purpose of this final project is to compare the method and the result of input data from the closing price of daily stock data of PT.Astra, Tbk by using K-Means and Fuzzy C-Means. In the result, Fuzzy C-Means is better, because it can reach a convergence with small error value.

Keywords : Artificial Neural Network (ANN), Radial Basis Function Network, K-Means method, Fuzzy C-Means method



KATA PENGANTAR

Segala puji dan syukur penulis panjatkan ke hadirat Allah SWT, yang telah melimpahkan rahmat, taufik dan hidayah-Nya sehingga penulis sebagai hamba-Nya yang lemah mampu menyelesaikan skripsi ini. Shalawat serta salam tetap tercurahkan kepada Nabi Muhammad SAW.

Dalam penulisan skripsi ini, banyak dukungan dan bantuan yang penulis terima. Oleh sebab itu, penulis ingin berterima kasih kepada:

1. Syaiful Anam, S.Si, M.T, selaku pembimbing I atas segala bimbingan, nasehat, dan sarannya, serta waktu yang telah diberikan kepada penulis selama penulisan tugas akhir ini.
2. Drs. Hery Subagio, M.kes, selaku Pembimbing II atas dorongan, semangat, kritik dan sarannya, serta waktu yang telah diberikan kepada penulis selama penulisan tugas akhir ini.
3. Drs. Noor Hidayat, MSi., Drs. M. Muslikh, MSi., dan Dra. Trisilowati, MSc. selaku dosen penguji atas segala saran yang diberikan untuk perbaikan skripsi ini.
4. Dr. Agus Suryanto, M.Sc selaku Ketua Jurusan Matematika atas semua bantuannya dalam kelancaran kelulusan penulis.
5. Bapak dan Ibu Dosen jurusan Matematika yang telah memberikan bekal ilmu kepada penulis.
6. Ibunda dan ayahanda tercinta atas segala doa, kasih sayang, motivasi, didikan dan bimbingannya yang tak pernah kenal lelah beliau berikan kepada penulis secara tulus dan ikhlas.
7. Zakijah, S.Si., atas kesabaran, semangat dan motivasi tiada henti yang diberikan selama penulisan skripsi ini.
8. Semua pihak yang tidak dapat penulis sebutkan satu persatu.

Akhirnya, semoga Allah SWT yang akan membalas semuanya. Penulis menyadari sepenuhnya bahwa dalam penulisan tugas akhir ini tidak lepas dari kesalahan atau kekurangan. Oleh sebab itu, saran dan kritik yang membangun sangat diharapkan oleh penulis.

Akhir kata, semoga tulisan ini bermanfaat bagi penulis khususnya dan semua pihak pada umumnya.

Malang, 11 Februari 2009

Penulis



DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
LEMBAR PENGESAHAN	iii
LEMBAR PERNYATAAN	v
ABSTRAK	vii
ABSTRACT	ix
KATA PENGANTAR	xi
DAFTAR ISI	xiii
DAFTAR GAMBAR	xv
DAFTAR TABEL	xvii
DAFTAR LAMPIRAN	xix
 BAB I PENDAHULUAN	
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan Masalah	2
1.4 Batasan Masalah	3
 BAB II TINJAUAN PUSTAKA	
2.1 <i>Data Mining</i>	5
2.2 Ruang Vektor	6
2.3 Teknik <i>Clustering</i>	7
2.3.1 Partisi klasik (<i>Hard Partition</i>)	8
2.3.2 Partisi fuzzy (<i>Fuzzy Partition</i>).....	9
2.3.3 <i>K-Means clustering</i>	10
2.3.4 <i>Fuzzy C-Means clustering</i>	11
2.4 Jaringan Saraf Tiruan	17
2.4.1 Arsitektur Jaringan	18
2.4.2 Metode "Pembelajaran"	20
2.4.3 Fungsi Aktivasi.....	21
2.5 Jaringan <i>Radial Basis Function</i> (RBF).....	22
2.6 Optimasi	23
2.6.1 Metode <i>Gradient Descent</i>	25

BAB III METODOLOGI

3.1 Jenis dan Sumber Data	27
3.2 Metode Pengumpulan Data	27
3.3 Normalisasi Data	27
3.4 Membangun JST	28
3.4.1 Metode <i>clustering</i>	28
3.4.2 Jaringan RBF	28
3.5 Pengukuran Kinerja	29
3.5.1 Metode <i>clustering</i>	29
3.5.2 Jaringan RBF	30
3.6 Perbandingan Hasil "Pembelajaran"	31
3.7 Perbandingan Hasil Pengujian	32

BAB IV PEMBAHASAN

4.1 Penerapan Metode <i>Gradient Descent</i> Pada Model "Pembelajaran" Jaringan RBF	33
4.2 Hasil "Pembelajaran" Pada Proses <i>Clustering</i>	39
4.2.1 Hasil <i>clustering</i> dengan metode <i>K-Means</i>	39
4.2.2 Hasil <i>clustering</i> dengan metode FCM	40
4.2.3 Pengukuran kinerja metode <i>clustering</i>	41
4.3 Hasil "Pembelajaran" Pada Jaringan RBF	41
4.3.1 Hasil "pembelajaran" jaringan RBF dengan metode <i>K-Means</i> dan metode <i>Gradient Descent</i>	42
4.3.2 Hasil "pembelajaran" jaringan RBF dengan metode FCM dan metode <i>Gradient Descent</i>	45
4.3.3 Pengukuran kinerja jaringan RBF	48
4.4 Pemilihan Model Arsitektur Untuk Pengujian Jaringan RBF ..	49
4.5 Perbandingan Hasil Pengujian	52

BAB V PENUTUP

5.1 Kesimpulan	57
5.2 Saran	58

DAFTAR PUSTAKA	59
LAMPIRAN-LAMPIRAN	61

DAFTAR GAMBAR

	Halaman
Gambar 2.1 <i>Cluster</i>	7
Gambar 2.2 <i>Cluster</i> Data Menggunakan Partisi Klasik	9
Gambar 2.3 Arsitektur Jaringan Lapisan Tunggal	19
Gambar 2.4 Arsitektur Jaringan Multi Lapisan	19
Gambar 2.5 Arsitektur Jaringan <i>Reccurent</i>	20
Gambar 2.6 Struktur Neuron JST	21
Gambar 2.7 Minimum Lokal Dan Minimum Global	24
Gambar 4.1 Arsitektur JST Dengan Model "Pembelajaran" Jaringan RBF Secara Umum	33
Gambar 4.2 Grafik Hubungan Antara Fungsi Objektif Dengan Iterasi Pada Metode <i>K-Means</i>	39
Gambar 4.3 Grafik Hubungan Antara Fungsi Objektif Dengan Iterasi Pada Metode FCM.....	40
Gambar 4.4 Hasil "Pembelajaran" Metode <i>K-Means</i> Dengan $\alpha = 0,2$ Dan <i>Spread</i> =2	43
Gambar 4.5 Hasil "Pembelajaran" Metode <i>K-Means</i> Dengan $\alpha = 0,2$ Dan <i>Spread</i> =2	43
Gambar 4.6 Hasil "Pembelajaran" Metode <i>K-Means</i> Dengan $\alpha = 0,5$ Dan <i>Spread</i> =2	44
Gambar 4.7 Hasil "Pembelajaran" Metode <i>K-Means</i> Dengan $\alpha = 0,8$ Dan <i>Spread</i> =2	44
Gambar 4.8 Hasil "Pembelajaran" Metode FCM Dengan $\alpha = 0,2$ Dan <i>Spread</i> =2	46
Gambar 4.9 Hasil "Pembelajaran" Metode FCM Dengan $\alpha = 0,35$ Dan <i>Spread</i> =2	46
Gambar 4.10 Hasil "Pembelajaran" Metode FCM Dengan $\alpha = 0,5$ Dan <i>Spread</i> =2	47
Gambar 4.11 Hasil "Pembelajaran" Metode FCM Dengan $\alpha = 0,8$ Dan <i>Spread</i> =2	48
Gambar 4.12 Hasil "Pembelajaran" Dari Metode <i>K-Means</i> Dengan <i>Spread</i> Yang Berbeda	49
Gambar 4.13 Hasil "Pembelajaran" Dari Metode FCM Dengan <i>Spread</i> Yang Berbeda	50
Gambar 4.14 Hasil "Pembelajaran" Dari Metode <i>K-Means</i> Dengan Laju "Pembelajaran" Yang Berbeda	51

Gambar 4.15	Hasil "Pembelajaran" Dari Metode FCM Dengan Laju "Pembelajaran" Yang Berbeda.....	52
Gambar 4.16	Perbandingan Target Dengan <i>Output</i> Jaringan Dengan Metode <i>K-Means</i>	54
Gambar 4.17	Selisih Antara Target Dengan <i>Output</i> Jaringan..	54
Gambar 4.18	Perbandingan Target Dengan <i>Output</i> Jaringan Dengan Metode FCM	56
Gambar 4.19	Selisih Antara Target Dengan <i>Output</i> Jaringan..	56



DAFTAR TABEL

	Halaman
Tabel 4.1 Hasil <i>clustering</i> dengan metode <i>K-means</i>	39
Tabel 4.2 Hasil <i>clustering</i> dengan metode FCM	40
Tabel 4.3 Perbandingan hasil <i>clustering</i>	41
Tabel 4.4 Hasil "pembelajaran" dengan metode <i>K-Means</i>	42
Tabel 4.5 Hasil "pembelajaran" dengan metode FCM.....	45
Tabel 4.6 Perbandingan hasil jaringan RBF	48
Tabel 4.7 Hasil simulasi menggunakan metode <i>K-Means</i>	53
Tabel 4.8 Hasil simulasi menggunakan metode FCM.....	55





DAFTAR LAMPIRAN

	Halaman
Lampiran 1. Nilai bobot dan pusat <i>cluster</i> dari hasil "pembelajaran" metode <i>K-Means</i> dengan model 5-5-1	61
Nilai bobot dan pusat <i>cluster</i> dari hasil "pembelajaran" metode FCM dengan model 5-5-1	61
Lampiran 2 Diagram alir pelatihan jaringan RBF dengan metode <i>Fuzzy C-Means</i>	62
Lampiran 3. Data Perdagangan Saham Harian PT.Astra, Tbk	66





BAB I PENDAHULUAN

1.1 LATAR BELAKANG

Saham dapat didefinisikan sebagai tanda penyertaan atau kepemilikan atas suatu perusahaan tertentu, di mana tiap saham menunjukkan satu suara kepemilikan. Saham dikenal memiliki karakteristik *high risk-high return*. Saham memungkinkan pemodal mendapatkan keuntungan dalam jumlah besar dalam waktu singkat. Namun dengan berfluktuasinya harga saham, saham juga dapat membuat investor mengalami kerugian dalam waktu yang singkat pula.

Untuk mempelajari perilaku saham, para peneliti menciptakan alat, baik berupa teori, teknik analisa, model, dan lain-lain yang dibuat untuk mengetahui perilaku harga saham tersebut. Karena sifatnya yang menyederhanakan, maka alat yang diciptakan tersebut tidak akan pernah sama dengan realitanya, hanya mendekati saja. Meskipun ada kekurangannya, alat akan sangat membantu memahami perilaku harga saham yang akan muncul dikemudian hari dengan cara memprediksikan pergerakan harga saham di masa yang akan datang. Karena itu, analisa mengenai perilaku harga saham semakin berkembang dan menghasilkan berbagai metode peramalan harga saham. Metode peramalan dengan perfomansi yang baik serta keakuratan yang tinggi menjadi sangat diperlukan dalam melakukan analisa terhadap harga saham.

Salah satu bentuk analisa terhadap perilaku harga saham yang banyak dikembangkan adalah analisa teknikal di mana bentuk analisa ini menggunakan pola pergerakan harga saham yang berfluktuasi naik turun dan berulang untuk memprediksikan harga di masa yang akan datang. Salah satu teknologi yang sangat populer untuk analisa teknikal adalah Jaringan Saraf Tiruan (JST). Metode JST memberikan solusi dengan akurasi yang lebih baik dibandingkan analisa statistik walaupun kelengkapan datanya tidak sempurna data untuk statistik (Purnomo dan Agus, 2006).

Jaringan *Radial Basis Function* (Jaringan RBF) adalah salah satu jenis dari JST yang menggunakan metode "pembelajaran" hibrida. Artinya, jaringan RBF mengalami proses "pembelajaran" terpisah, yaitu "pembelajaran" tidak terawasi pada lapisan masukan ke lapisan tersembunyi dan "pembelajaran" terawasi pada lapisan

tersembunyi ke lapisan keluaran. Pada langkah "pembelajaran" tidak terawasi digunakan analisa *cluster* klasik, yaitu *K-Means* untuk menentukan pusat-pusat *cluster*. Namun metode ini memiliki kelemahan yaitu pada proses "pembelajaran" sulit untuk mencapai kekonvergenan walaupun waktu yang dibutuhkan relatif singkat (Augusta, 2007).

Oleh karena itu, dalam skripsi ini digunakan metode yang lain yaitu metode *Fuzzy C-Means* (FCM). FCM adalah sebuah metode untuk mengelompokkan data yang memungkinkan satu data menjadi dua *cluster* atau lebih. Algoritma FCM digunakan karena pada metode FCM kemungkinan kegagalan untuk konvergen lebih kecil dibandingkan metode *K-Means* (Augusta, 2007).

Pada langkah "pembelajaran" terawasi digunakan metode optimasi, yaitu *Gradient Descent* dalam perubahan bobotnya untuk meminimumkan kesalahan. Pada setiap iterasi, bobot-bobot baru dihitung dengan menggunakan laju "pembelajaran" yang ada.

Skripsi ini membandingkan metode RBF standar menggunakan *K-Means* dengan metode RBF modifikasi menggunakan FCM untuk menentukan pusat-pusat *cluster*. Untuk lebih mempermudah pemahaman, maka baik metode RBF standar maupun RBF modifikasi akan diimplementasikan pada suatu studi kasus, dalam hal ini untuk memprediksi harga saham penutupan pada data perdagangan saham harian PT.Astra, Tbk.

1.2 RUMUSAN MASALAH

Berdasarkan latar belakang permasalahan di atas, pokok permasalahan yang dikemukakan dalam skripsi ini adalah sebagai berikut.

1. Bagaimana perbandingan metode *K-Means* dengan metode FCM dalam proses *clustering* untuk mendapatkan pusat-pusat *cluster*?
2. Bagaimana implementasi metode FCM dan metode *Gradient Descent* pada jaringan RBF?
3. Bagaimana perbandingan metode RBF standar dengan metode RBF modifikasi setelah diaplikasikan pada prediksi harga saham penutupan dari data perdagangan saham harian PT.Astra, Tbk?

1.3 TUJUAN

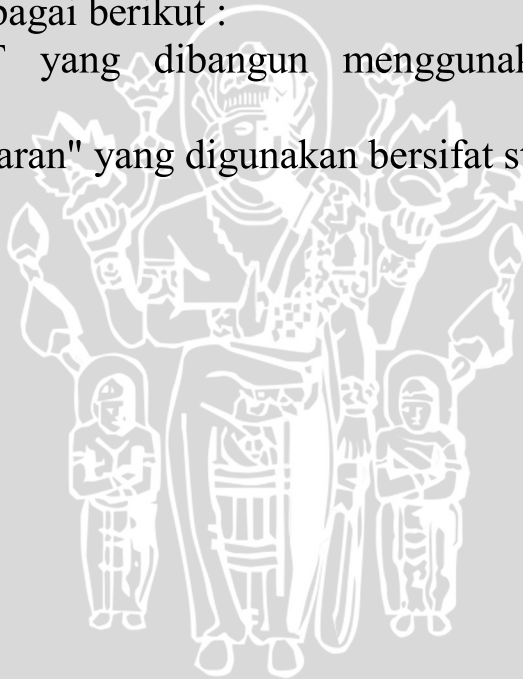
Berdasarkan pada permasalahan yang telah diuraikan di atas, sehingga tujuan dari penulisan skripsi ini adalah sebagai berikut.

1. Membandingkan metode *K-Means* dengan metode FCM setelah diaplikasikan pada proses *clustering*.
2. Menerapkan metode FCM dan metode *Gradient Descent* pada jaringan RBF.
3. Membandingkan metode RBF standar dan RBF modifikasi setelah diaplikasikan pada prediksi harga saham penutupan dari data perdagangan saham harian PT.Astra, Tbk.

1.4 BATASAN MASALAH

Topik mengenai JST cukup luas sehingga perlu diberikan batasan masalah sebagai berikut :

1. Arsitektur JST yang dibangun menggunakan satu lapisan tersembunyi.
2. Laju "pembelajaran" yang digunakan bersifat statis.





BAB II

TINJAUAN PUSTAKA

2.1 *Data Mining*

Data Mining, sering juga disebut *knowledge discovery in database* (KDD), adalah kegiatan yang meliputi pengumpulan, pemakaian data historis untuk menemukan keteraturan, pola atau hubungan dalam koleksi data berukuran besar. Keluaran dari *Data Mining* ini bisa dipakai untuk memperbaiki pengambilan keputusan di masa depan. Dalam *Data Mining* dikenal dua metode yaitu metode "pembelajaran" terawasi dengan metode "pembelajaran" tidak terawasi.

Dalam metode "pembelajaran" terawasi, dilatih dulu teknik yang digunakan dengan data yang ada. Setelah dilatih dengan cukup data, maka alat yang dipakai (misalnya Jaringan Saraf Tiruan), bisa digunakan untuk meramal (prediksi) keputusan apa yang harus diambil bila data baru masuk dan belum diketahui keluarannya. Dalam kategori ini dikenal dua istilah yang sangat populer yakni *training* ("pembelajaran") dan *testing* (pengujian). Di dalam tahap "pembelajaran", teknik yang digunakan dilatih dengan beberapa data yang dianggap mewakili kondisi tugas yang diteliti. Dari tahap "pembelajaran" ini didapat beberapa parameter yang menentukan model dari teknik yang digunakan. Dalam Jaringan Saraf Tiruan, parameter itu adalah *weight* (bobot) dan bias. Setelah parameter ini ditemukan, maka parameter ini digunakan melakukan tugas peramalan (Santosa, 2007).

Dalam pendekatan metode "pembelajaran" tidak terawasi, metode diterapkan tanpa adanya latihan dan tanpa ada "guru". "Guru" di sini adalah label keluaran dari data. Misalkan ada sekelompok pengamatan atau data tanpa ada label tertentu, maka dalam "pembelajaran" tidak terawasi, diharuskan mengelompokkan data tersebut ke dalam beberapa kelas yang dikehendaki. Ini terutama dilakukan karena data yang ada memang tidak mempunyai label. Label menandai kemana data dikelompokkan. Keluaran dari metode "pembelajaran" tidak terawasi inilah yang dipakai sebagai "guru" dalam proses "pembelajaran" untuk metode "pembelajaran" terawasi.

2.2 Ruang Vektor

Definisi 2.1 Ruang *Euclidean*

Jika n adalah sebuah bilangan bulat positif, maka *ordered- n -tuple* adalah sebuah urutan n bilangan real $(a_1, a_2, \dots, a_n)$. Himpunan semua *ordered- n -tuple* dinamakan ruang *Euclidean* dan dinyatakan dengan $\mathbb{R}^n$ (Anton, 2004).

Definisi 2.2 Norma *Euclidean*

Norma atau panjang suatu vektor $\mathbf{x} = (x_1, x_2, \dots, x_n)$ di dalam $\mathbb{R}^n$, dilambangkan dengan $\|\mathbf{x}\|$ didefinisikan sebagai

$$\|\mathbf{x}\| = \sqrt{\mathbf{x} \cdot \mathbf{x}} = \sqrt{x_1^2 + x_2^2 + \dots + x_n^2} \quad (2.1)$$

yang secara geometris dapat ditafsirkan sebagai jarak dari titik asal ke $\mathbf{x}$. Norma titik $\mathbf{x} - \mathbf{y}$ untuk $\mathbf{y} = (y_1, y_2, \dots, y_n)$ adalah

$$\|\mathbf{x} - \mathbf{y}\| = \sqrt{(\mathbf{x} - \mathbf{y}) \cdot (\mathbf{x} - \mathbf{y})} = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \dots + (x_n - y_n)^2} \quad (2.2)$$

yang secara geometris merupakan jarak titik $\mathbf{x}$ ke $\mathbf{y}$ (Soemantri, 1988).

Definisi 2.3 Fungsi Jarak

Diberikan himpunan X yang tidak kosong, yang elemen-elemennya disebut titik. Didefinisikan fungsi bernilai real tidak negatif d pada $X \times X$. Fungsi d dinamakan fungsi jarak untuk setiap x, y di dalam X jika memenuhi sifat sebagai berikut (Soemantri, 1988):

1. $d(x, y) \geq 0$ (positif).
Tidak ada jarak yang mempunyai nilai negatif.
2. $d(x, y) = 0$ jika dan hanya jika $x = y$.
Jarak antara suatu obyek atau titik dengan obyek atau titik itu sendiri adalah nol.
3. $d(x, y) = d(y, x)$ (simetri).
Jarak dari x ke y sama dengan jarak dari y ke x .
4. $d(x, z) \leq d(x, y) + d(y, z)$ (ketidaksamaan segitiga).

Beberapa macam fungsi jarak di dalam $\mathbb{R}^n$ yang sering digunakan dalam *data mining* adalah (Shihab, 2000):

1. jarak *Euclidean*

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}, \quad (2.3)$$

2. jarak *Manhattan*

$$d(x, y) = \sum_{i=1}^n (|x_i - y_i|), \quad (2.4)$$

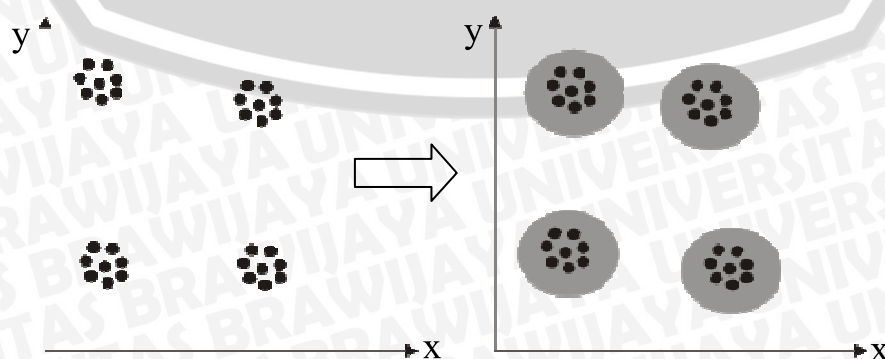
3. jarak *Minkowski*

$$d(x, y) = \left(\sum_{i=1}^n |x_i - y_i|^p \right)^{\frac{1}{p}}; p > 0. \quad (2.5)$$

2.3 Teknik *Clustering*

Tujuan utama dari metode *clustering* adalah pengelompokan sejumlah data/obyek ke dalam *cluster* (grup) sehingga dalam setiap *cluster* berisi data yang semirip mungkin. Dalam *clustering*, program berusaha untuk menempatkan obyek yang mirip (jaraknya dekat) dalam satu *cluster* dan membuat jarak antar *cluster* sejauh mungkin. Ini berarti obyek dalam satu *cluster* sangat mirip satu sama lain dan berbeda dengan obyek dalam *cluster-cluster* yang lain. *Clustering* adalah salah satu teknik "pembelajaran" tidak terawasi di mana pengguna tidak perlu melatih metode tersebut atau dengan kata lain, tidak ada fase "pembelajaran".

Secara umum, algoritma *clustering* dicirikan berdasarkan ukuran kedekatan (Shihab, 2000). Ukuran kedekatan menunjukkan seberapa dekat kedekatan fitur antara 2 data. Gambar 2.1 menunjukkan mengenai pengelompokan data menjadi 4 buah *cluster* didasarkan pada persamaan kriteria jarak antara dua atau lebih obyek (Harjoko, dkk., 2006).



Gambar 2.1 *Cluster*

Definisi 2.4 (Fungsi Keanggotaan)

Jika X adalah kumpulan obyek yang secara umum dinyatakan dengan $\{x\}$. V disebut semesta pembicaraan dan v mewakili elemen-elemen V . Suatu himpunan A dalam semesta pembicaraan V dapat dinyatakan oleh suatu fungsi keanggotaan μ_A untuk setiap v dalam V dan dinyatakan sebagai (Kaya and , 2006)

$$\mu_A : A \rightarrow [0,1] \quad (2.6)$$

2.3.1 Partisi klasik (*Hard Partition*)

Konsep partisi menjadi bagian yang sangat penting bagi proses *clustering*. Tujuan proses *clustering* pada partisi klasik adalah membagi himpunan data X ke dalam l ($l \in \mathbb{N}$) kelompok (grup atau kelas), dengan asumsi bahwa l diketahui (Babuska, 2005). Partisi klasik X dapat didefinisikan sebagai suatu keluarga dari himpunan bagian-himpunan bagian $(A_i | 1 \leq i \leq l) \subset P(X)$, $P(X)$ adalah *power set* dari X , dengan properti sebagai berikut (Bezdek, 1981):

$$\bigcup_{i=1}^l A_i = X \quad (2.7)$$

$$A_i \cap A_j = \emptyset; \quad 1 \leq i \neq j \leq l \quad (2.8)$$

Persamaan (2.7) menunjukkan bahwa union dari himpunan bagian A_i berisi semua data. Himpunan bagian-himpunan bagian harus bersifat disjoint (persamaan 2.8). Dalam bentuk fungsi keanggotaan, suatu partisi dapat direpresentasikan sebagai matriks partisi $U = [\mu_{ik}]_{l \times n}$, $1 \leq i \leq l$; $1 \leq k \leq n$. Baris ke- i pada matriks tersebut berisi nilai keanggotaan μ_i pada himpunan bagian A_i . Elemen-elemen x ($x \in X$) pada matriks U harus memenuhi kondisi sebagai berikut:

$$\mu_{ik}(x) = \begin{cases} 0; & x \notin A \\ 1; & x \in A \end{cases} \quad (2.9)$$

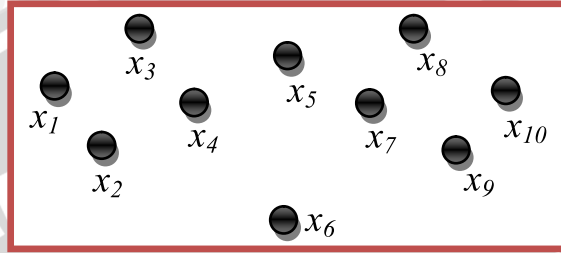
$$\sum_{i=1}^l \mu_{ik}(x) = 1; \quad 1 \leq k \leq n \quad (2.10)$$

Semua kemungkinan partisi dari matriks X disebut dengan *hard partitioning space* (M_{hc}), yang didefinisikan sebagai (Bezdek, 1981):

$$M_{hc} = \left\{ U \in \mathcal{R}^{l \times n} \mid \mu_{ik}(x) \in \{0,1\}, \forall i,k; \sum_{i=1}^l \mu_{ik}(x) = 1, \forall k \right\} \quad (2.11)$$

Contoh 2.1 (partisi klasik)

Misalkan terdapat himpunan $X = \{x_1, x_2, \dots, x_{10}\}$ seperti terlihat pada Gambar 2.2. Data-data tersebut terbagi atas 2 *cluster*, yaitu : *cluster*-1 terdiri atas x_1 sampai x_4 ; dan *cluster*-2 terdiri atas x_7 sampai x_{10} . Dua data, yaitu x_5 dan x_6 , terletak di antara kedua *cluster*.



Gambar 2.2 Cluster Data Menggunakan Partisi Klasik

Matriks partisi $U \in M_{hc}$ terbagi atas 2 himpunan bagian, yaitu:

$$U = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

Baris pertama matriks U berisi nilai keanggotaan data pada himpunan bagian A_1 dan baris kedua matriks U berisi nilai keanggotaan data pada himpunan bagian A_2 . Setiap data hanya terletak pada satu *cluster*. Pada contoh 2.1 terlihat bahwa baik x_5 dan x_6 terletak pada *cluster* A_1 .

2.3.2 Partisi fuzzy (Fuzzy Partition)

Jika pada partisi klasik, suatu data menjadi anggota hanya pada satu *cluster* saja, tidak demikian halnya dengan partisi *fuzzy*. Pada partisi *fuzzy*, nilai keanggotaan suatu data pada suatu *cluster*, μ_{ik} , terletak pada interval $[0,1]$. Matriks partisi pada partisi *fuzzy* memenuhi kondisi sebagai berikut (Shihab, 2000):

$$\mu_{ik}(x) \in [0,1]; \quad 1 \leq i \leq l; \quad 1 \leq k \leq n \quad (2.12)$$

$$\sum_{i=1}^l \mu_{ik}(x) = 1; \quad 1 \leq k \leq n \quad (2.13)$$

Baris ke- i pada matriks partisi U berisi nilai keanggotaan data pada himpunan bagian *fuzzy*. Jumlah fungsi keanggotaan setiap data pada semua *cluster* (jumlah setiap kolom) bernilai 1 (persamaan 2.13).

Semua kemungkinan partisi dari matriks X disebut dengan *fuzzy partitioning space* (M_{fc}), yang didefinisikan sebagai:

$$M_{fc} = \left\{ U \in \mathbb{R}^{l \times n} \mid \mu_{ik}(x) \in [0,1], \forall i,k; \sum_{i=1}^l \mu_{ik}(x) = 1, \forall k \right\} \quad (2.14)$$

Contoh 2.2 (partisi *fuzzy*)

Lihat kembali Contoh 2.1. Matriks partisi $U \in M_{fc}$ terbagi atas 2 himpunan bagian, yaitu:

$$U = \begin{bmatrix} 1 & 1 & 1 & 0,8 & 0,5 & 0,5 & 0,2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0,2 & 0,5 & 0,5 & 0,8 & 1 & 1 & 1 \end{bmatrix}$$

Data x_5 dan x_6 sekarang mempunyai nilai keanggotaan 0,5 pada kedua *cluster*, hal ini menunjukkan bahwa x_5 dan x_6 terletak antara 2 *cluster*. Dari contoh tersebut dapat dilihat bahwa dengan menggunakan partisi *fuzzy*, dapat dimungkinkan suatu data menjadi anggota 2 *cluster*, yang masing-masing keberadaannya dapat dilihat dari nilai keanggotaannya pada setiap *cluster* (Harjoko, dkk., 2006).

2.3.3 *K-Means clustering*

Teknik *clustering* yang paling sederhana dan umum dikenal adalah *K-means clustering*. Dalam teknik ini, program ingin mengelompokkan obyek ke dalam l kelompok atau *cluster*. Untuk melakukan *clustering* ini, nilai l harus ditentukan terlebih dahulu. *K-means clustering* menggunakan partisi klasik dalam mengalokasikan kembali data ke dalam masing-masing *cluster* (Santosa, 2007).

Cara kerja *K-means clustering* pada awalnya mengambil sebagian dari banyaknya komponen dari populasi untuk dijadikan pusat *cluster* awal. Pada langkah ini pusat *cluster* dipilih secara acak dari sekumpulan populasi data. Masing-masing komponen diuji di dalam populasi data dan memindahkan komponen tersebut ke salah satu pusat *cluster* yang telah didefinisikan tergantung dari jarak minimum antar komponen dengan tiap-tiap pusat *cluster*. Posisi pusat *cluster* dihitung kembali sampai semua komponen data digolongkan ke dalam tiap-tiap pusat *cluster* dan terakhir akan terbentuk posisi pusat *cluster* baru (Shihab, 2000).

Algoritma *K-means clustering* diberikan sebagai berikut (Teknomo, 2007):

Input : X (matriks $n \times m$), $l (\geq 2)$

Output : V (pusat *cluster*)

Langkah-langkah :

Langkah 1 : Inisialisasi pusat *cluster* awal.

2 : Ulangi langkah 3-5 hingga tidak ada perpindahan anggota antara tiap *cluster*.

3 : Hitung jarak antara tiap data dengan pusat *cluster* dengan menggunakan persamaan (2.3).

4 : Kelompokkan data berdasarkan jarak minimum.

5 : Hitung pusat *cluster*, V , untuk setiap *cluster*

$$V = \frac{\sum_{k=1}^n x_k}{n}, \quad n = \text{jumlah data.} \quad (2.15)$$

Berdasarkan cara kerjanya, *K-means clustering* memiliki karakteristik (Shihab, 2000):

1. proses *clustering* sangat cepat,
2. sangat sensitif pada pembangkitan *centroids* awal secara random,
3. hasil *clustering* bersifat selalu berubah-ubah, terkadang baik, terkadang jelek,
4. sangat sulit untuk mencapai global optimum.

2.3.4 Fuzzy C-Means (FCM) clustering

Metode FCM adalah suatu teknik pengelompokkan data yang mana keberadaan tiap-tiap data dalam suatu *cluster* ditentukan oleh fungsi keanggotaan. Teknik ini pertama kali diperkenalkan oleh Jim Bezdek pada tahun 1981. Konsep dasar FCM, pertama kali adalah menentukan pusat *cluster* yang menandakan lokasi rata-rata untuk tiap-tiap *cluster*. Pada kondisi awal, pusat *cluster* ini masih belum akurat. Tiap-tiap data, memiliki fungsi keanggotaan untuk tiap-tiap *cluster*. Dengan cara memperbaiki pusat *cluster* dan nilai keanggotaan tiap-tiap data secara berulang, maka dapat dilihat bahwa pusat *cluster* akan bergerak menuju lokasi yang tepat. Perulangan ini didasarkan pada minimisasi fungsi objektif (Gelley, 2000). Fungsi objektif yang digunakan pada FCM adalah (Ross, 2005):

$$J_w(U, V) = \sum_{k=1}^n \sum_{i=1}^l (\mu_{ik})^w d_{ik}^2 \quad (2.16)$$

di mana

- l adalah jumlah *cluster* yang berada di dalam X ,
- n adalah jumlah data yang diproses,
- w adalah pangkat (pembobot), $w \in [1, \infty)$,
- X adalah data yang diproses,

$$X = \begin{bmatrix} x_{11} & \cdots & x_{1m} \\ \vdots & & \vdots \\ x_{n1} & \cdots & x_{nm} \end{bmatrix} \quad (2.17)$$

- m adalah jumlah variabel (kriteria),
- U adalah matriks partisi awal,

$$U = \begin{bmatrix} \mu_{11}(x_1) & \mu_{12}(x_2) & \cdots & \mu_{1l}(x_n) \\ \mu_{21}(x_1) & \mu_{22}(x_2) & \cdots & \mu_{2l}(x_n) \\ \vdots & & & \vdots \\ \mu_{n1}(x_1) & \mu_{n2}(x_2) & \cdots & \mu_{nl}(x_n) \end{bmatrix} \quad (2.18)$$

(matriks partisi awal biasanya dipilih secara acak)

- V adalah matriks pusat *cluster*,

$$V = \begin{bmatrix} v_{11} & \cdots & v_{1l} \\ \vdots & & \vdots \\ v_{m1} & \cdots & v_{ml} \end{bmatrix} \quad (2.19)$$

- d_{ik} adalah fungsi jarak. Pada skripsi ini, fungsi jarak yang digunakan ialah fungsi *Euclidean distance* pada persamaan (2.3).

Lebih lanjut, dalam proses meminimumkan fungsi objektif algoritma FCM, ada beberapa hal yang perlu diperhatikan. Fungsi objektif J_w merupakan fungsi *non linear* di mana terdapat banyak variabel dalam perhitungannya sehingga komponen U dan V tersebut harus dioptimasi.

Di antara metode optimasi yang lazim digunakan adalah konsep turunan. Ketika suatu fungsi tergantung kepada lebih dari satu

variabel, digunakanlah konsep turunan parsial. Turunan parsial dapat dipandang secara informal sebagai turunan dari suatu fungsi yang melibatkan semuanya dengan menahan satu variabel sebagai suatu konstanta. Turunan parsial direpresentasikan sebagai $\partial / \partial x$.

Berikut ini, diberikan teorema mengenai penjabaran dari konsep turunan parsial terhadap fungsi objektif J_w .

Teorema 2.1

Jika $d_{ik} > 0, \forall i, k; w > 1$, dan X setidaknya memiliki l elemen, maka $\mu_{ik} \in U$ dan $v_{ij} \in V$ dapat meminimumkan fungsi objektif J_w jika (Pal, 2005):

$$\mu_{ik} = \left[\sum_{j=1}^l \left(\frac{d_{ik}}{d_{jk}} \right)^{2/(w-1)} \right]^{-1}; \quad 1 \leq i \leq l, \quad 1 \leq k \leq n \quad (2.20)$$

$d_{ik} = \|x_k - v_i\|, \quad d_{jk} = \|x_k - v_j\|$

dan

$$v_{ij} = \frac{\sum_{k=1}^n (\mu_{ik})^w x_{kj}}{\sum_{k=1}^n (\mu_{ik})^w}; \quad 1 \leq i \leq l; \quad 1 \leq j \leq m \quad (2.21)$$

Bukti:

1. Pencarian matriks partisi U

$U_t = F_\partial(V_{t-1})$, merupakan nilai turunan J_w terhadap U saat nilai V dianggap konstan. Nilai U diselesaikan dengan metode *lagrange* di mana pada rumusan fungsi objektif ditambahkan nilai perkalian faktor *lagrange* dengan nilai syaratnya,

$$\frac{\partial}{\partial \mu_i} \left(\left(\sum_{i=1}^l \mu_i^w \|x - v_i\|^2 \right) + \lambda \left(1 - \sum_{i=1}^l \mu_i \right) \right) = w \mu_i^{w-1} (\|x - v_i\|^2) - \lambda \quad (2.22)$$

dengan mengatur nilai dari turunan ini bernilai sama dengan nol, maka didapatkan nilai μ sebagai berikut:

$$\frac{\partial}{\partial \mu_i} \left(\left(\sum_{i=1}^l \mu_i^w \|x - v_i\|^2 \right) + \lambda \left(1 - \sum_{i=1}^l \mu_i \right) \right) = 0$$

$$w\mu_i^{w-1}(\|x - v_i\|^2) - \lambda = 0$$

$$\mu_i = \left(w\|x - v_i\|^2 \cdot \frac{1}{\lambda} \right)^{-1/(w-1)} \quad (2.23)$$

selanjutnya, dibutuhkan perhitungan nilai λ . Penyelesaian untuk λ sebagai perhitungan syarat dihasilkan berdasarkan perhitungan berikut:

$$\sum_{i=1}^l \mu_i = \sum_{i=1}^l \left(\frac{w\|x - v_i\|^2}{\lambda} \right)^{-1/(w-1)} \quad (2.24)$$

dari persamaan (2.13) di mana nilai $\sum_{i=1}^l \mu_i = 1$ maka

$$\sum_{i=1}^l \left(\frac{w\|x - v_i\|^2}{\lambda} \right)^{-1/(w-1)} = 1$$

$$\left(\frac{1}{\lambda} \right)^{-1/(w-1)} \sum_{i=1}^l (w\|x - v_i\|^2)^{-1/(w-1)} = 1$$

$$\left(\frac{1}{\lambda} \right)^{-1/(w-1)} = \frac{1}{\sum_{i=1}^l (w\|x - v_i\|^2)^{-1/(w-1)}}$$

$$\left(\frac{1}{\lambda} \right)^{-1/(w-1)} = \left(\sum_{i=1}^l (w\|x - v_i\|^2)^{-1/(w-1)} \right)^{-1} \quad (2.25)$$

dengan mensubstitusikan persamaan (2.25) ke dalam persamaan (2.23), maka:

$$\mu_i = (w\|x - v_i\|^2)^{-1/(w-1)} \left(\frac{1}{\lambda} \right)^{-1/(w-1)}$$

$$= (w\|x - v_i\|^2)^{-1/(w-1)} \left(\sum_{j=1}^l (w\|x - v_j\|^2)^{-1/(w-1)} \right)^{-1}$$

$$\begin{aligned}
 &= \left(\left(w \|x - v_i\|^2 \right)^{1/(w-1)} \sum_{j=1}^l \left(w \|x - v_j\|^2 \right)^{-1/(w-1)} \right)^{-1} \\
 &= \left(\left(w \|x - v_i\|^2 \right)^{1/(w-1)} \sum_{j=1}^l \frac{1}{\left(w \|x - v_j\|^2 \right)^{1/(w-1)}} \right)^{-1} \\
 &= \left(\sum_{j=1}^l \frac{\left(w \|x - v_i\|^2 \right)^{1/(w-1)}}{\left(w \|x - v_j\|^2 \right)^{1/(w-1)}} \right)^{-1} \\
 &= \left(\sum_{j=1}^l \left(\frac{\|x - v_i\|}{\|x - v_j\|} \right)^{2/(w-1)} \right)^{-1} \\
 \mu_{ik} &= \left(\sum_{j=1}^l \left(\frac{\|x_k - v_i\|}{\|x_k - v_j\|} \right)^{2/(w-1)} \right)^{-1}, \forall i, k \\
 \mu_{ik} &= \left[\sum_{j=1}^l \left(\frac{d_{ik}}{d_{jk}} \right)^{2/(w-1)} \right]^{-1}, \forall i, k
 \end{aligned} \tag{2.26}$$

2. Pencarian pusat *cluster* V

Berikut ini rumusan turunan dari fungsi objektif terhadap pusat *cluster*:

$$\begin{aligned}
 J_w(U, V) &= \sum_{k=1}^n \sum_{i=1}^l (\mu_{ik})^w \|x_k - v_i\|^2 \\
 J_w(U, V) &= \sum_{k=1}^n \sum_{i=1}^l (\mu_{ik})^w ((x_k - v_i) \cdot (x_k - v_i)) \\
 J_w(U, V) &= \sum_{k=1}^n \sum_{i=1}^l (\mu_{ik})^w (x_k^2 - 2x_k v_i + v_i^2)
 \end{aligned}$$

$$J_w(U, V) = \sum_{k=1}^n \sum_{i=1}^l (\mu_{ik})^w \left(\sum_{j=1}^m x_{kj}^2 - 2 \left(\sum_{j=1}^m x_{kj} v_{ij} \right) + \sum_{j=1}^m v_{ij}^2 \right)$$

$$\frac{\partial J_w(U, V)}{\partial v_{ij}} = \sum_{k=1}^n \sum_{i=1}^l (\mu_{ik})^w (0 - 2x_{kj} + 2v_{ij}) \quad (2.27)$$

dengan mengatur nilai dari turunan ini bernilai sama dengan nol, maka didapatkan nilai v sebagai berikut:

$$\frac{\partial J_w(U, V)}{\partial v_{ij}} = 0$$

$$2 \sum_{k=1}^n \sum_{i=1}^l (\mu_{ik})^w (v_{ij} - x_{kj}) = 0$$

$$2 \sum_{k=1}^n \sum_{i=1}^l (\mu_{ik})^w v_{ij} - 2 \sum_{k=1}^n \sum_{i=1}^l (\mu_{ik})^w x_{kj} = 0$$

$$\sum_{k=1}^n \sum_{i=1}^l (\mu_{ik})^w v_{ij} = \sum_{k=1}^n \sum_{i=1}^l (\mu_{ik})^w x_{kj}$$

$$\sum_{i=1}^l v_{ij} \sum_{k=1}^n (\mu_{ik})^w = \sum_{i=1}^l \sum_{k=1}^n (\mu_{ik})^w x_{kj}$$

$$v_{ij} = \frac{\sum_{k=1}^n (\mu_{ik})^w x_{kj}}{\sum_{k=1}^n (\mu_{ik})^w} \quad (2.28)$$

Perhitungan di atas mencapai optimal pada saat pusat *cluster* tidak lagi berubah setelah dilakukan iterasi terhadap komponen fungsi objektif. Apabila pusat *cluster* tidak berubah mengindikasikan bahwa titik tersebut merupakan titik terdekat yang dapat merepresentasikan anggota-anggota *cluster* yang lain.

Algoritma FCM diberikan sebagai berikut (Zimmerman, 1991);(Yan,1994);(Ross,2005):

Input : X (matriks $n \times m$)

$l (\geq 2)$

w

Maksimum iterasi

ξ (kriteria penghentian, nilai positif yang sangat kecil)

Output : V (pusat *cluster*)

Δ (error)

Langkah-langkah :

Langkah 1 : {Inisialisasi awal}

$t=1; \Delta = 1.$

2 : Bentuk matriks partisi awal, U^0 .

3 : Jika $\Delta \geq \xi$ lakukan langkah 4-7.

4 : Hitung pusat *cluster*, V , untuk setiap *cluster* dengan menggunakan persamaan (2.21).

5 : Perbaiki fungsi keanggotaan setiap data pada setiap *cluster* (perbaiki matriks partisi) dengan menggunakan persamaan (2.20).

6 : Tentukan kriteria berhenti, yaitu perubahan matriks partisi pada iterasi sekarang dengan iterasi sebelumnya, sebagai berikut.

$$\Delta = \|U^t - U^{t-1}\| \quad (2.29)$$

7 : $t = t + 1.$

2.4 Jaringan Saraf Tiruan

Haykin (1999) menyatakan bahwa jaringan saraf adalah suatu sistem pemroses besar yang terdiri atas unit-unit pemroses sederhana yang memiliki kemampuan menyimpan pengetahuan/pengalaman dan membuatnya menjadi dapat digunakan dalam proses berikutnya. Jaringan saraf menyerupai sistem kerja otak dengan ketentuan:

1. pengetahuan diperoleh jaringan dari lingkungannya melalui suatu proses "pembelajaran",
2. kekuatan hubungan antar neuron, disebut bobot sinapsis, digunakan untuk menyimpan pengetahuan tersebut.

Jaringan saraf tiruan (JST) menurut Fausett (1994) adalah sistem pemrosesan informasi yang memiliki performansi tertentu yang berhubungan dengan jaringan saraf biologis. JST dikembangkan sebagai generalisasi model matematika dari pemahaman manusia yang didasarkan pada asumsi (Siang, 2005):

1. pemrosesan informasi terjadi pada banyak elemen sederhana yang disebut neuron,
2. sinyal dikirimkan antar neuron melalui jaringan penghubung,
3. masing-masing jaringan penghubung mempunyai bobot yang akan memperkuat atau memperlemah sinyal,

4. untuk menentukan keluaran, setiap neuron menggunakan fungsi aktivasi yang dikenakan pada jumlahan masukan yang diterima.

Adapun cara "belajar" JST sebagai berikut: ke dalam JST dimasukkan informasi yang sebelumnya telah diketahui hasil keluarannya. Informasi ini dimasukkan lewat node-node atau unit-unit masukan. Bobot-bobot antarkoneksi dalam suatu arsitektur diberi nilai awal dan kemudian JST dijalankan. Bobot-bobot ini digunakan untuk "belajar" dan mengingat suatu informasi. Pengaturan bobot dilakukan secara terus-menerus dan dengan menggunakan kriteria tertentu sampai diperoleh keluaran yang diharapkan.

Hal-hal yang ingin dicapai dengan mengajari JST adalah untuk mencapai keseimbangan antara kemampuan *memorisasi* dan *generalisasi*. Yang dimaksud dengan kemampuan *memorisasi* adalah kemampuan JST untuk memanggil kembali secara sempurna sebuah pola yang telah dipelajari. Kemampuan *generalisasi* adalah kemampuan JST untuk menghasilkan respons yang bisa diterima terhadap pola-pola masukan yang serupa (namun tidak identik) dengan pola-pola yang sebelumnya telah dipelajari. Hal ini sangat bermanfaat bila pada suatu saat ke dalam JST dimasukkan informasi baru yang belum pernah dipelajari, maka JST masih tetap dapat memberikan tanggapan yang baik, memberikan keluaran yang paling mendekati (Puspitaningrum, 2006).

Lapisan-lapisan penyusun JST dapat dibagi menjadi tiga, yaitu lapisan masukan, lapisan tersembunyi dan lapisan keluaran. JST ditentukan oleh 3 hal (Siang, 2005):

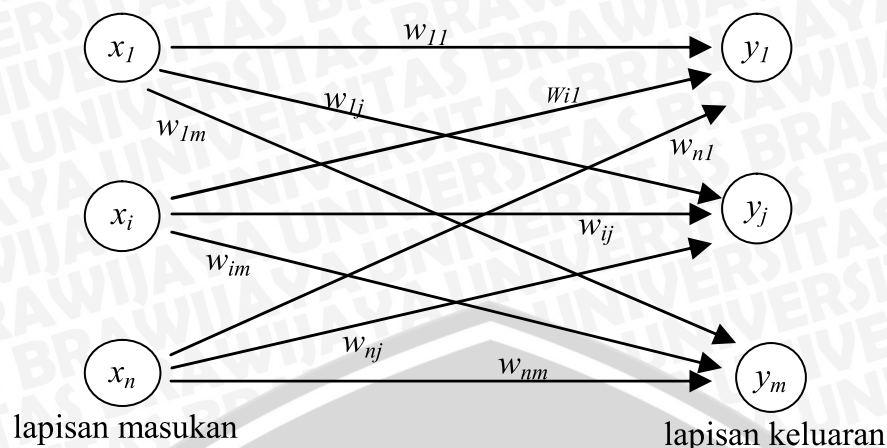
1. pola hubungan antar neuron (disebut arsitektur jaringan),
2. metode untuk menentukan bobot penghubung (disebut metode "pembelajaran"),
3. fungsi aktivasi.

2.4.1 Arsitektur Jaringan

Beberapa arsitektur jaringan yang sering dipakai dalam JST antara lain:

1. Jaringan Lapisan Tunggal

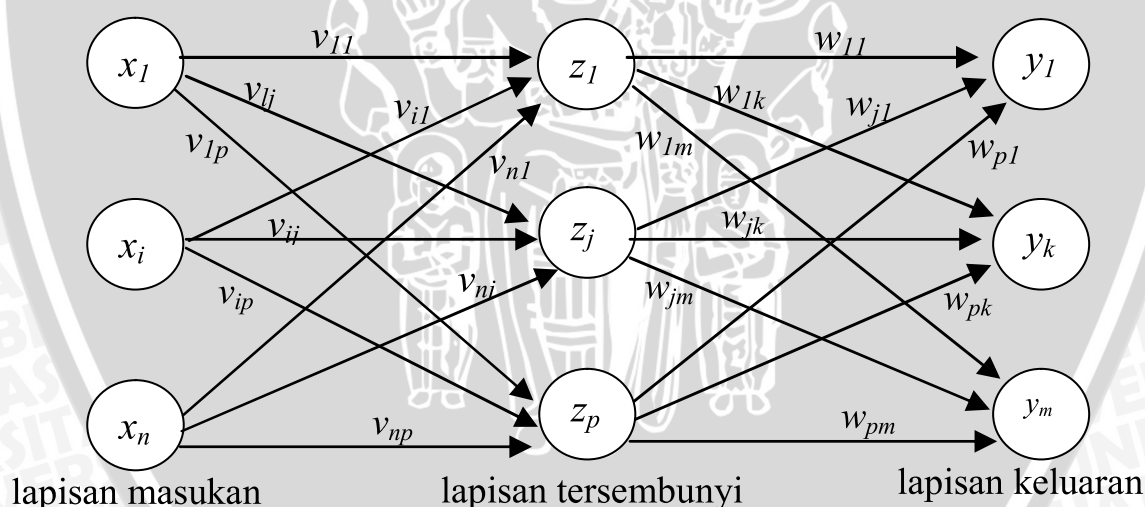
Dalam jaringan ini, neuron-neuron dapat dikelompokkan menjadi dua lapisan yaitu lapisan masukan dan lapisan keluaran. Pada Gambar 2.3 menunjukkan lapisan masukan yang memiliki n unit neuron dan lapisan keluaran yang memiliki m unit neuron.



Gambar 2.3 Arsitektur Jaringan Lapisan Tunggal

2. Jaringan Multi Lapisan

Jaringan ini merupakan perluasan dari jaringan lapisan tunggal. Diantara lapisan masukan dan lapisan keluaran ditambahkan lapisan yang lain yang disebut sebagai lapisan tersembunyi, jumlah lapisan tersembunyi bisa lebih dari satu. Pada Gambar 2.4 ditunjukkan bahwa jaringan memiliki n unit masukan dan m unit keluaran serta sebuah lapisan tersembunyi yang terdiri dari p buah unit.



Gambar 2.4 Arsitektur Jaringan Multi Lapisan

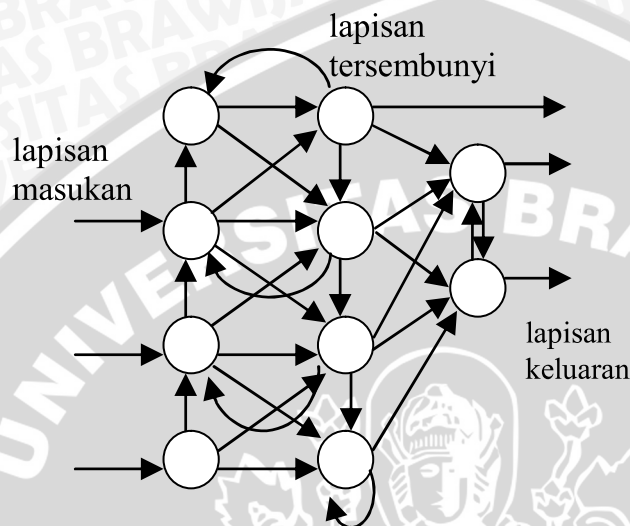
di mana :

- i adalah indeks untuk unit masukan,
- j adalah indeks untuk unit tersembunyi,
- k adalah indeks untuk unit keluaran,
- x adalah unit masukan,
- y adalah unit keluaran,
- z adalah unit tersembunyi,

w_{jk} adalah bobot antara lapisan tersembunyi dan lapisan keluaran,
 v_{ij} adalah bobot antara lapisan masukan dan lapisan tersembunyi.

3. Jaringan *Reccurent*

Model jaringan *reccurent* mirip dengan jaringan lapisan tunggal ataupun ganda. Hanya saja, ada unit keluaran yang memberikan sinyal pada unit masukan (*feedback loop*) (Siang, 2005).



Gambar 2.5 Arsitektur Jaringan *Recurrent*

2.4.2 Metode "Pembelajaran"

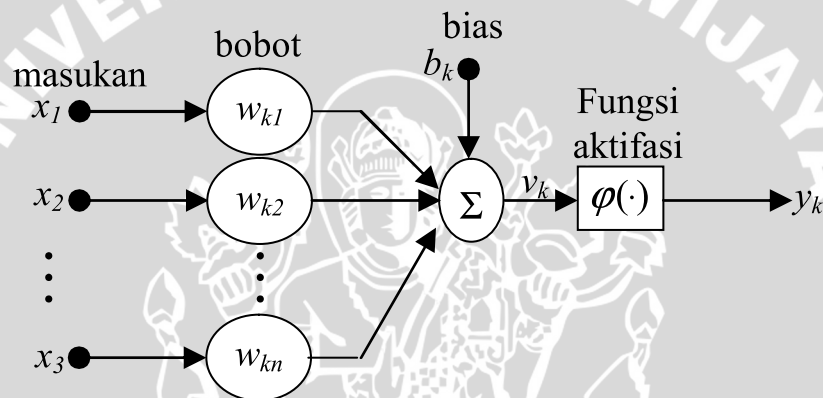
Metode "pembelajaran" digunakan untuk mengajari JST, yaitu dengan cara mengajarnya dengan contoh-contoh kasus/pola sampai JST berhasil mengenali pola tersebut. Setiap kali keluaran yang dihasilkan oleh jaringan tidak sesuai dengan target yang diharapkan maka setiap kali pula bobotnya diperbarui. Hal ini terus menerus dilakukan sampai tidak ada lagi bobot yang berubah. Bobot-bobot terakhir yang diperoleh pada saat "pembelajaran" JST ini yang digunakan pada saat pengaplikasian.

Berdasarkan cara memodifikasi bobotnya, ada 3 macam metode "pembelajaran" yang dikenal yaitu dengan pengawasan (*supervised learning*), tanpa pengawasan (*unsupervised learning*) atau merupakan gabungan keduanya ("pembelajaran" hibrida). Paradigma "pembelajaran" hibrida merupakan kombinasi dari "pembelajaran" terawasi dengan "pembelajaran" tidak terawasi. Sebagian dari bobot-bobotnya ditentukan melalui "pembelajaran" terawasi dan sebagian lainnya melalui "pembelajaran" tidak terawasi (Puspitaningrum, 2006).

Dalam skripsi ini metode "pembelajaran" yang digunakan adalah "pembelajaran" hibrida dengan model jaringan *Radial Basis Function* (RBF) yang dibahas pada subbab 2.5.

2.4.3 Fungsi Aktivasi

Seperti halnya otak manusia, JST juga terdiri dari beberapa neuron, dan ada hubungan antara neuron-neuron tersebut. Neuron adalah unit pemroses informasi dasar untuk operasi suatu jaringan saraf. Neuron-neuron tersebut mentransformasikan informasi yang diterima melalui sambungan keluarannya menuju ke neuron-neuron yang lain. Pada JST, hubungan ini dikenal dengan nama bobot. Informasi tersebut disimpan pada suatu nilai tertentu pada bobot tersebut.



Gambar 2.6 Struktur Neuron JST

Informasi dikirim ke neuron dengan bobot kedatangan tertentu. Informasi ini diproses oleh suatu fungsi perambatan yang akan menjumlahkan nilai-nilai semua bobot yang datang. Hasil penjumlahan ini kemudian dibandingkan dengan suatu nilai ambang (*threshold*) tertentu melalui fungsi aktivasi setiap neuron.

Fungsi aktivasi digunakan untuk menentukan keluaran dari JST. Di mana sebelum melakukan proses pengaktifan, neuron-neuron dihitung melalui kombinasi *linear* antara masukan dengan bobotnya ditambah bobot dari luar atau lebih dikenal dengan bias.

Gambar 2.6 menunjukkan model suatu neuron yang menjadi dasar perancangan JST (Haykin, 1999), dengan $x_1, x_2, \dots, x_n$ adalah masukan, $w_{k1}, w_{k2}, \dots, w_{kn}$ adalah bobot neuron k , b_k adalah bias, $\phi(\cdot)$ adalah fungsi aktifasi dan y_k adalah keluaran.

Secara matematis proses yang terjadi pada neuron k ditulis dengan persamaan berikut:

$$u_k = \sum_{j=1}^n w_{kj} x_j \quad (2.30)$$

dan

$$y_k = \phi(v_k) \quad (2.31)$$

dengan

$$v_k = u_k + b_k \quad (2.32)$$

2.5 Jaringan Radial Basis Function (RBF)

Jaringan RBF didesain untuk membentuk pemetaan *non linear* dari unit lapisan masukan ke unit lapisan tersembunyi dan pemetaan *linear* dari lapisan tersembunyi ke lapisan keluaran. Oleh karena itu, pada jaringan RBF dilakukan pemetaan masukan dari ruang berdimensi p ke keluaran ruang berdimensi 1.

$$s: \mathcal{R}^p \rightarrow \mathcal{R}^1 \quad (2.33)$$

Jaringan RBF dapat dinyatakan dengan

$$F(x) = \sum_{i=1}^N w_i \phi(\|x - x_i\|) \quad (2.34)$$

dengan $\{\phi(\|x - x_i\|) \mid i = 1, 2, \dots, N\}$ adalah himpunan fungsi *non linear* yang disebut fungsi Radial Basis, dan $\|\cdot\|$ adalah norma *Euclidean* (Rivera, dkk., 2006).

Ada beberapa fungsi Radial Basis diantaranya adalah (Gutmann, 2001)

1. fungsi *Thin Plate Spline*

$$\phi(x) = (x - \mu)^2 \log(x - \nu) \quad , \quad (2.35)$$

2. fungsi Multikuadratik

$$\phi(x) = [(x - \nu)^2 + \sigma^2]^{1/2} \quad , \quad (2.36)$$

3. fungsi Invers Multikuadratik

$$\phi(x) = [(x - \nu)^2 + \sigma^2]^{-1/2} \quad , \quad (2.37)$$

4. fungsi Gaussian

$$\phi(x) = \exp[-(x - \nu)^2 / \sigma^2] \quad . \quad (2.38)$$

Proses "pembelajaran" jaringan RBF dilakukan untuk perbaikan pusat *cluster* yang menghubungkan unit-unit lapisan masukan dengan lapisan tersembunyi dan perbaikan bobot yang menghubungkan unit-unit lapisan tersembunyi dengan unit-unit lapisan keluaran dari JST. Dengan kata lain, jaringan RBF mengalami proses "pembelajaran" terpisah, yaitu "pembelajaran" tidak terawasi pada lapisan masukan dan "pembelajaran" terawasi pada lapisan keluaran.

Untuk "pembelajaran" tidak terawasi, metode yang biasa digunakan adalah metode *clustering*. Secara terpisah, metode "pembelajaran" terawasi untuk menghitung bobot *linear* dari lapisan keluaran, dilakukan dengan metode *Gradient Descent*. Keluaran dari lapisan tersembunyi pada jaringan RBF akan menjadi masukan dari algoritma *Gradient Descent*.

Algoritma metode "pembelajaran" Jaringan RBF adalah (Anam, 2005):

1. inisialisasi pembobot

Pusat-pusat *cluster* pada lapisan tersembunyi ditentukan dengan algoritma *clustering*. Bobot pada lapisan keluaran diinisialisasi sebagai sebarang nilai yang kecil. Pada langkah ini bertujuan untuk menentukan pusat *cluster* (v_j) dan *spread* (σ_j) dari fungsi Radial Basis ϕ_j ,

2. perhitungan aktivasi

Perhitungan aktivasi menggunakan fungsi Gaussian pada persamaan (2.38),

3. menentukan pembobot antara lapisan tersembunyi dan lapisan keluaran

Pada tahap ini dilakukan proses perubahan pembobot dengan

$$w_{ji}(t+1) = w_{ji}(t) + \Delta w_{ji} . \quad (2.39)$$

Dimana $w_{ji}(t)$ adalah bobot dari unit ke- i ke unit ke- j pada waktu ke- t (atau iterasi ke- t) dan Δw_{ji} adalah perubahan bobot,

4. mengulangi iterasi sampai konvergen.

2.6 Optimasi

Optimasi merupakan suatu cara untuk mengambil keputusan. Di dalam matematika, optimasi diartikan sebagai suatu metode untuk meminimumkan atau memaksimumkan fungsi tujuan dengan kendala

berdasarkan variabel-variabel yang dicari. Permasalahan optimasi dapat ditulis sebagai

$$\min_{\mathbf{x} \in \mathcal{R}^n} f(\mathbf{x}) \quad (2.40)$$

dengan kendala berbentuk
$$\begin{cases} c_i(\mathbf{x}) = 0, & i \in \xi \\ c_i(\mathbf{x}) \geq 0, & i \in \xi \end{cases}$$

di mana $\mathbf{x}$ adalah vektor dari variabel $(x_1, x_2, \dots, x_n)$

f adalah fungsi objektif

c_i adalah fungsi bernilai skalar dari kriteria $\mathbf{x}$

ξ adalah himpunan dari indeks

(Nocedal dan Stephen, 1999).

Definisi 2.5 (Minimum Lokal)

Diberikan $\mathbf{x}^* \in S \subseteq \mathcal{R}^n$, $\mathbf{x}^*$ dikatakan sebagai minimum lokal dari fungsi f dalam S jika terdapat persekitaran $N(\mathbf{x}^*)$ di dalam S sedemikian sehingga jika $\mathbf{x} \in N(\mathbf{x}^*)$ maka berlaku $f(\mathbf{x}^*) \leq f(\mathbf{x})$ (Leader, 2004).

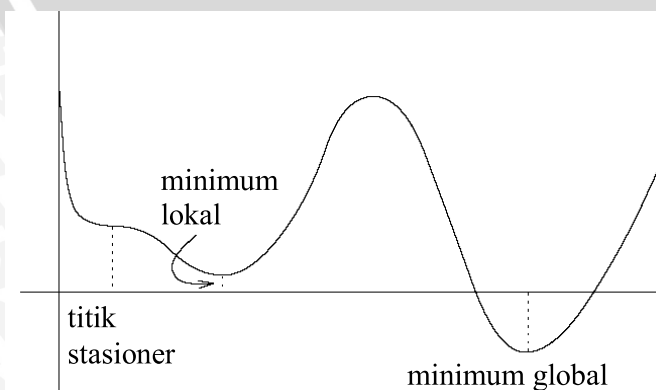
Definisi 2.6 (Minimum Global)

Diberikan $\mathbf{x}^* \in S \subseteq \mathcal{R}^n$, $\mathbf{x}^*$ disebut sebagai minimum global dari fungsi f jika untuk semua $\mathbf{x} \in S$ berlaku

$$f(\mathbf{x}^*) \leq f(\mathbf{x}) \quad (2.41)$$

(Bronshtein, dkk. 2007).

Untuk lebih memahami mengenai minimum lokal dan minimum global dijelaskan pada Gambar 2.7.



Gambar 2.7 Minimum Lokal dan Minimum Global

Definisi 2.7 (*Gradient*)

Misalkan $f(\mathbf{x})$ adalah fungsi dari $\mathbf{x}$, di mana $\mathbf{x} = (x_1, x_2, \dots, x_n)$ suatu vektor dalam $\mathbb{R}^n$, sedemikian sehingga $\frac{\partial f(\mathbf{x})}{\partial x_k}$ ada untuk $k = 1, 2, \dots, n$. *Gradient* dari $f(\mathbf{x})$, dinotasikan dengan $\nabla f(\mathbf{x})$, adalah vektor kolom (Mathews dan Kurtis, 2004).

$$\nabla f(\mathbf{x}) = \left(\frac{\partial f(\mathbf{x})}{\partial x_1}, \frac{\partial f(\mathbf{x})}{\partial x_2}, \dots, \frac{\partial f(\mathbf{x})}{\partial x_n} \right)^T \quad (2.42)$$

Dalam skripsi ini, kajian lebih difokuskan pada permasalahan optimasi *non linear* tanpa kendala, sehingga diperoleh optimasi yang berbentuk

$$\min f(x), x \in \mathbb{R}^n \quad (2.43)$$

Dimana $f: \mathbb{R}^n \rightarrow \mathbb{R}$ merupakan suatu fungsi yang dapat diturunkan dan kontinu. Kondisi yang diperlukan untuk memperoleh titik $\mathbf{x}^*$ menjadi optimal adalah

$$\nabla f(\mathbf{x}^*) = 0 \quad (2.44)$$

Pada umumnya permasalahan tersebut dipecahkan menggunakan algoritma perulangan yang membentuk sebuah barisan dari titik-titik $x_0, x_1, \dots \in \text{dom } f$, dengan $f(x_k) \rightarrow f^*$ dan $k \rightarrow \infty$.

Secara umum untuk bisa menyelesaikan persamaan (2.44) dibentuk barisan

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k \quad (2.45)$$

di mana $\alpha_k > 0$ adalah ukuran langkah

$\mathbf{d}_k$ adalah arah penurunan

(Boyd and Lieven, 2004).

2.6.1 Metode *Gradient Descent*

Metode *Gradient Descent* merupakan salah satu metode untuk meminimumkan suatu masalah optimasi tanpa kendala. Suatu fungsi dikatakan menurun jika memenuhi

$$f(\mathbf{x}_{k+1}) < f(\mathbf{x}_k) \quad (2.46)$$

dengan arah penurunan yang harus memenuhi

$$\nabla f(\mathbf{x}_k)^T \mathbf{d}_k < 0 \quad (2.47)$$

Pada metode *Gradient Descent* arah penurunan dari fungsi dipilih dengan mengambil *gradient* bernilai negatif atau $\mathbf{d}_k = -\nabla f(\mathbf{x}_k)$.

(Boyd dan Lieven, 2004).

Algoritma metode *Gradient Descent* adalah (Buchanan dan Peter, 1992):

Input : $\mathbf{x}_0$ dan kriteria pemberhentian ε

Output : $\mathbf{x}_{k+1}$

Langkah-langkah :

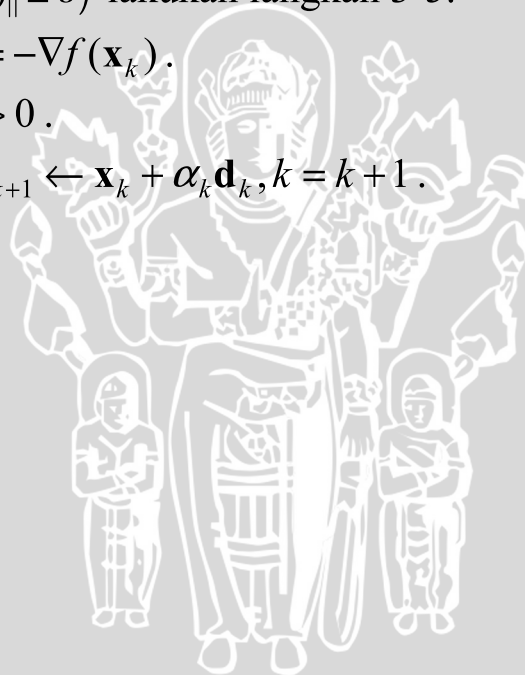
Langkah 1: Inisialisasi $\mathbf{x}_0 \in \mathcal{R}^n$.

2 : Jika $(\|\nabla f(\mathbf{x}_k)\| \geq \varepsilon)$ lakukan langkah 3-5.

3 : Pilih $\mathbf{d}_k = -\nabla f(\mathbf{x}_k)$.

4 : Pilih $\alpha_k > 0$.

5 : Hitung $\mathbf{x}_{k+1} \leftarrow \mathbf{x}_k + \alpha_k \mathbf{d}_k, k = k + 1$.



BAB III METODOLOGI

3.1 Jenis dan Sumber Data

Data yang dipakai dalam skripsi ini adalah data sekunder yang diambil dan diolah dari Bursa Efek Indonesia (BEI). Pemilihan BEI sebagai sumber utama skripsi ini karena merupakan pasar saham terbesar dan paling representatif di Indonesia. Data saham harian yang digunakan adalah data saham harian dari PT.Astra, Tbk. Data saham harian diperoleh dari situs <http://finance.yahoo.com>.

3.2 Metode Pengumpulan Data

Metode pengumpulan data yang digunakan dalam skripsi ini adalah *non-participant observer*, di mana penulis hanya mengamati data yang sudah tersedia tanpa ikut menjadi bagian dari suatu sistem data. Data yang dipakai adalah data saham harian PT.Astra, Tbk periode 2 Januari 2007 sampai dengan 28 Desember 2007.

3.3 Normalisasi Data

Sebelum data saham harian milik PT.Astra, Tbk tahun 2007 diproses ke dalam JST, data tersebut dinormalisasikan terlebih dahulu untuk mempermudah perhitungan. Dalam skripsi ini, total seluruh data yang digunakan berjumlah 241 data, dengan perincian 90% untuk data "pembelajaran" dan 10% sisanya digunakan sebagai data pengujian. Berikut rumus normalisasi yang digunakan

$$x_i^* = \frac{x_i - \mu}{\sqrt{\frac{\sum_{i=1}^n (x_i - \mu)^2}{n - 1}}} \quad (3.1)$$

di mana : i adalah indeks untuk seluruh data,

x_i adalah data yang belum dinormalkan,

x_i^* adalah data yang telah dinormalkan,

μ adalah rata-rata data (Walpole, 1995).

3.4 Membangun JST

3.4.1 Metode *clustering*

1. Jumlah *cluster*

Jumlah *cluster* diinisialisasi di awal secara uji coba. Tujuannya untuk mencari jumlah *cluster* manakah yang paling sesuai untuk meminimumkan *error*. Jumlah *cluster* yang diujicoba adalah 5,7,10 dan 12.

2. Perhitungan jarak

Fungsi jarak yang digunakan ialah fungsi jarak *Euclidean*. *Euclidean* digunakan karena perhitungan jarak dalam fungsi ini merupakan jarak terpendek yang bisa didapatkan antara dua titik yang diperhitungkan.

3.4.2 Jaringan RBF

1. Model arsitektur

a. Unit lapisan masukan

Penentuan banyaknya unit lapisan masukan tidak terlepas dari banyaknya variabel-variabel yang mempengaruhi data saham harian milik PT.Astra, Tbk tahun 2007. Variabel-variabel tersebut antara lain: harga sebelum, harga tertinggi, harga terendah, harga penutupan, volume penjualan dan *stock* indeks. Dalam skripsi ini, variabel-variabel yang digunakan sebagai masukan jaringan berjumlah 5 variabel yaitu harga sebelum, harga tertinggi, harga terendah, volume penjualan dan *stock* indeks.

b. Lapisan tersembunyi

Penentuan banyaknya unit lapisan tersembunyi bergantung pada hasil *clustering*. Jumlah *cluster* yang baik digunakan sebagai jumlah unit lapisan tersembunyi.

c. Lapisan keluaran

Variabel yang dijadikan sebagai keluaran jaringan yaitu harga penutupan, sehingga banyaknya unit lapisan keluaran yang digunakan sebanyak 1 unit.

2. Fungsi aktivasi

Fungsi aktivasi yang digunakan dalam skripsi ini adalah fungsi *Gaussian*. Fungsi *Gaussian* dipilih karena mempunyai sifat lokal, yaitu bila masukan dekat dengan pusat *cluster* maka fungsi akan menghasilkan nilai satu. Bila masukan jauh dari pusat *cluster* maka fungsi akan memberikan nilai nol.

3. *Spread*

Satu parameter yang sangat menentukan pada jaringan RBF adalah besarnya *spread* yang mempengaruhi nilai bobot dan bias pada lapisan tersembunyi. Pada skripsi ini, nilai *spread* diinisialisasi di awal proses "pembelajaran" dengan nilai yang diberikan antara lain: 1,2,3,4 dan 5.

4. Laju "pembelajaran" (α)

Pada skripsi ini, laju "pembelajaran" yang diberikan bersifat statis. Hal ini dimaksudkan untuk melihat bagaimana pengaruh yang diberikan laju "pembelajaran" terhadap kemampuan JST dalam mengenali pola. Dalam hal ini dipilih laju "pembelajaran" (α) antara lain: 0,2; 0,35; 0,5 dan 0,8.

5. Kriteria pemberhentian

Kriteria pemberhentian dari "pembelajaran" JST dengan model jaringan RBF menggunakan metode *K-Means* dan metode FCM adalah jumlah iterasi maksimum sebesar 500. Artinya jika "pembelajaran" JST telah mencapai kriteria tersebut maka proses "pembelajaran" akan berhenti. Dengan memberikan kriteria "pembelajaran" yang sama, maka dapat dibandingkan nilai *error* dari metode *K-Means* dengan metode FCM untuk mencapai target yang diberikan.

3.5 Pengukuran Kinerja

3.5.1 Metode *clustering*

Pengukuran kinerja pada metode *clustering* dilakukan untuk mencari jumlah *cluster* yang paling baik dalam meminimumkan *error*. Jumlah *cluster* yang baik dipenuhi oleh kriteria sebagai berikut:

1. Delta

Delta adalah nilai *error* yang terjadi pada proses *clustering*. Karena metode *K-Means* dengan metode FCM merupakan metode yang berbeda, maka delta yang digunakan juga berbeda.

a. *K-Means*

Untuk *K-Means*, nilai delta merupakan hasil selisih dari perbaikan matriks pusat *cluster*.

$$\Delta = \|V^t - V^{t-1}\| \quad (3.2)$$

di mana :

Δ = delta,

t = iterasi,
 V^t = pusat *cluster* saat iterasi ke t ,
 V^{t-1} = pusat *cluster* saat iterasi ke $t-1$.

b. FCM

Nilai delta dari metode FCM merupakan hasil selisih dari perbaikan matriks partisinya.

$$\Delta = \|U^t - U^{t-1}\| \quad (3.3)$$

di mana :

Δ = delta,
 t = iterasi,
 U^t = matriks partisi saat iterasi ke t ,
 U^{t-1} = matriks partisi saat iterasi ke $t-1$.

2. Waktu

Waktu adalah lamanya proses *clustering* untuk mendapatkan nilai delta yang minimum.

Untuk setiap jumlah *cluster* yang diujicoba, dibandingkan nilai delta dan waktunya. Jumlah *cluster* yang memiliki nilai delta minimum dan waktu yang cepat akan dipilih dalam pengujian data.

3.5.2 Jaringan RBF

Pengukuran kinerja pada jaringan RBF dilakukan untuk menentukan jumlah *spread* dan laju "pembelajaran" yang akan digunakan dalam pengujian data. Pemilihan *spread* dan laju "pembelajaran" berdasarkan kriteria sebagai berikut :

1. *Mean squared error*

Dalam statistik, *Mean Squared Error* (MSE) adalah nilai yang diharapkan dari kuadrat *error*. *Error* yang ada menunjukkan seberapa besar perbedaan hasil target yang diberikan dengan hasil keluaran dari proses JST.

$$MSE = \frac{\sum_{k=1}^n (t_k - y_k)^2}{n} \quad (3.4)$$

di mana :

MSE = *Mean Squared Error*,
 n = jumlah semua data,
 t_k = target,
 y_k = keluaran dari JST.

2. Waktu

Waktu adalah lamanya proses "pembelajaran" jaringan RBF untuk mendapatkan nilai MSE yang minimum.

Untuk setiap nilai *spread* dan laju "pembelajaran", dibandingkan nilai MSE dan waktunya. Nilai *spread* dan laju "pembelajaran" yang memiliki MSE minimum akan dipilih dalam pengujian data.

3.6 Perbandingan Hasil "Pembelajaran"

1. Metode *clustering*

Setelah nilai delta dari kedua metode didapatkan, maka dilakukan perbandingan sebagai berikut.

1. Jika $\Delta_{K-Means} < \Delta_{FCM}$ maka metode *K-Means* memiliki performa yang lebih baik dibandingkan metode FCM karena tingkat kesalahan yang dihasilkan oleh metode *K-Means* relatif lebih kecil, begitu juga sebaliknya.
2. Jika waktu_{*K-Means*} < waktu_{FCM} maka metode *K-Means* memiliki performa yang lebih baik dibandingkan metode FCM karena waktu yang dibutuhkan oleh metode *K-Means* untuk mendapatkan delta minimum relatif lebih cepat, begitu juga sebaliknya.

2. Jaringan RBF

Setelah nilai MSE didapatkan, maka dilakukan perbandingan sebagai berikut.

1. Jika $MSE_{K-Means} < MSE_{FCM}$ maka metode *K-Means* memiliki performa yang lebih baik dibandingkan metode FCM karena tingkat kesalahan yang dihasilkan oleh metode *K-Means* relatif lebih kecil, begitu juga sebaliknya.
2. Jika waktu_{*K-Means*} < waktu_{FCM} maka metode *K-Means* memiliki performa yang lebih baik dibandingkan metode FCM karena waktu yang dibutuhkan oleh metode *K-Means* untuk mendapatkan delta minimum relatif lebih cepat, begitu juga sebaliknya.

3.7 Perbandingan Hasil Pengujian

Setelah jaringan RBF "belajar", maka langkah selanjutnya ialah menguji kemampuan jaringan dalam menghasilkan nilai keluaran jika dimasukkan pola masukan yang serupa. Nilai keluaran yang dihasilkan dibandingkan dengan nilai target yang sebenarnya. Kemudian dicari nilai MSE masing-masing metode untuk kemudian dibandingkan. Jika nilai $MSE_{K-Means} < MSE_{FCM}$ maka metode *K-Means* memiliki performa yang lebih baik untuk melakukan prediksi dibandingkan metode FCM karena tingkat kesalahan yang dihasilkan oleh metode *K-Means* relatif lebih kecil, begitu juga sebaliknya.

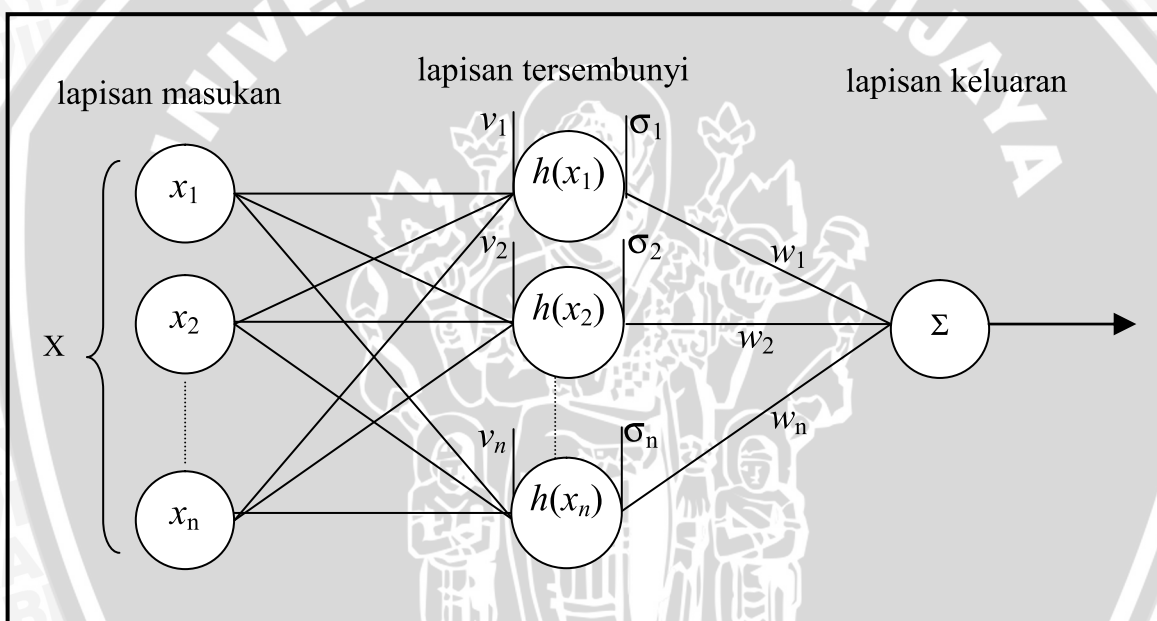


BAB IV PEMBAHASAN

Sebelum mengulas lebih jauh mengenai penerapan dari subbab 2.6.1 pada model "pembelajaran" jaringan RBF, perlu diketahui bahwa notasi $\mathbf{x}$ dan $f(\mathbf{x})$ dalam subbab 2.6.1 digantikan oleh bobot w dan E_s secara berturut-turut dalam model "pembelajaran" jaringan RBF.

4.1 Penerapan Metode *Gradient Descent* Pada Model "Pembelajaran" jaringan RBF

Untuk lebih memahami penerapan metode *Gradient Descent* pada "pembelajaran" jaringan RBF, perhatikan Gambar 4.1.



Gambar 4.1 Arsitektur JST dengan Model Pembelajaran Jaringan RBF Secara Umum

Gambar 4.1 merupakan JST yang memiliki arsitektur jaringan multi lapisan dengan model "pembelajaran" jaringan RBF, di mana untuk masing-masing lapisan memiliki jumlah unit yang berbeda-beda. Dalam hal ini setiap unit menyampaikan informasi kepada unit selanjutnya. Unit-unit lapisan masukan x_n mengirimkan informasi berupa pusat *cluster* v_n ke unit-unit lapisan tersembunyi. Di lapisan tersembunyi, pusat *cluster* v_n digunakan untuk menghitung fungsi basis radial $h(x_n)$ dengan terlebih dahulu menginisialisasi nilai *spread* σ_n . Dari lapisan tersembunyi akan dikirimkan informasi berupa bobot w_n ke lapisan keluaran. Di lapisan keluaran nilai w_n, v_n dan σ_n terus

diperbarui untuk kemudian dikirim kembali ke lapisan tersembunyi. Demikian seterusnya hingga jaringan mencapai kriteria pemberhentian yang diinginkan.

Pada dasarnya model "pembelajaran" jaringan RBF pada JST bertujuan untuk meminimumkan *error* yang terjadi dari selisih antara keluaran jaringan dengan target yang diberikan. *Error* tersebut berbentuk

$$E_s = \frac{1}{2} \sum_{k=1}^n (t_k - y_k)^2 \quad (4.1)$$

di mana : $E_s = \text{error}$,
 $t_k = \text{target}$,
 $y_k = \text{keluaran jaringan}$,
 $n = \text{banyak data}$.

Dari persamaan (4.1) diperoleh penurunan algoritma "pembelajaran" jaringan RBF sebagai berikut.

1. Perubahan bobot antara lapisan keluaran dengan lapisan tersembunyi.

a. Perubahan bobot w_{ik}

$$\begin{aligned} \frac{\partial E_s}{\partial w_{ik}} &= \frac{\partial E_s}{\partial y_k} \frac{\partial y_k}{\partial y_{in_k}} \frac{\partial y_{in_k}}{\partial w_{ik}} \\ &= \frac{\partial}{\partial y_k} \left(\frac{1}{2} \sum_{k=1}^n (t_k - y_k)^2 \right) \frac{\partial}{\partial y_{in_k}} (f(y_{in_k})) \frac{\partial}{\partial w_{ik}} \left(w_{k0} + \sum_{i=1}^C h_i w_{ik} \right) \\ &= - \sum_{k=1}^n (t_k - y_k) f'(y_{in_k}) h_i \end{aligned} \quad (4.2)$$

misal

$$\delta_k = \sum_{k=1}^n (t_k - y_k) f'(y_{in_k}) \quad (4.3)$$

dari persamaan (4.3) yang disubstitusikan pada persamaan (4.2) diperoleh

$$\frac{\partial E_s}{\partial w_{ik}} = -\delta_k h_i \quad (4.4)$$

Dengan meninjau kembali persamaan (2.47) di mana arah penurunan dari metode jaringan RBF adalah negatif dari *gradient* sehingga perubahan bobot w_{ik} berbentuk

$$w_{ik}(t+1) = w_{ik}(t) + \alpha \delta_k h_i . \quad (4.5)$$

b. Perubahan bobot bias w_{0k}

$$\begin{aligned} \frac{\partial E_s}{\partial w_{0k}} &= \frac{\partial E_s}{\partial y_k} \frac{\partial y_k}{\partial y_{in_k}} \frac{\partial y_k}{\partial w_{0k}} \\ &= \frac{\partial}{\partial y_k} \left(\frac{1}{2} \sum_{k=1}^n (t_k - y_k)^2 \right) \frac{\partial}{\partial y_{in_k}} (f(y_{in_k})) \frac{\partial}{\partial w_{0k}} \left(w_{0k} + \sum_{i=1}^C h_i w_{ik} \right) \\ &= - \sum_{k=1}^n (t_k - y_k) f'(y_{in_k}) \end{aligned} \quad (4.6)$$

dengan menggunakan persamaan (4.3) diperoleh perubahan bobot bias w_{0k} yang berbentuk

$$w_{ik}(t+1) = w_{ik}(t) + \alpha \delta_k . \quad (4.7)$$

c. Perubahan pusat *cluster* v_{ik}

$$\begin{aligned} \frac{\partial E_s}{\partial v_{ik}} &= \frac{\partial E_s}{\partial y_k} \frac{\partial y_k}{\partial h_i} \frac{\partial h_i}{\partial v_{ik}} \\ &= \frac{\partial}{\partial y_k} \left(\frac{1}{2} \sum_{k=1}^n (t_k - y_k)^2 \right) \frac{\partial}{\partial h_i} \left(w_{0k} + \sum_{i=1}^C h_i w_{ik} \right) \\ &\quad \frac{\partial}{\partial v_{ik}} \left(\exp[-(x_i - v_{ik})^2 / 2\sigma^2] \right) \\ &= - \sum_{k=1}^n (t_k - y_k) w_{ik} \frac{(x_i - v_{ik})}{\sigma^2} \exp[-(x_i - v_{ik})^2 / 2\sigma^2] \end{aligned} \quad (4.8)$$

misal

$$\beta_i = - \sum_{k=1}^n (t_k - y_k) \frac{(x_i - v_{ik})}{\sigma^2} \exp[-(x_i - v_{ik})^2 / 2\sigma^2] \quad (4.9)$$

dari persamaan (4.9) yang disubstitusikan pada persamaan (4.8) diperoleh

$$\frac{\partial E_s}{\partial v_{ik}} = -\beta_i w_{ik} . \quad (4.10)$$

Dengan menggunakan metode yang sama yakni arah penurunan yang bernilai negatif dari *gradient*, sehingga perubahan pusat *cluster* v_{ik} adalah

$$v_{ik}(t+1) = v_{ik}(t) + \alpha \beta_i w_{ik} . \quad (4.11)$$

d. Perubahan *spread* σ

$$\begin{aligned} \frac{\partial E_s}{\partial \sigma} &= \frac{\partial E_s}{\partial y_k} \frac{\partial y_k}{\partial h_i} \frac{\partial h_i}{\partial \sigma} \\ &= \frac{\partial}{\partial y_k} \left(\frac{1}{2} \sum_{k=1}^n (t_k - y_k)^2 \right) \frac{\partial}{\partial h_i} \left(w_{0k} + \sum_{i=1}^c h_i w_{ik} \right) \\ &\quad \frac{\partial}{\partial \sigma} \left(\exp[-(x_i - v_{ik})^2 / 2\sigma^2] \right) \\ &= -\sum_{k=1}^n (t_k - y_k) w_{ik} \frac{(x_i - v_{ik})^2}{\sigma^3} \exp[-(x_i - v_{ik})^2 / 2\sigma^2] \quad (4.12) \end{aligned}$$

misal

$$\gamma_i = -\sum_{k=1}^n (t_k - y_k) \frac{(x_i - v_{ik})^2}{\sigma^3} \exp[-(x_i - v_{ik})^2 / 2\sigma^2] \quad (4.13)$$

dari persamaan (4.13) yang disubstitusikan pada persamaan (4.12) diperoleh

$$\frac{\partial E_s}{\partial \sigma} = -\gamma_i w_{ik} . \quad (4.14)$$

Dengan menggunakan metode yang sama yakni arah penurunan yang bernilai negatif dari *gradient*, sehingga perubahan pusat *spread* σ adalah

$$\sigma(t+1) = \sigma(t) + \alpha \gamma_i w_{ik} . \quad (4.15)$$

Algoritma "pembelajaran" jaringan RBF dengan metode *Gradient Descent* yaitu

Input : Data "pembelajaran" (X), data target (T), jumlah *cluster* (l),
 w , laju "pembelajaran" (α).

Output : W_0 , MSE, Iterasi.

Algoritma

1. $[m,n] \leftarrow$ ukuran (X)
 $[l,m] \leftarrow$ ukuran matriks partisi (U)
 $[n,l] \leftarrow$ ukuran pusat *cluster* (V)
 $\text{delta} \leftarrow 1$
 $\text{itr} \leftarrow 1$
2. While ($\text{itr} < \text{MaxItr}$) do
 - a. For $i=1$ to l do
 - i. $T_k \leftarrow 0$
 - ii. For $k=1$ to m do
 $T_k \leftarrow T_k + U(i,k) \cdot w$
 - iii. For $j=1$ to n do
 1. $T_j \leftarrow 0$
 2. For $k=1$ to m do
 $T_j \leftarrow T_j + (U(i,k)^w) \cdot X(k,j)$
 3. $V(j,i) \leftarrow T_j / T_k$
 - b. For $i=1$ to m do
 - i. For $j=1$ to l do
 1. $\text{Signal} \leftarrow 0$
 2. For $k=1$ to n do
 $\text{Signal} \leftarrow \text{Signal} + (X(i,k) - V(k,j))^2$
 3. $D(j,i) \leftarrow \text{sqrt}(\text{Signal})$
 - c. $P(\text{itr}+1) \leftarrow 0$
 - d. For $k=1$ to m do
 - i. For $i=1$ to l do
 $P(\text{itr}+1) \leftarrow P(\text{itr}+1) + (D(i,k)^2) \cdot (U(i,k)^w)$
 - e. $U_{\text{lama}} \leftarrow U$
 - f. $U \leftarrow \text{zeros}(l,m)$
 - g. For $k=1$ to m do
 - i. For $i=1$ to l do
 1. If $D(i,k) \neq 0$ then
 - a. For $j=1$ to l do
 $U(i,k) \leftarrow U(i,k) + ((D(i,k)^2) / (D(j,k)^2))^{(1/(w-1))}$
 - b. $U(i,k) \leftarrow 1 / U(i,k)$

```

else
  c.  $U(i,k) \leftarrow 0$ 
  h.  $\text{delta} \leftarrow \max(\max(\text{abs}(U-U_{\text{lama}})))$ 
  i.  $\text{itr} \leftarrow \text{itr}+1$ 
3.  $\text{wb} \leftarrow \text{rand}(l,1)$ 
4.  $\text{bias} \leftarrow 0$ 
5.  $\text{MSE} \leftarrow 1$ 
6.  $\text{It} \leftarrow \text{itr}$ 
7.  $\text{spread} \leftarrow 0,2$ 
8.  $\text{alpha} \leftarrow 0,2$ 
9. While ( $\text{MSE} > \text{Eps}$ ) and ( $\text{it} \leq 15000$ ) do
  a.  $\text{It} \leftarrow \text{It}+1$ 
  b. For  $k=1$  to  $m$  do
    i. For  $i=1$  to  $l$  do
      1.  $d(k,i) \leftarrow 0$ 
      2. for  $j=1$  to  $n$  do
         $d(k,i) \leftarrow (X(k,j)-V(j,i))^2 + d(k,i)$ 
      3.  $h(k,i) \leftarrow \exp(-d(k,i)/(2*\text{spread}(i)*\text{spread}(i)))$ 
    ii.  $y(k) \leftarrow \text{bias}$ 
    iii. For  $i=1$  to  $l$ 
       $y(k) \leftarrow h(k,i)*\text{wb}(i)+y(k)$ 
    iv.  $\text{Beta\_temp} \leftarrow (X(k,j)-V(j,i))/(\text{spread}*\text{spread});$ 
    v.  $\text{Beta} \leftarrow (t(k)-y(k))*\text{Beta\_temp}*h(k,i);$ 
    vi.  $\text{Gamma\_temp} \leftarrow (X(k,j)-V(j,i))^2/(\text{spread}*\text{spread}*$ 
       $\text{spread});$ 
    vii.  $\text{Gamma} \leftarrow (t(k)-y(k))*\text{Gamma\_temp}*h(k,i);$ 
    viii. For  $i=1$  to  $l$  do
      1.  $V(1,i) \leftarrow V(1,i)+\text{alpha}*\text{Beta}*\text{wb}(i);$ 
      2.  $V(2,i) \leftarrow V(2,i)+\text{alpha}*\text{Beta}*\text{wb}(i);$ 
      3.  $V(3,i) \leftarrow V(3,i)+\text{alpha}*\text{Beta}*\text{wb}(i);$ 
      4.  $V(4,i) \leftarrow V(4,i)+\text{alpha}*\text{Beta}*\text{wb}(i);$ 
      5.  $V(5,i) \leftarrow V(5,i)+\text{alpha}*\text{Beta}*\text{wb}(i);$ 
      6.  $\text{spread} \leftarrow \text{spread}+\text{alpha}*\text{Gamma}*\text{wb}(i);$ 
      7.  $\text{wb}(i) \leftarrow \text{wb}(i)+\text{alpha}*(t(k)-y(k))*h(k,i);$ 
    ix.  $\text{bias} \leftarrow \text{bias}+\text{alpha}*(t(k)-y(k))$ 
  c. For  $i=1$  to  $m$  do
     $\text{Err} \leftarrow (y(i)-t(i))^2 + \text{delta}$ 
  d.  $\text{MSE}(\text{it}) \leftarrow \text{Err}/(m)$ 
10. Stop

```


4.2 Hasil "Pembelajaran" Pada Proses *Clustering*

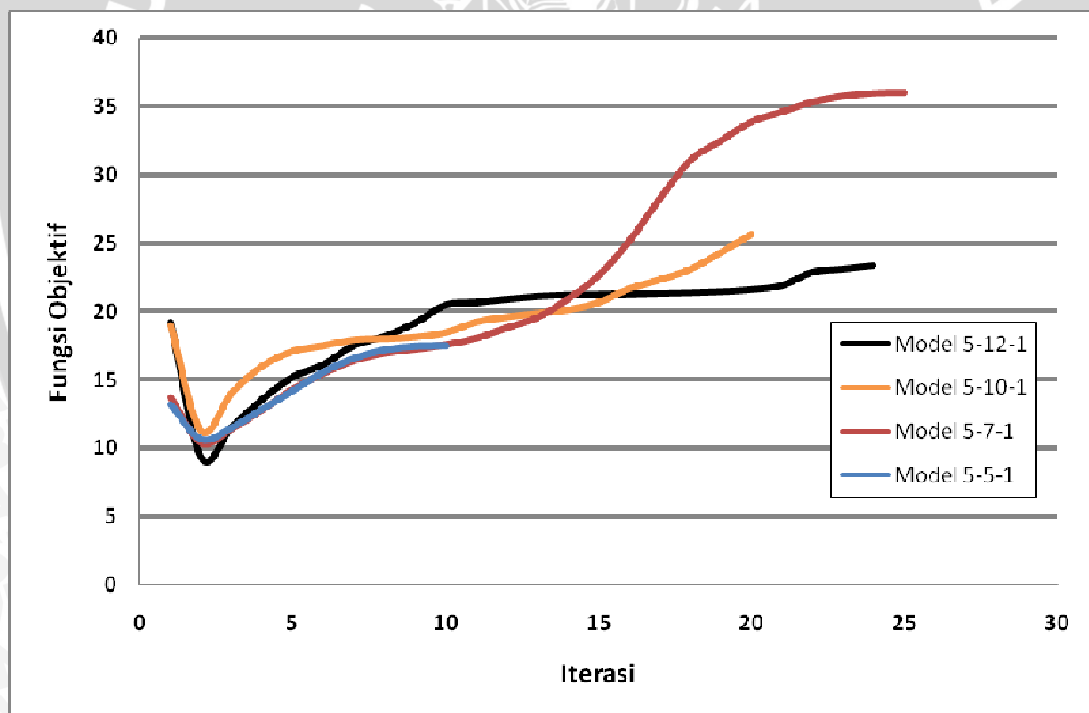
Hasil "pembelajaran" pada proses *clustering* memperlihatkan hasil yang berbeda, seperti yang dibahas pada subbab berikut ini.

4.2.1 Hasil *clustering* dengan metode *K-means*

Tabel 4.1 Hasil *clustering* dengan metode *K-means*

Jumlah <i>cluster</i>	Delta	Waktu
5	1,2272	1,9060
7	1,6006	3,7820
10	1,5252	3,3750
12	1,5655	4,0150

Dari Tabel 4.1 bisa dilihat bahwa nilai delta paling minimum dan waktu paling efisien untuk meminimumkan nilai delta diperoleh dari jumlah *cluster* sama dengan 5. Untuk lebih memperjelasnya bisa dilihat grafik hubungan antara fungsi objektif dengan jumlah iterasi seperti disajikan dalam Gambar 4.2.



Gambar 4.2 Grafik Hubungan Antara Fungsi Objektif Dengan Iterasi Pada Metode *K-Means*

Dari Gambar 4.2 bisa dilihat bahwa nilai fungsi objektif minimum dicapai pada saat jumlah *cluster* sama dengan 12. Akan tetapi, secara keseluruhan nilai minimum dari fungsi objektif diperoleh pada saat jumlah *cluster* sama dengan 5. Selain itu pada

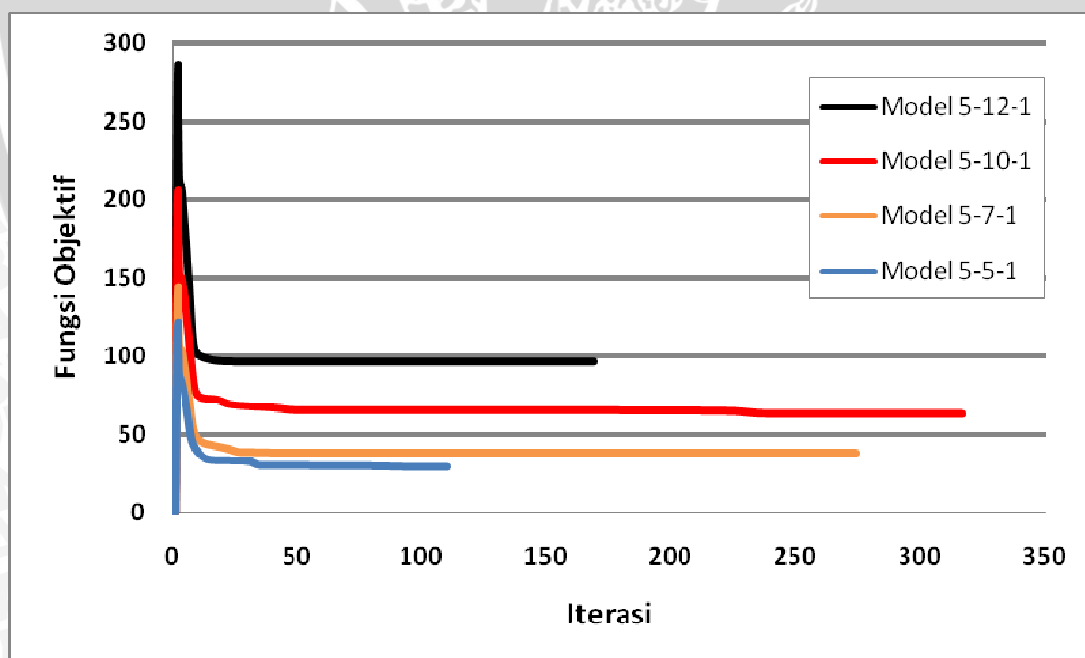
saat jumlah *cluster* sama dengan 5, jumlah iterasi yang dibutuhkan untuk meminimumkan fungsi objektif lebih sedikit dibandingkan dengan jumlah *cluster* yang lain. Sehingga bisa disimpulkan bahwa jumlah *cluster* sama dengan 5 lebih baik dalam melakukan proses *clustering* menggunakan metode *K-Means* dibandingkan jumlah *cluster* yang lain.

4.2.2 Hasil *clustering* dengan menggunakan metode FCM

Tabel 4.2 Hasil *clustering* dengan metode FCM

Jumlah <i>cluster</i>	Delta	Waktu
5	8,9775e-007	9,4370
7	9,3187e-007	42,5310
10	9,8260e-007	63,3120
12	9,9662e-007	51,6880

Sama halnya dengan metode *K-Means*, pada metode FCM, nilai delta paling minimum dan waktu paling efisien untuk meminimumkan nilai delta diperoleh dari jumlah *cluster* sama dengan 5 seperti yang terlihat pada Tabel 4.2. Untuk lebih memperjelasnya bisa dilihat grafik hubungan antara fungsi objektif dengan jumlah iterasi seperti disajikan dalam Gambar 4.3.



Gambar 4.3 Grafik Hubungan Antara Fungsi Objektif Dengan Iterasi Pada Metode FCM

Dari Gambar 4.3 bisa dilihat bahwa nilai fungsi objektif minimum dicapai pada jumlah *cluster* sama dengan 5 dengan jumlah

iterasi yang sedikit, sehingga bisa disimpulkan bahwa jumlah *cluster* sama dengan 5 lebih baik dalam melakukan proses *clustering* menggunakan metode FCM dibandingkan jumlah *cluster* yang lain.

4.2.3 Pengukuran kinerja metode *clustering*

Tabel 4.3 Perbandingan hasil *clustering*

Metode	Delta Rata-Rata	Waktu Rata-Rata
<i>K-Means</i>	1,48	3,27
FCM	9,52e-07	4,17e+01
Hasil	$\Delta_{\text{FCM}} < \Delta_{\text{K-Means}}$	$\text{waktu}_{\text{K-Means}} < \text{waktu}_{\text{FCM}}$

Dari Tabel 4.3 bisa dilihat bahwa nilai delta pada metode FCM lebih kecil daripada nilai delta pada metode *K-Means*. Hal ini menunjukkan bahwa metode FCM lebih baik dibandingkan dengan metode *K-Means* dalam hal meminimumkan nilai delta. Akan tetapi, dari segi waktu yang dibutuhkan untuk meminimumkan nilai delta, waktu yang dibutuhkan metode *K-Means* lebih sedikit dibandingkan dengan metode FCM. Hal ini menunjukkan bahwa metode *K-Means* memiliki performansi yang lebih baik dibandingkan dengan metode FCM.

Dari Gambar 4.2 dan 4.3 bisa dilihat bahwa proses *clustering* menggunakan metode FCM lebih baik dibandingkan dengan metode *K-Means*. Pada metode FCM, untuk setiap jumlah *cluster* yang diberikan, grafik fungsi objektif terhadap iterasi bisa mencapai konvergen. Hal ini berbeda dengan metode *K-Means*, di mana grafik fungsi objektif terhadap iterasinya tidak ada yang mencapai konvergen walaupun dari segi waktu melakukan proses *clustering* lebih cepat dibandingkan dengan metode FCM.

4.3 Hasil "Pembelajaran" Pada Jaringan RBF

Sebelum melakukan "pembelajaran" dalam jaringan RBF maka perlu ditentukan terlebih dahulu jumlah unit lapisan tersembunyi yang digunakan pada jaringan RBF berdasarkan jumlah *cluster* yang diperoleh dari proses *clustering*. Dari proses *clustering*, maka jumlah unit lapisan tersembunyi yang digunakan adalah sama dengan 5 unit.

4.3.1 Hasil "pembelajaran" jaringan RBF dengan metode *K-Means* dan metode *Gradient Descent*

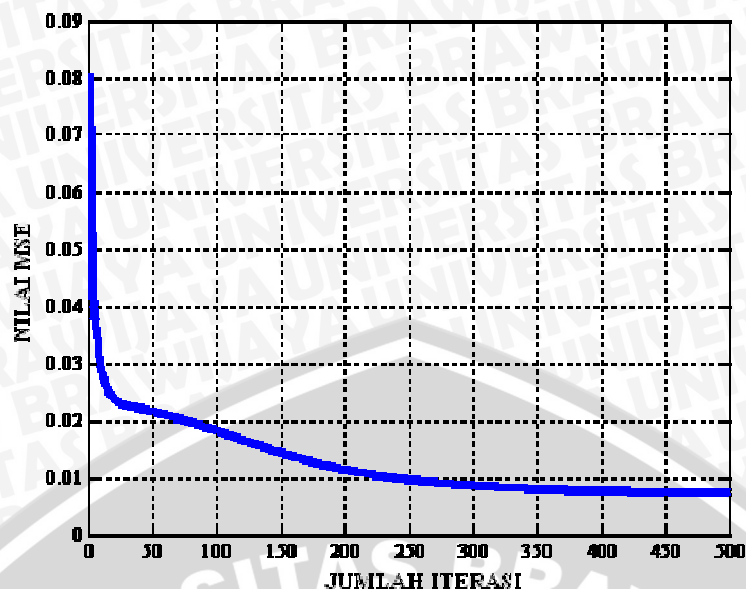
Tabel 4.4 Hasil "pembelajaran" dengan metode *K-means*

Model	Alpha	Spread	MSE	Waktu(Detik)
5-5-1	0,2	1	0,0239	75,0470
		2	0,0076	74,6880
		3	0,0099	76,9060
		4	0,0099	76,2970
		5	0,0107	75,3910
	0,35	1	0,0381	74,5940
		2	0,0216	76,5000
		3	0,0285	77,6400
		4	0,0271	76,9840
		5	0,0263	76,3910
	0,5	1	0,0621	75,0780
		2	0,0531	74,1250
		3	0,0558	74,4690
		4	0,0629	77,1570
		5	0,0599	71,8750
	0,8	1	0,0492	75,3750
		2	0,0492	74,0160
		3	0,0512	75,6720
		4	0,0494	76,2090
		5	0,0515	79,9530

Berdasarkan Tabel 4.4 dapat diperoleh nilai MSE terkecil dari masing-masing laju "pembelajaran" pada saat MSE mencapai minimum yaitu:

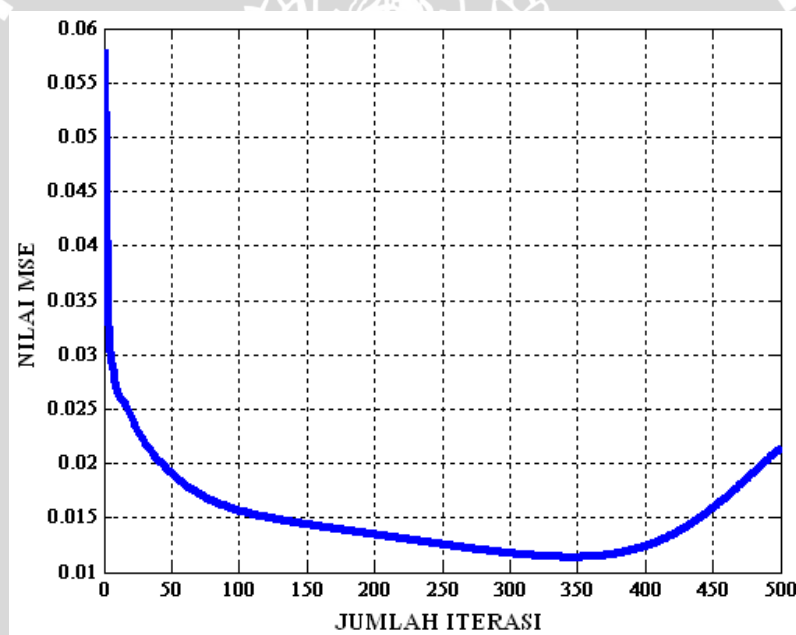
1. Laju "pembelajaran" 0,2.

Dari Gambar 4.4 terlihat perolehan nilai MSE terkecil untuk model arsitektur 5-5-1 dengan laju "pembelajaran" 0,2 dan *spread* 2 yaitu 0,0076. Dari gambar tersebut terlihat bahwa grafik MSE bergerak secara stabil mulai iterasi ke-200. Total waktu yang diperlukan JST untuk melakukan "pembelajaran" ialah selama 74,6880 detik.



Gambar 4.4 Hasil "Pembelajaran" Dengan $\alpha = 0,2$ Dan $Spread = 2$

2. Laju "pembelajaran" 0,35

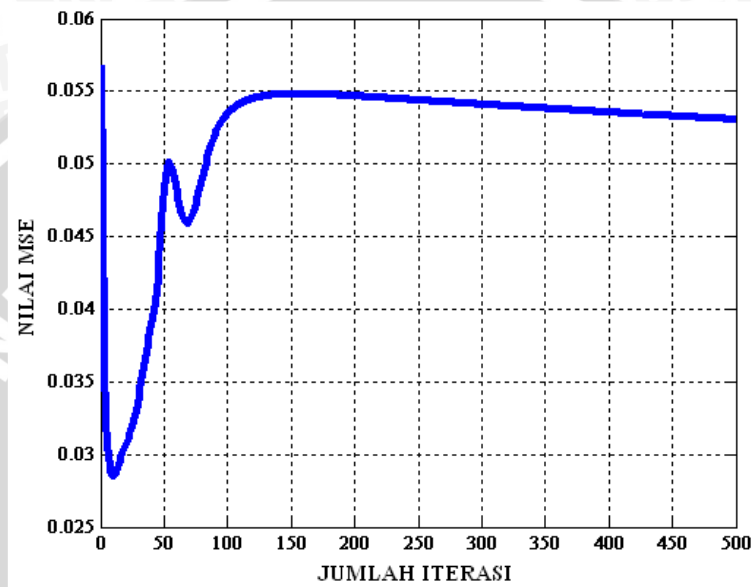


Gambar 4.5 Hasil "Pembelajaran" Dengan $\alpha = 0,2$ Dan $Spread = 2$

Dari Gambar 4.5 terlihat perolehan MSE terkecil untuk model arsitektur 5-5-1 dengan laju "pembelajaran" 0,2 dan *spread* 2 adalah 0,0216. Pada Gambar 4.5 terlihat bahwa jaringan tidak dapat mencapai minimum global. Total waktu yang diperlukan jaringan untuk berhenti melakukan "pembelajaran" adalah selama 76,5 detik.

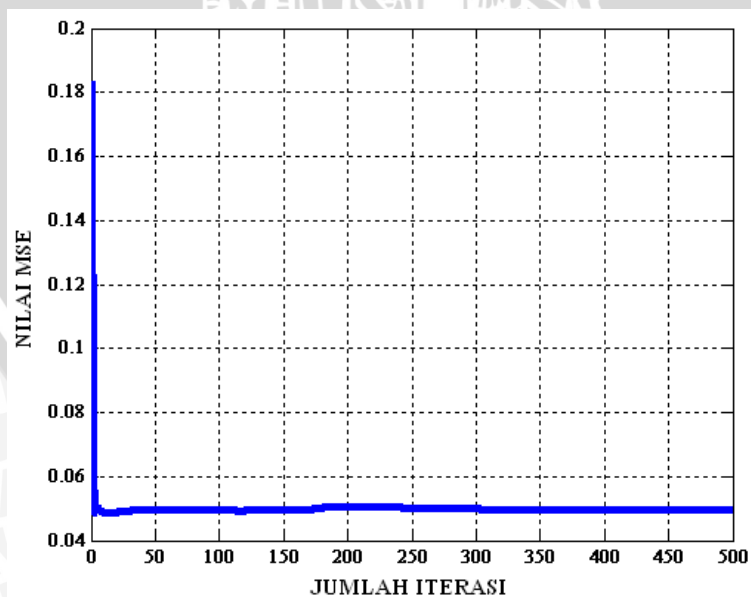
3. Laju "pembelajaran" 0,5

Dari Gambar 4.6 terlihat perolehan MSE terkecil untuk model arsitektur 5-5-1 dengan laju "pembelajaran" 0,5 dan *spread* 2 yaitu 0,0531. Sama halnya seperti Gambar 4.5, pada Gambar 4.6 terlihat bahwa jaringan tidak dapat mencapai minimum global. Jaringan pada awalnya bergerak tidak stabil dan mulai bergerak stabil pada saat iterasi ke-100. Total waktu yang diperlukan jaringan untuk melakukan "pembelajaran" adalah selama 74,1250 detik.



Gambar 4.6 Hasil "Pembelajaran" Dengan $\alpha = 0,5$ Dan *Spread* = 2

4. Laju "pembelajaran" 0,8



Gambar 4.7 Hasil "Pembelajaran" Dengan $\alpha = 0,8$ Dan *Spread* = 2

Dari Gambar 4.7 terlihat perolehan MSE terkecil untuk model arsitektur 5-5-1 dengan laju "pembelajaran" sebesar 0,2 dan *spread* 2 yaitu 0,0492. Dari Gambar 4.7 terlihat bahwa dari awal iterasi jaringan bergerak stabil dengan total waktu yang diperlukan jaringan untuk melakukan "pembelajaran" adalah selama 74,016 detik.

4.3.2 Hasil "pembelajaran" jaringan RBF dengan metode FCM dan metode *Gradient Descent*

Tabel 4.5 Hasil "pembelajaran" dengan metode FCM

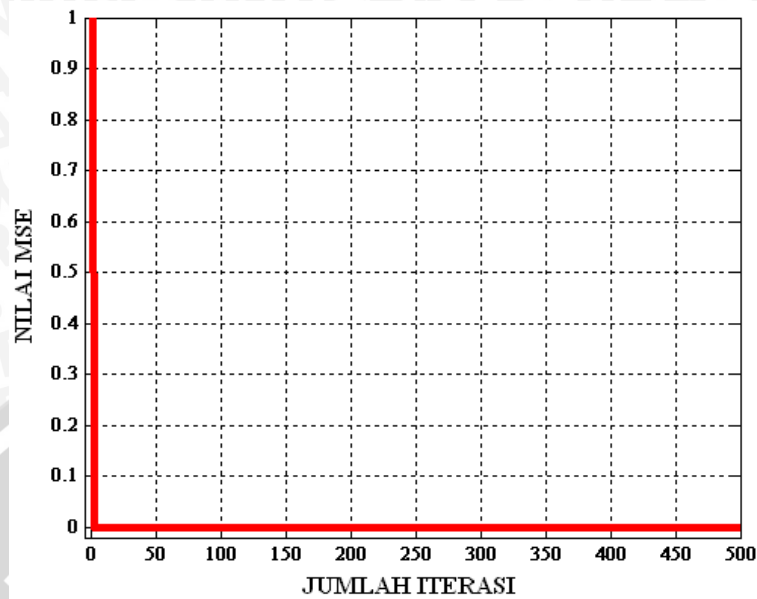
Model	Alpha	Spread	MSE	Waktu(Detik)
5-5-1	0,2	1	2,9474e-006	59,8130
		2	1,4489e-006	59,6100
		3	2,3160e-006	57,9070
		4	6,4677e-006	56,8590
		5	1,1051e-005	57,6720
	0,35	1	2,4913e-005	58,9220
		2	2,1990e-007	60,2970
		3	5,1712e-007	56,5780
		4	1,7872e-006	56,6090
		5	5,3936e-006	56,8280
	0,5	1	3,1687e-005	59,1410
		2	3,1686e-005	59,3900
		3	3,1685e-005	70,5620
		4	5,5882e-005	68,9380
		5	3,1685e-005	56,7500
	0,8	1	2,6250e-005	58,4840
		2	7,3233e-006	60,2180
		3	3,5225e-005	68,8900
		4	2,6688e-005	59,1880
		5	2,6250e-005	65,0630

Berdasarkan Tabel 4.5 dapat diperoleh nilai MSE terkecil dari masing-masing laju "pembelajaran" pada saat MSE mencapai minimum yaitu:

1. Laju "pembelajaran" 0,2

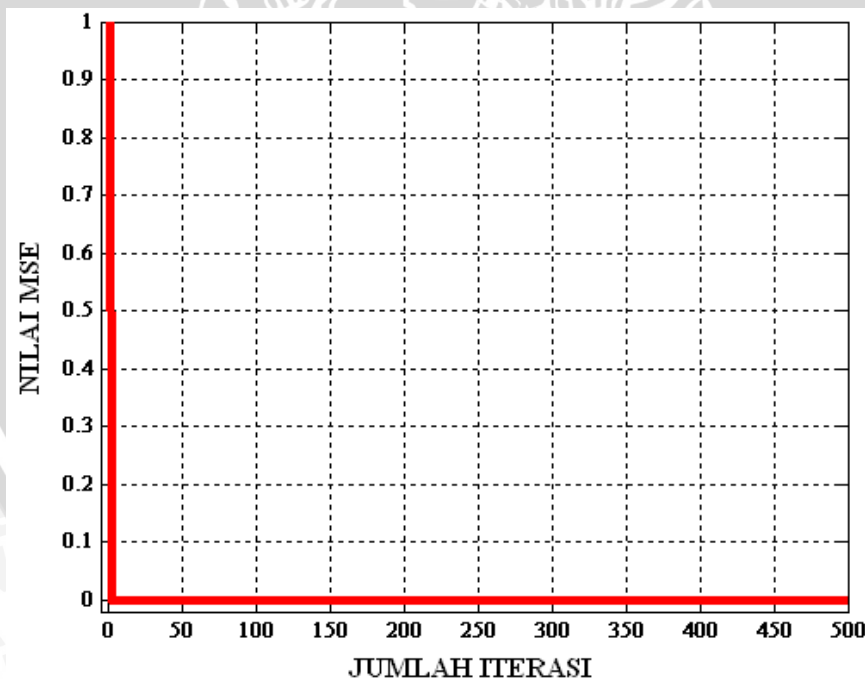
Dari Gambar 4.8 terlihat perolehan nilai MSE terkecil untuk model arsitektur 5-5-1 dengan laju "pembelajaran" 0,2 dan *spread* 2 yaitu 1,4489e-006. Dari gambar tersebut terlihat bahwa grafik MSE

bergerak secara stabil dengan total waktu yang diperlukan untuk melakukan "pembelajaran" selama 59,61 detik.



Gambar 4.8 Hasil "Pembelajaran" Dengan $\alpha = 0,2$ Dan $Spread = 2$

2. Laju "pembelajaran" 0,35

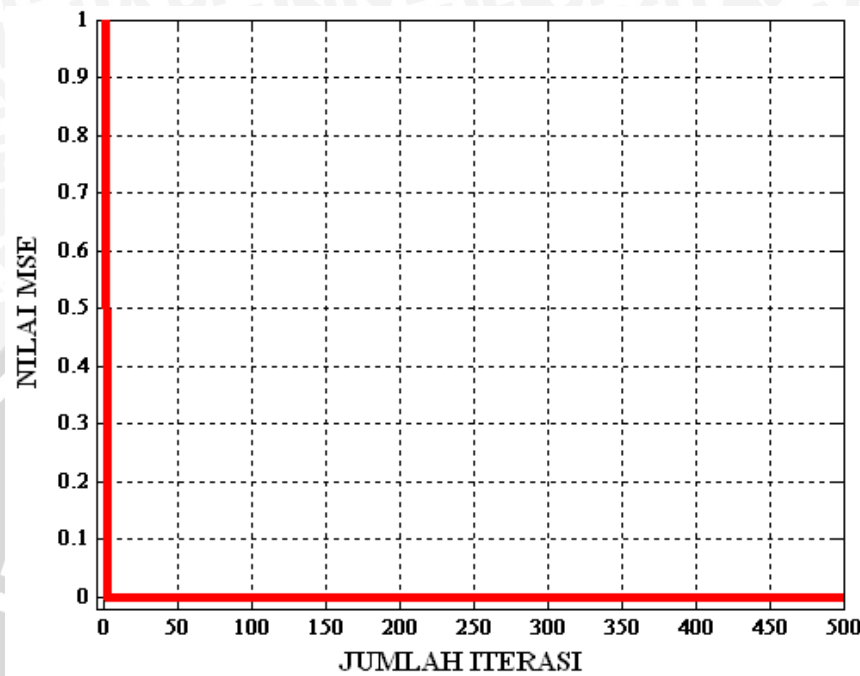


Gambar 4.9 Hasil "Pembelajaran" Dengan $\alpha = 0,35$ Dan $Spread = 2$

Dari Gambar 4.9 terlihat perolehan MSE terkecil untuk model arsitektur 5-5-1 dengan laju "pembelajaran" 0,35 dan $spread$ 2 adalah $2,1990e-007$. Pada model ini terlihat bahwa mulai dari iterasi

awal hingga akhir grafik MSE stabil dan jaringan memerlukan waktu untuk berhenti melakukan "pembelajaran" selama 60,297 detik.

3. Laju "pembelajaran" 0,5

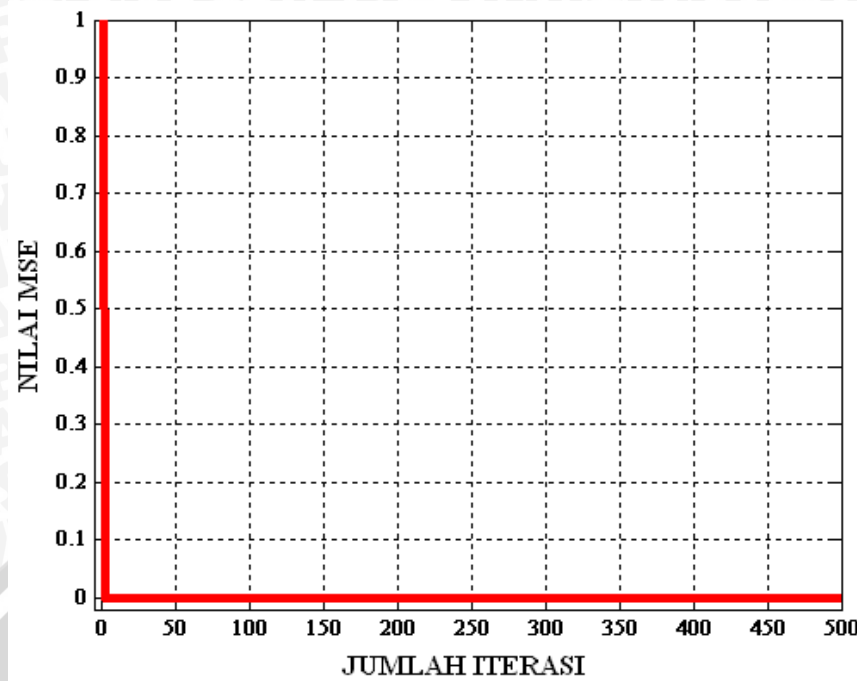


Gambar 4.10 Hasil "Pembelajaran" Dengan $\alpha = 0,5$ Dan $Spread = 2$

Dari Gambar 4.10 terlihat perolehan MSE terkecil untuk model arsitektur 5-5-1 dengan laju "pembelajaran" 0,5 dan *spread* 2 yaitu 3,1686e-005. Total waktu yang diperlukan jaringan untuk melakukan "pembelajaran" ialah selama 59,39 detik.

4. Laju "pembelajaran" 0,8

Dari Gambar 4.11 terlihat perolehan MSE terkecil untuk model arsitektur 5-5-1 dengan laju "pembelajaran" sebesar 0,8 dan *spread* 2 yaitu 7,3233e-006. Dari gambar tersebut terlihat bahwa grafik MSE bergerak secara stabil dan memerlukan waktu selama 60,218 untuk berhenti melakukan proses "pembelajaran".



Gambar 4.11 Hasil "Pembelajaran" Dengan $\alpha = 0,8$ Dan $Spread = 2$

4.3.3 Pengukuran kinerja jaringan RBF

Tabel 4.6 Perbandingan hasil jaringan RBF

Metode	MSE Rata-Rata	Waktu Rata-Rata
<i>K-Means</i>	0,037395	75,7184
FCM	1,81e-05	60,38595
Hasil	$MSE_{FCM} < MSE_{K-Means}$	$waktu_{FCM} < waktu_{K-Means}$

Berdasarkan Tabel 4.6 diperoleh suatu perbandingan dari kedua metode, antara lain.

1. Nilai MSE

Dari seluruh nilai laju "pembelajaran" dan *spread* yang diberikan menghasilkan nilai MSE yang berbeda-beda. Selanjutnya jika ditelaah lebih jauh nilai MSE terkecil terdapat pada metode FCM jika dibandingkan dengan metode *K-Means*. Namun dari kedua metode tersebut dapat dilihat bahwa pada metode FCM kegagalan untuk konvergen lebih kecil jika dibandingkan dengan metode *K-Means*.

2. Efisiensi waktu

Dari hasil "pembelajaran" dengan metode FCM dapat dilihat waktu yang dibutuhkan jaringan untuk meminimumkan nilai MSE relatif lebih cepat jika dibandingkan dengan metode *K-Means*.

4.4 Pemilihan Model Arsitektur Untuk Pengujian Jaringan RBF

Pemilihan model arsitektur untuk pengujian jaringan RBF dilakukan untuk menentukan model arsitektur yang paling sesuai untuk mengenali pola data "pembelajaran" dari data saham harian PT.Astra, Tbk. Dalam hal ini digunakan beberapa kriteria antara lain:

1. Jumlah Cluster

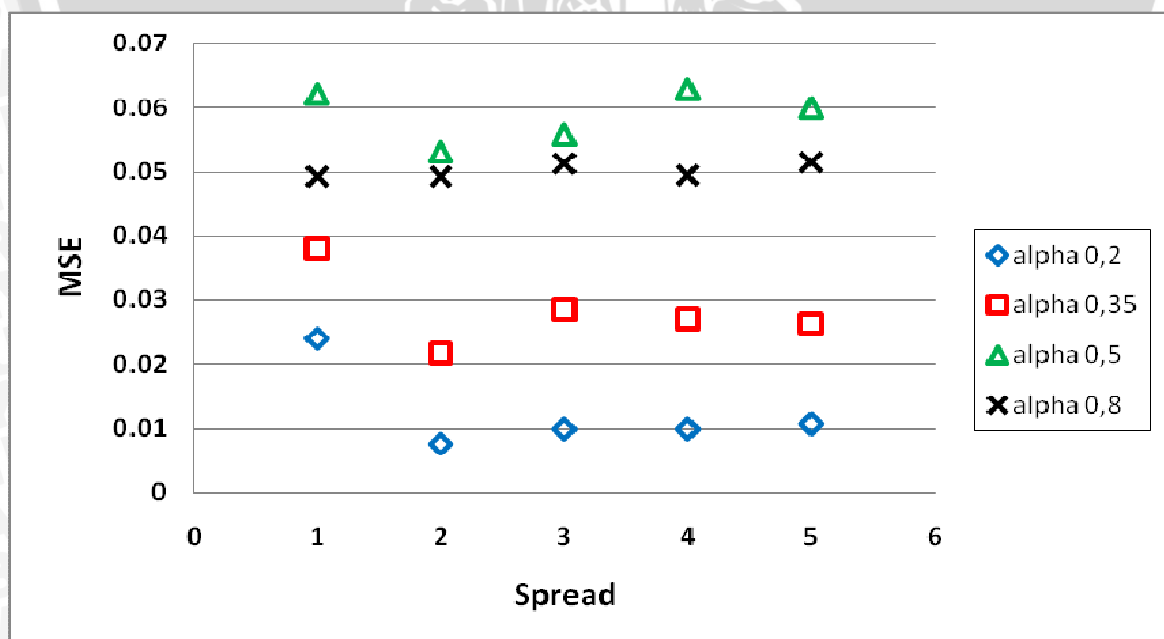
Berdasarkan pembahasan pada subbab 4.2.1 dan subbab 4.2.2 maka jumlah *cluster* yang dipilih untuk melakukan pengujian data ialah sama dengan 5 baik untuk metode *K-Means* maupun metode FCM.

2. Spread

Karena hasil "pembelajaran" pada subbab 4.2.1 dan subbab 4.2.2 menunjukkan jumlah *cluster* sama dengan 5 adalah jumlah yang baik untuk melakukan "pembelajaran", maka untuk melihat perolehan nilai MSE berdasarkan nilai *spread* yang berbeda-beda digunakan model arsitektur 5-5-1 untuk kedua metode.

a). Metode *K-Means*

Berdasarkan Tabel 4.4 dapat dilihat perolehan MSE berdasarkan *spread* untuk nilai laju "pembelajaran" yang berbeda-beda pada model arsitektur 5-5-1 seperti yang ditunjukkan oleh Gambar 4.12.

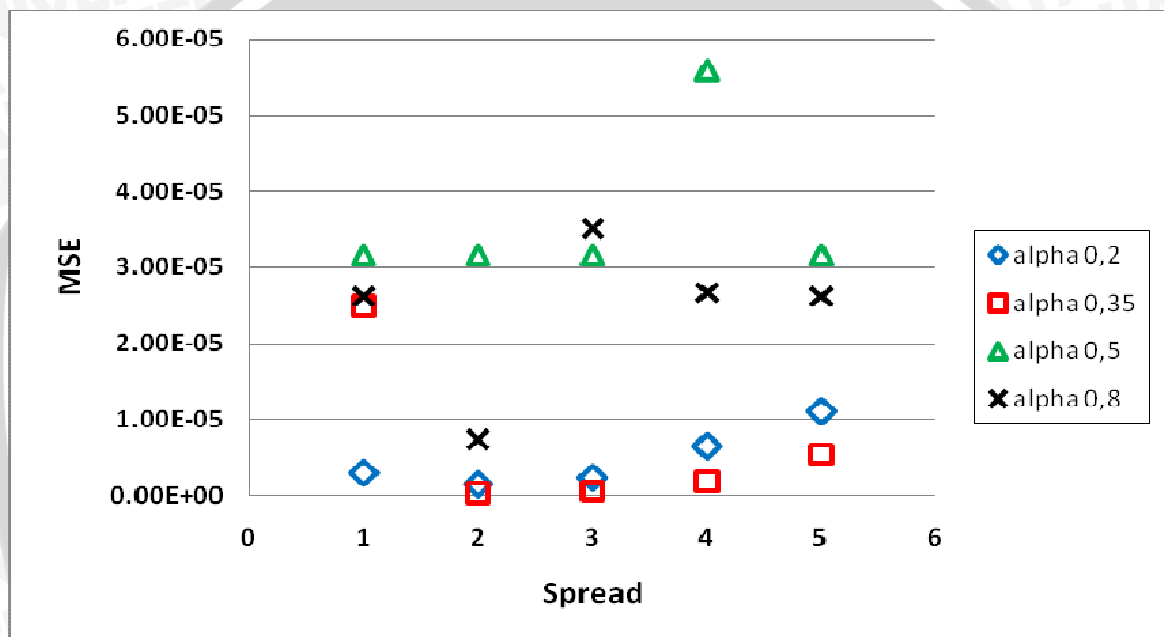


Gambar 4.12 Hasil "Pembelajaran" Dari Metode *K-Means* Dengan *Spread* yang Berbeda

Gambar 4.12 menunjukkan bahwa untuk setiap laju "pembelajaran" yang diberikan, nilai MSE minimum diperoleh dari *spread* sama dengan 2. Hal ini menunjukkan bahwa *spread* sama dengan 0,2 merupakan *spread* yang baik untuk meminimumkan nilai MSE pada metode *K-Means*.

b). Metode FCM

Berdasarkan Tabel 4.5 dapat dilihat perolehan MSE berdasarkan *spread* untuk nilai laju "pembelajaran" yang berbeda-beda pada model arsitektur 5-5-1 seperti yang ditunjukkan oleh Gambar 4.13.



Gambar 4.13 Hasil "Pembelajaran" Dari Metode FCM Dengan *Spread* yang Berbeda

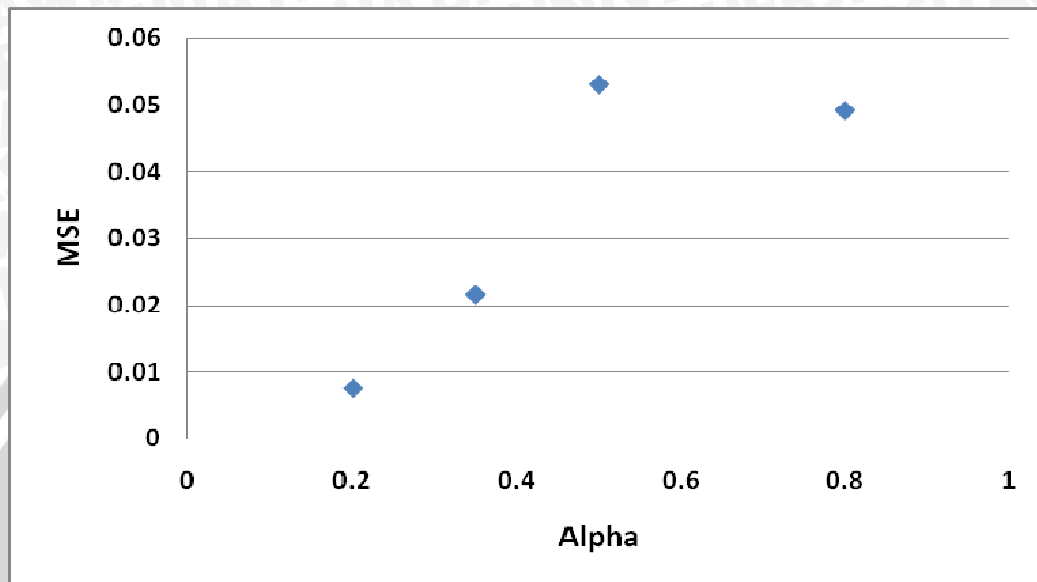
Sama halnya dengan metode *K-Means*, dari Gambar 4.13 ditunjukkan bahwa nilai MSE minimum dicapai jaringan pada saat *spread* sama dengan 2 untuk semua nilai laju "pembelajaran". *Spread* sama dengan 2 merupakan *spread* yang baik digunakan untuk meminimumkan nilai MSE pada metode FCM.

3. Laju "pembelajaran"

Dari jumlah *cluster* dan nilai *spread* yang diperoleh dari point 1 dan 2 pada subbab 4.4, maka untuk mengetahui hubungan nilai laju "pembelajaran" dengan nilai MSE digunakan model arsitektur 5-5-1 dengan *spread* sama dengan 2 untuk masing-masing metode.

a). Metode *K-Means*

Berdasarkan Tabel 4.4 dapat dilihat perolehan MSE berdasarkan nilai laju "pembelajaran" yang berbeda-beda untuk model arsitektur 5-5-1 dan *spread* sama dengan 2 seperti yang ditunjukkan oleh Gambar 4.14.



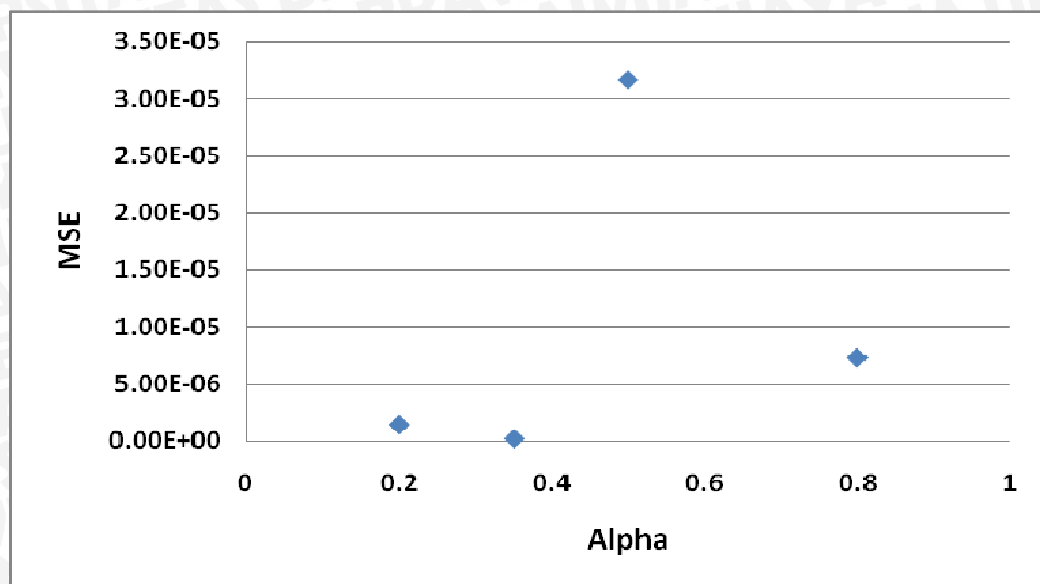
Gambar 4.14 Hasil "Pembelajaran" Dari Metode *K-Means* Dengan Laju "Pembelajaran" yang Berbeda

Dari Gambar 4.14 dapat dilihat bahwa nilai MSE tertinggi diperoleh pada saat laju "pembelajaran" sama dengan 0,5 dan nilai MSE terendah diperoleh pada saat laju "pembelajaran" sama dengan 0,2. Hal ini menunjukkan bahwa laju "pembelajaran" sama dengan 0,2 merupakan laju "pembelajaran" yang baik untuk meminimumkan nilai MSE pada metode *K-Means*.

2). Metode FCM

Berdasarkan Tabel 4.5 dapat dilihat perolehan MSE berdasarkan nilai laju "pembelajaran" yang berbeda-beda untuk model arsitektur 5-5-1 dan *spread* sama dengan 2 seperti yang ditunjukkan oleh Gambar 4.15.

Dari Gambar 4.15 dapat dilihat bahwa nilai MSE tertinggi diperoleh pada saat laju "pembelajaran" sama dengan 0,5 dan nilai MSE terendah diperoleh pada saat laju "pembelajaran" sama dengan 0,35. Laju "pembelajaran" sama dengan 0,35 merupakan laju "pembelajaran" yang baik digunakan untuk meminimumkan nilai MSE pada metode FCM.



Gambar 4.15 Hasil "Pembelajaran" Dari Metode FCM Dengan Laju "Pembelajaran" yang Berbeda

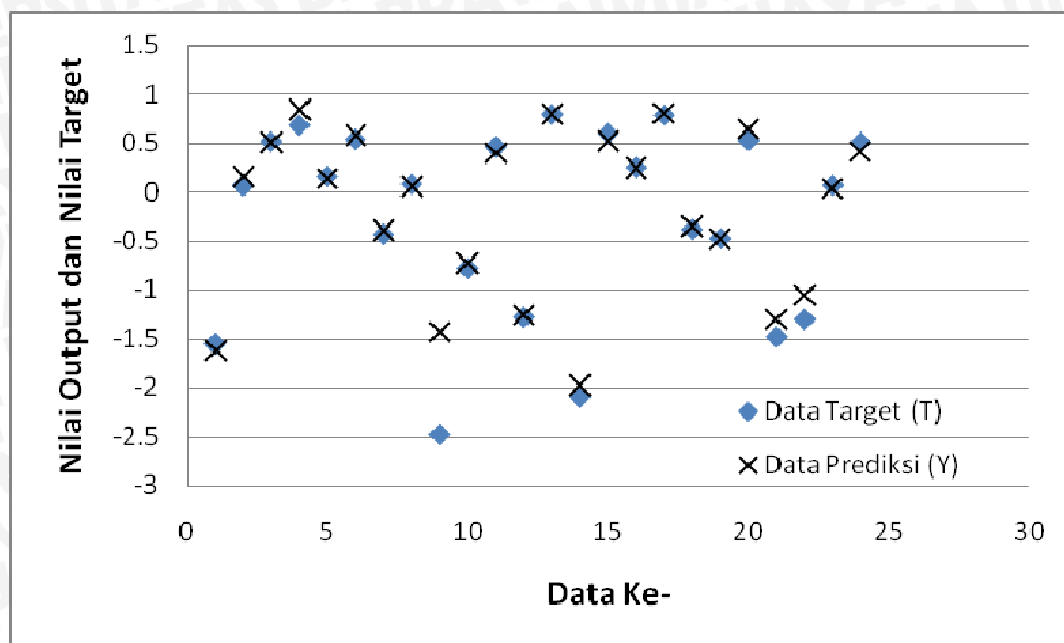
4.5 Perbandingan Hasil Pengujian

1. Metode *K-Means*

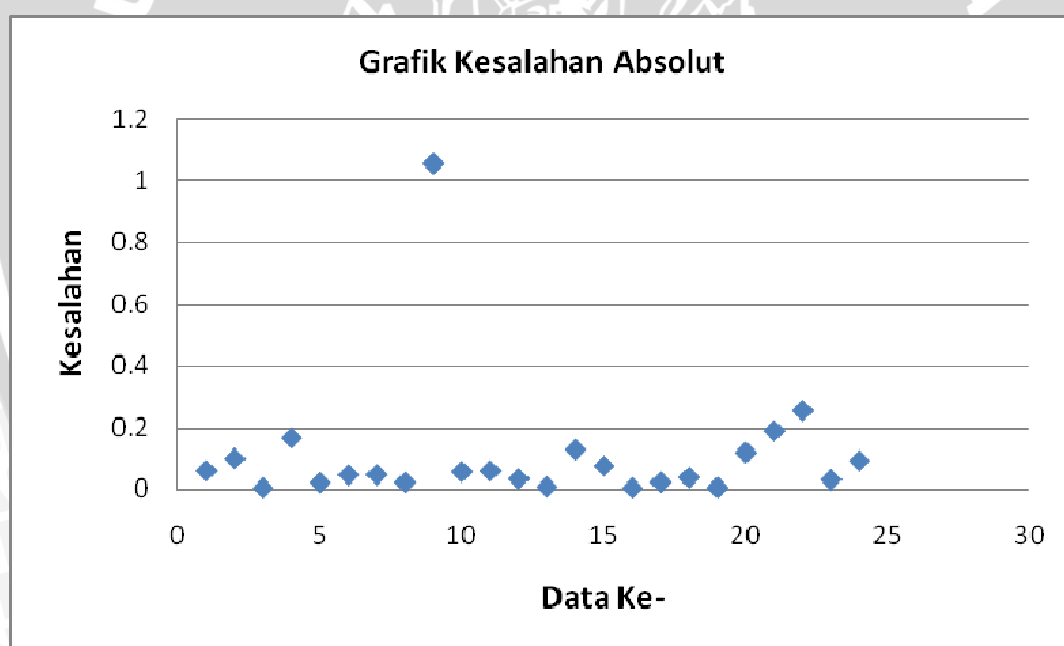
Dengan menggunakan nilai awal (Lampiran 1) yang diperoleh dari proses "pembelajaran" dengan metode *K-Means* diperoleh hasil simulasi yang disajikan dalam Tabel 4.7. Berdasarkan Tabel 4.7 dapat dilihat selisih antara nilai target dan keluaran dari jaringan terbesar adalah 0,0744 dan menghasilkan MSE 2,7300e-02. Selanjutnya dari Tabel 4.7 dapat digambarkan nilai dari target dan keluaran jaringan yang ditunjukkan oleh Gambar 4.16 dan kesalahan yang terjadi antara nilai target dan keluaran dari jaringan yang ditunjukkan oleh Gambar 4.17.

Tabel 4.7 Hasil simulasi menggunakan metode *K-Means*

Data ke-	Harga Penutupan (T)	Harga Penutupan dari JST (Y)	Selisih
1.	-1,8532	-1,6138	0,0586
2.	0,4702	0,16	-0,0983
3.	1,4659	0,5102	0,0032
4.	0,948	0,8447	-0,1647
5.	1,0133	0,138	0,0201
6.	-0,7367	0,5803	-0,0451
7.	-0,1634	-0,3892	-0,0457
8.	-0,6462	0,057	0,0211
9.	0,2615	-1,4267	-1,0544
10.	0,8625	-0,7275	-0,0555
11.	0,3797	0,4003	0,0579
12.	0,2288	-1,25	-0,0321
13.	-2,2153	0,7981	-0,0067
14.	-1,7325	-1,9728	-0,1273
15.	-1,0385	0,5224	0,0744
16.	1,3754	0,2475	0,003
17.	0,8323	0,8059	-0,0216
18.	0,8323	-0,3473	-0,0366
19.	0,2892	-0,4728	-0,0041
20.	0,0946	0,6439	-0,1176
21.	0,0176	-1,2966	-0,1878
22.	0,1383	-1,0497	-0,2535
23.	0,44	0,0371	0,0295
24.	0,7418	0,4151	0,0911
	MSE		2,7300e-02



Gambar 4.16 Perbandingan Target Dengan Keluaran Jaringan Pada Metode *K-means*



Gambar 4.17 Selisih Antara Target Dengan Keluaran Jaringan

Berdasarkan Gambar 4.16 terlihat bahwa hasil prediksi yang dilakukan oleh JST masih belum sempurna, di mana data prediksi ada yang mendekati data target, ada juga yang menjauhi data target. Selanjutnya kesalahan terbesar mencapai 1,0544 yang ditunjukkan oleh Gambar 4.17.

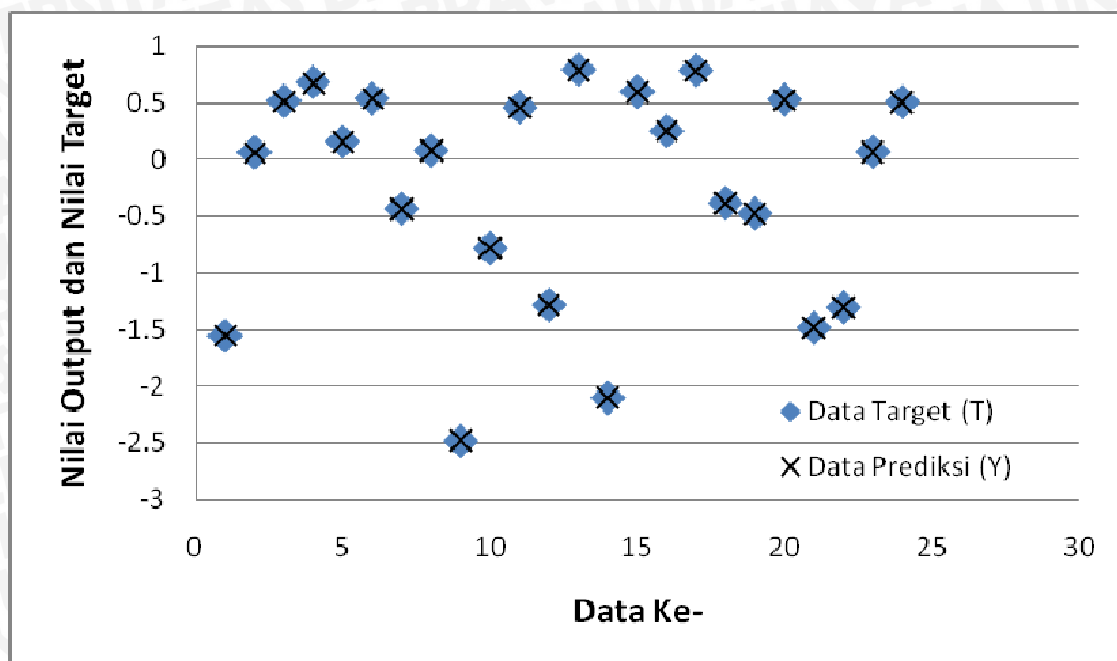
2. Metode FCM

Dengan menggunakan model arsitektur 5-5-1 dan nilai awal (Lampiran 1) yang diperoleh dari proses "pembelajaran" metode FCM. Selanjutnya dilakukan simulasi data yang baru dan diperoleh hasil yang disajikan dalam Tabel 4.8.

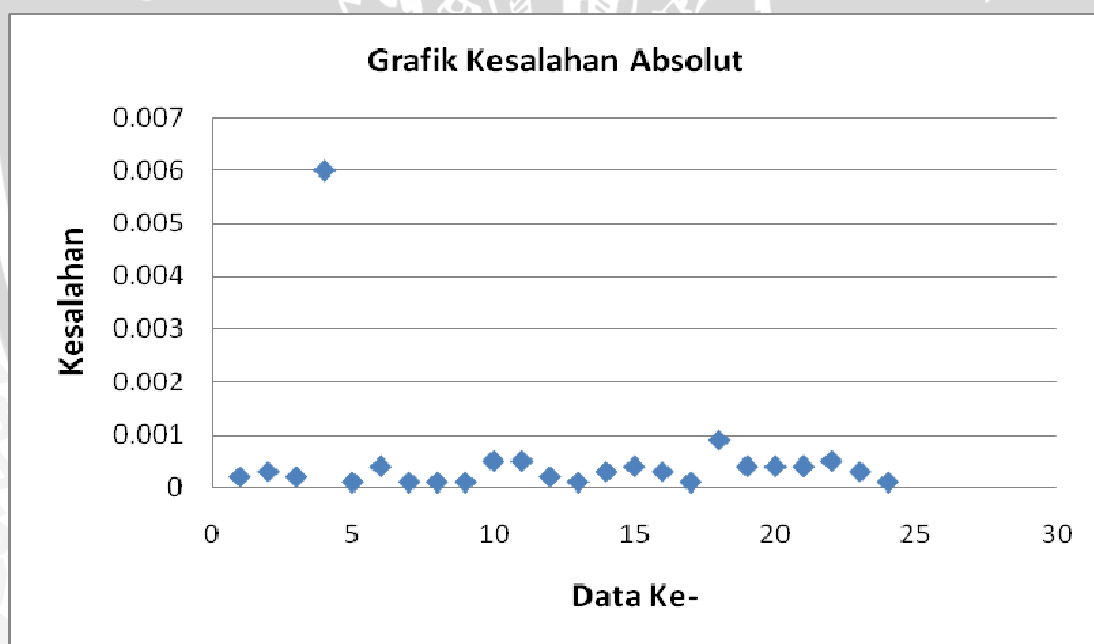
Tabel 4.8 Hasil simulasi menggunakan metode FCM

Data ke-	Harga Penutupan (T)	Harga Penutupan dari JST (Y)	Selisih
1.	-1,5552	-1,555	-0,0002
2.	0,0617	0,0614	0,0003
3.	0,5134	0,5132	0,0002
4.	0,6800	0,674	0,006
5.	0,1581	0,1582	-0,0001
6.	0,5352	0,5348	0,0004
7.	-0,4349	-0,4348	-1e-04
8.	0,0781	0,078	0,0001
9.	-2,4811	-2,481	-0,0001
10.	-0,7830	-0,7825	-0,0005
11.	0,4582	0,4587	-0,0005
12.	-1,2821	-1,2823	0,0002
13.	0,7914	0,7915	-1e-04
14.	-2,1001	-2,1004	0,0003
15.	0,5968	0,5964	0,0004
16.	0,2505	0,2502	0,0003
17.	0,7843	0,7844	-1e-04
18.	-0,3839	-0,383	-0,0009
19.	-0,4769	-0,4765	-0,0004
20.	0,5263	0,5267	-0,0004
21.	-1,4844	-1,4848	0,0004
22.	-1,3032	-1,3037	0,0005
23.	0,0666	0,0663	0,0003
24.	0,5062	0,5063	-1e-04
MSE			8,1063e-007

Dari Tabel 4.8 dapat dilihat selisih antara nilai target dan keluaran dari jaringan terbesar adalah 0,006 dan menghasilkan MSE 8,1063e-007. Selanjutnya dari Tabel 4.8 dapat digambarkan nilai dari target dan keluaran jaringan yang ditunjukkan oleh Gambar 4.18 dan kesalahan yang terjadi antara nilai target dan keluaran dari jaringan yang ditunjukkan oleh Gambar 4.19.



Gambar 4.18 Perbandingan Target Dengan Keluaran Jaringan Pada Metode FCM



Gambar 4.19 Selisih Antara Target Dengan Keluaran Jaringan

Berdasarkan Gambar 4.18 dapat dikatakan bahwa hasil prediksi harga penutupan oleh JST sangat bagus karena mampu mendekati nilai sebenarnya. Hal ini terlihat dari menyatunya grafik data target dan data prediksi. Selisih tertinggi yang terjadi antara data target dengan data prediksi yakni 0,006 yang ditunjukkan oleh Gambar 4.19.

BAB V PENUTUP

5.1 Kesimpulan

Berdasarkan hasil dan pembahasan, dapat disimpulkan:

1. Perbandingan proses *clustering*
 - a. Metode *K-Means*
Metode ini rata-rata menghasilkan nilai delta (*error* yang terjadi saat proses *clustering*) yang besar namun membutuhkan waktu yang relatif cepat dengan jumlah iterasi yang sedikit. Dari grafik fungsi objektif terhadap iterasi, metode ini sulit untuk mencapai konvergen.
 - b. Metode FCM
Metode ini rata-rata menghasilkan nilai delta yang sangat kecil dibandingkan metode *K-Means* namun membutuhkan waktu yang relatif lama dengan jumlah iterasi yang lebih besar.
2. Penerapkan metode FCM dan metode *Gradient Descent* pada algoritma "pembelajaran" jaringan RBF yaitu:
 - a. Perubahan bobot antara lapisan keluaran dengan lapisan tersembunyi

$$w_{ik}(t+1) = w_{ik}(t) + \alpha \delta_k h_i .$$

$$\text{di mana : } \delta_k = \sum_{k=1}^n (t_k - y_k) f'(y_{in_k})$$

- b. Perubahan bobot bias antara lapisan keluaran dengan lapisan tersembunyi

$$w_{ik}(t+1) = w_{ik}(t) + \alpha \delta_k .$$

- c. Perubahan pusat *cluster* μ_{ik}

$$\mu_{ik}(t+1) = \mu_{ik}(t) + \alpha \beta_i w_{ik} .$$

$$\text{di mana : } \beta_i = - \sum_{k=1}^n (t_k - y_k) \frac{(x_i - v_{ik})}{\sigma^2} \exp[-(x_i - v_{ik})^2 / 2\sigma^2]$$

d. Perubahan *spread* σ

$$\sigma(t+1) = \sigma(t) + \alpha \gamma_i w_{ik}.$$

$$\text{di mana : } \gamma_i = - \sum_{k=1}^n (t_k - y_k) \frac{(x_i - v_{ik})^2}{\sigma^3} \exp[-(x_i - v_{ik})^2 / 2\sigma^2]$$

3. Perbandingan model jaringan RBF

a. Metode *K-Means*

Metode ini rata-rata menghasilkan nilai MSE yang besar namun membutuhkan waktu yang lama. Untuk tahap simulasi metode ini mampu memprediksi harga penutupan pada data saham harian milik PT.Astra, Tbk tahun 2007 dengan MSE yakni 2,7300e-02.

b). Metode FCM

Metode ini mampu mencapai kriteria yang diberikan dengan nilai MSE yang sangat kecil dan waktu "pembelajaran" yang cepat. Dari hasil simulasi, metode ini mampu memprediksi harga penutupan data saham harian milik PT.Astra, Tbk tahun 2007 dengan MSE yakni 8,1063e-007.

5.2 Saran

Untuk pengembangan yang lebih baik, saran yang dapat diberikan pada skripsi ini adalah metode *Gradient Descent* yang digunakan dalam skripsi ini, bisa dikembangkan dengan metode yang lebih baik seperti metode *Quasi Newton*.

DAFTAR PUSTAKA

- Anam, S. 2005. *Analisis Sefalometri Menggunakan Jaringan Fungsi Basis Radial*. ITS Surabaya.
- Anton, H. 2004. *Aljabar Linear Elementer*. Erlangga. Jakarta
- Babuska. 2005. *Fuzzy Clustering*. (Online). (<http://www.fuzzy-clustering.de/clustering.html>).
- Bezdek, J.C. 1981. *Pattern Recognition with Fuzzy Objective Function Algorithms*. Plenum Press. New York.
- Boyd, S dan Lieven, V. 2004. *Convex Optimization*. Cambridge University Press. United Kingdom.
- Bronstein, I.N. dkk. 2007. *Handbook of Mathematics*. Springer. Berlin.
- Buchanan J.L. dan Peter, R.T. 1992. *Numerical Methods and Analysis*. Mc. Graw-Hill. Singapore.
- Fausett, L. 1994. *Fundamental of Neural Networks (Architectures, Algorithms, and Applications)*. Prentice-Hall. New Jersey.
- Gelley, N. dan Jang, R. 2000. *Fuzzy Logic Toolbox*. Mathwork, Inc. USA.
- Harjoko, A., dkk. 2006. *Fuzzy Multi Attribute Decision Making*. Graha Ilmu. Yogyakarta.
- Haykin, S. 1994. *Neural Networks (A Comprehensive Foundation)*. Macmillan College Publishing Company. New York.
- Kaya, M dan Alhadj, R. 2006. Utilizing Genetic Algorithms to Optimize Membership Functions for Fuzzy Weighted Association Rules Mining. *Applied Intelligence*. Springer-Verlag New York, Inc. USA.
- Leader, J.J. 2004. *International Edition Numerical Analysis and Scientific Computation*. Pearson Education, Inc. USA.
- Mathews, J.H dan Kurtis, D.F. 2004. *International edition Numerical Methods Using Matlab Fourth Edition*. Pearson Education International. USA.

- Nocedal, J. dan Stephen, J. W. 1999. *Numerical Optimization*. Springer-Verlag New York, Inc. USA.
- Pal, N.P., dkk. 2005. A possibilistic Fuzzy C-Means Clustering Algorithm. *IEEE Transactions on Fuzzy Systems* (Vol.13 No.4).
- Puspitaningrum, D. 2006. *Pengantar Jaringan Saraf Tiruan*. Andi. Yogyakarta.
- Ross, T.J. 2005. *Fuzzy Logic with Engineering Applications*. Edisi ke-2. John Willey & Sons Inc. Inggris.
- Santosa, B. 2007. *Data Mining Teknik Pemanfaatan Data Untuk Keperluan Bisnis*. Graha Ilmu. Yogyakarta.
- Shihab, A.I. 2000. *Fuzzy Clustering Algorithms and Their Application to Medical Image Analysis*. University Of London.
- Siang, J.H. 2005. *Jaringan Syaraf Tiruan & Pemrogramannya Menggunakan Matlab*. Andi. Yogyakarta.
- Soemantri, R. 1988. *Materi Pokok Analisis Real I*. Karunika Jakarta. Jakarta.
- Teknomo, K. 2007. *K-Means Clustering Tutorial*. (Online). (<http://people.revoledu.com/kardi/tutorial/kMean.html>).
- Walpole, R.E. 1995. *Pengantar Statistika Edisi ketiga*. Gramedia Pustaka Utama. Jakarta.
- Yan, J., dkk. 1994. *Using Fuzzy Logic Towards Intelligent Systems*. Prentice Hall. New York.
- Zimmermann. 1991. *Fuzzy sets Theory and Its Applications*. Edisi 2. Kluwer Academic Publishers. Massachusetts.

Lampiran 1:

Nilai bobot dan pusat *cluster* dari hasil pembelajaran metode *K-Means* dengan model 5-5-1.

V= Kolom 1 hingga 5

4.2940	4.3714	4.2990	2.5368	4.3322
0.3271	0.3134	0.3450	-0.1914	0.3630
-1.0975	-1.1121	-1.1002	-0.9855	-1.1268
6.1580	6.1309	6.1316	3.4033	6.0931
-3.1227	-3.0901	-3.1374	-0.6069	-3.1040

W= 2.6466
0.4008
1.9441
2.1740
-3.5386

Nilai bobot dan pusat *cluster* dari hasil pembelajaran metode *Fuzzy C-Means* dengan model 5-5-1.

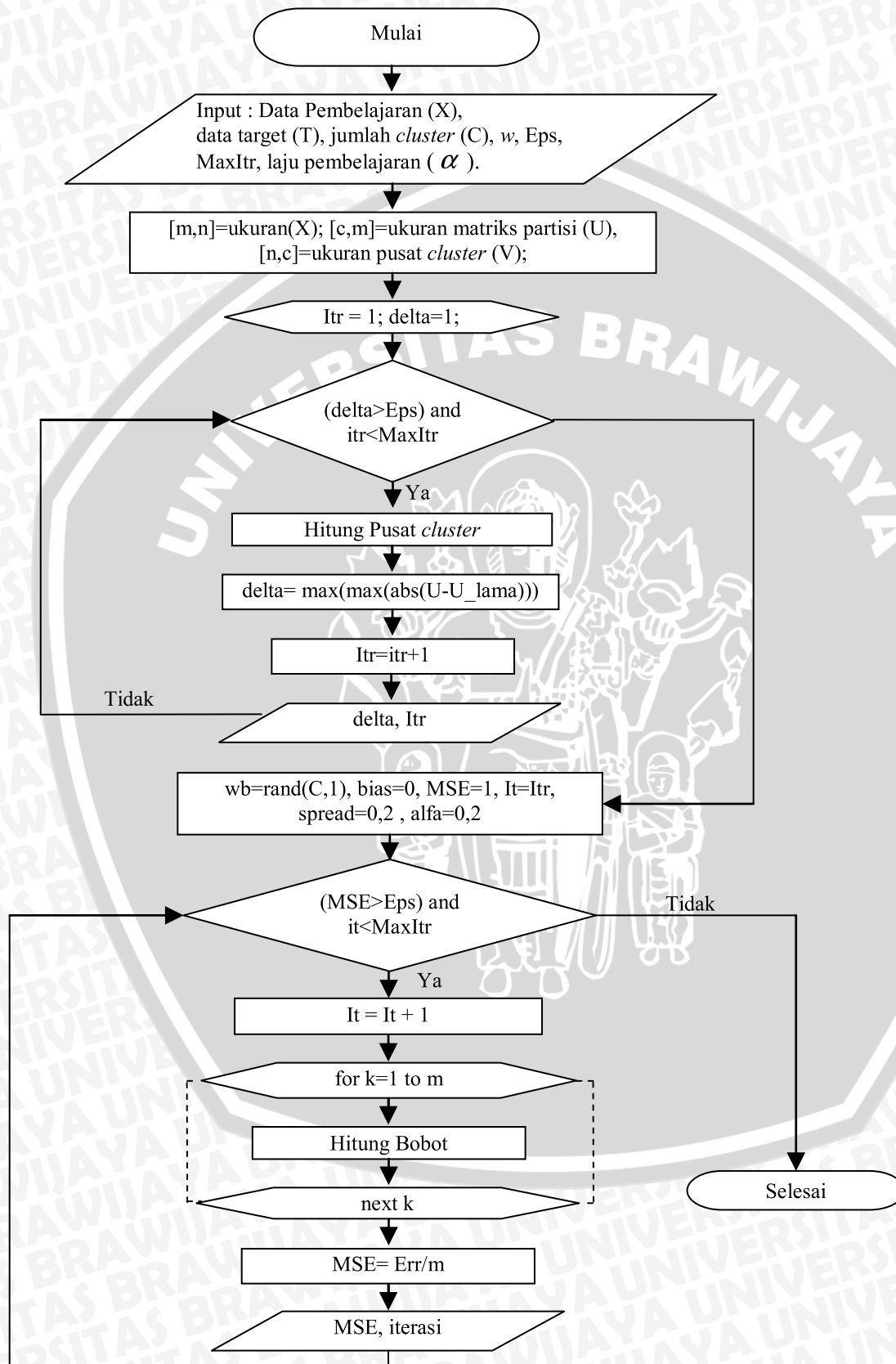
V=Kolom 1 hingga 5

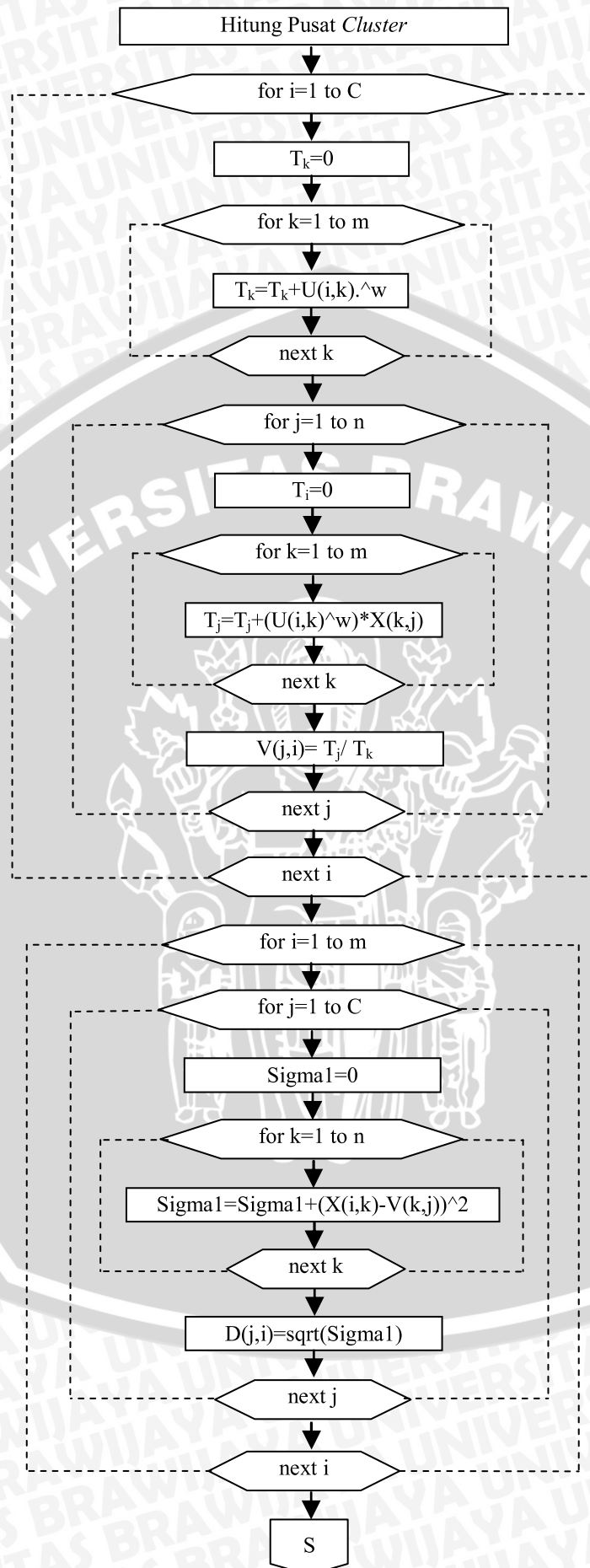
-0.8304	2.2579	-0.8813	0.4733	-0.2201
-0.8280	-2.2742	-0.8256	0.4537	-0.2414
-0.8522	-2.2413	-0.8843	0.4931	-0.2071
0.1475	-0.1787	2.2456	-0.3881	-0.5034
-0.8572	2.2399	-0.8415	0.4921	-0.2325

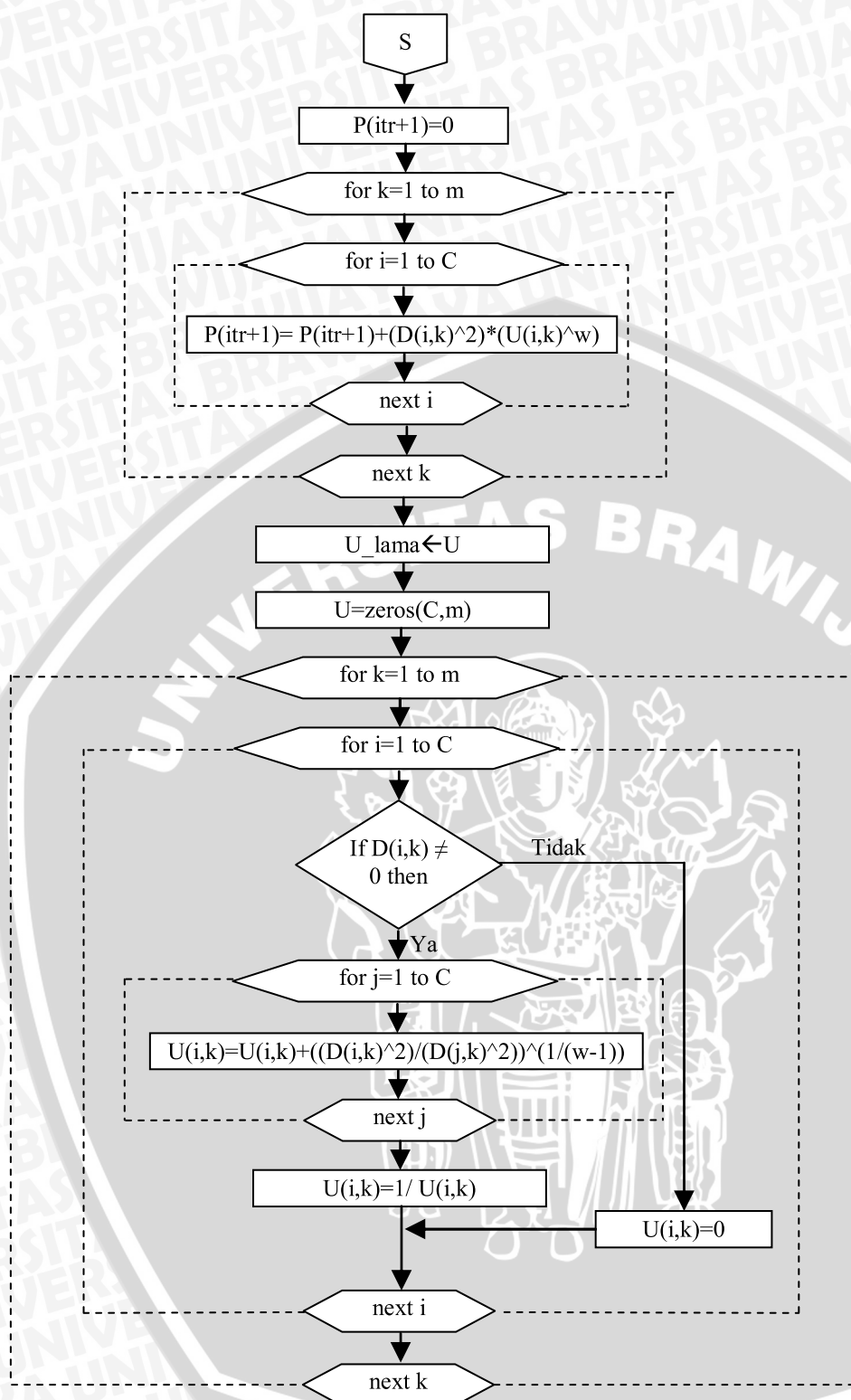
W = 4.3835
-2.2986
-1.5569
2.1098
-4.2671

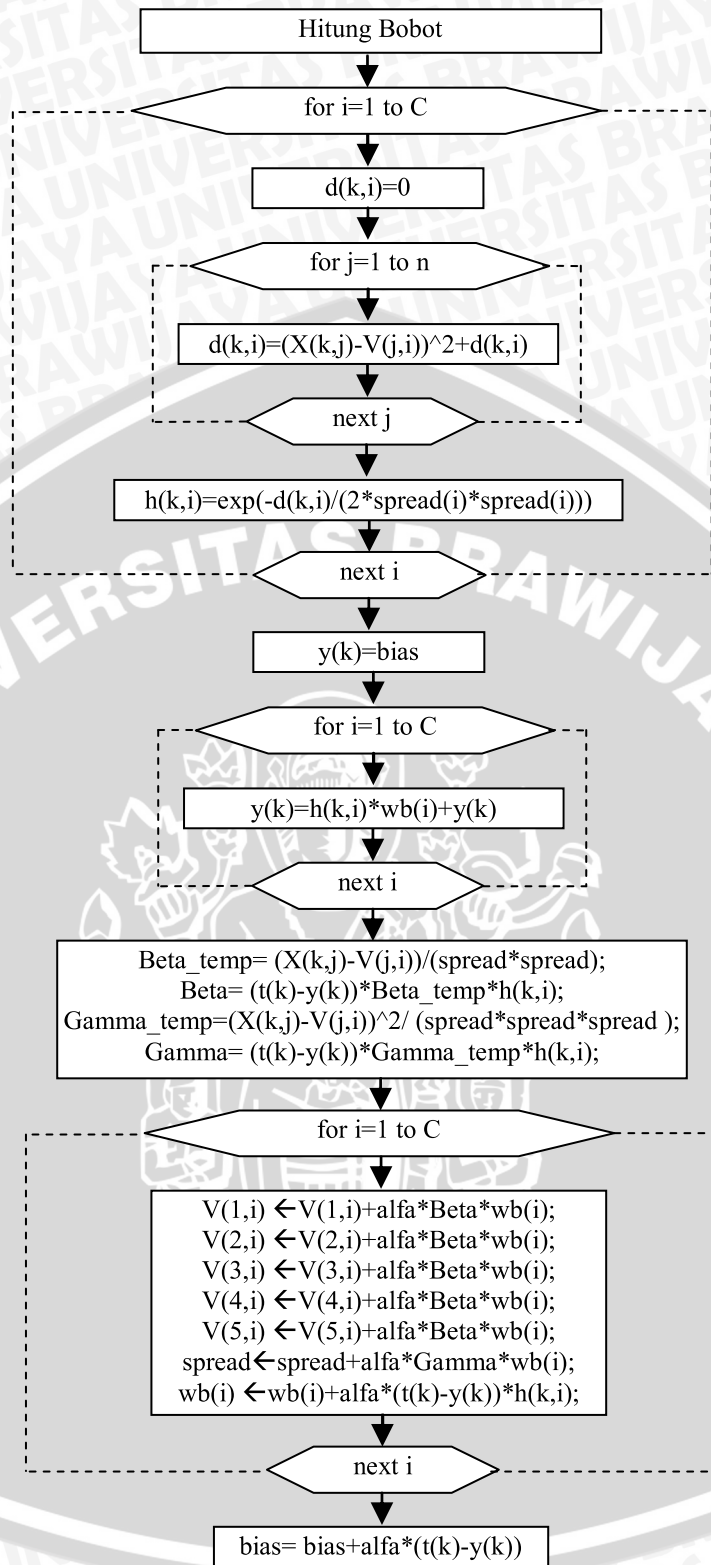
Lampiran 2:

Diagram alir model jaringan RBF dengan metode *Fuzzy C-Means*









Lampiran 3 : Data Perdagangan Harian Saham tentang Harga Sebelum, Harga Tertinggi, Harga Terendah, Harga Penutupan, Volume Penjualan dan *Stock Index* pada PT.Astra, Tbk tahun 2008.

Date	Open	High	Low	Volume	Adj Close	Close
01/02/07	16050	16350	16050	2934500	16060.88	16350
01/03/07	16250	16500	16250	3969000	16110	16400
01/04/07	16450	16600	16400	3068000	16208.23	16500
01/05/07	16400	16900	16400	2759000	16453.81	16750
01/08/07	16700	16700	16100	3695500	15864.42	16150
01/09/07	16600	16600	15650	4945500	15422.38	15700
01/10/07	15500	15500	14950	11890500	14783.87	15050
01/11/07	15300	15400	14800	5099500	14587.41	14850
01/12/07	15000	15100	14600	9370500	14734.76	15000
01/15/07	15200	15450	15100	6209000	15029.45	15300
01/16/07	15500	15650	15400	4071500	15324.15	15600
01/17/07	15650	15650	15400	3017000	15127.68	15400
01/18/07	15500	15700	15500	4771000	15373.26	15650
01/19/07	15650	16350	15600	9300000	15815.31	16100
01/22/07	16150	16250	15950	5279500	15766.19	16050
01/23/07	16050	16050	15700	6896000	15520.61	15800

Date	Open	High	Low	Volume	Adj Close	Close
01/24/07	16000	16000	15800	1800500	15520.61	15800
01/25/07	15800	15800	15450	6221500	15275.03	15550
01/26/07	15400	15400	15050	3943500	15127.68	15400
01/29/07	15300	15500	15050	2961000	14980.34	15250
01/30/07	15300	15300	14800	5168500	14734.76	15000
01/31/07	15000	15050	14700	4718000	14587.41	14850
02/01/07	14850	15050	14550	6100500	14292.71	14550
02/02/07	14650	15150	14550	13834500	14832.99	15100
02/05/07	15100	15150	14550	3746000	14341.83	14600
02/06/07	14600	14650	14400	21639500	14194.48	14450
02/07/07	14400	15050	14250	15501000	14636.52	14900
02/08/07	14850	15050	14600	9332000	14489.18	14750
02/09/07	14500	15200	14500	4438500	14882.1	15150
02/12/07	15100	15500	15100	7666000	14832.99	15100
02/13/07	15200	15200	14600	1562500	14538.29	14800
02/14/07	14950	15000	14850	2237500	14636.52	14900
02/15/07	15200	15200	14850	4782000	14685.64	14950
02/16/07	15000	15150	14900	2375000	14734.76	15000

Date	Open	High	Low	Volume	Adj Close	Close
02/20/07	15100	15100	15050	216500	14832.99	15100
02/21/07	15100	15200	14800	4138000	14538.29	14800
02/22/07	15000	15000	14850	3774000	14685.64	14950
02/23/07	14950	14950	14200	6981500	14096.25	14350
02/26/07	14350	14350	13900	5631000	13752.44	14000
02/27/07	14100	14250	13750	9109500	13654.21	13900
02/28/07	13300	14050	12800	10805000	13801.56	14050
03/01/07	14050	14200	13600	5818500	13555.98	13800
03/02/07	13800	14000	13450	8128500	13752.44	14000
03/05/07	13550	13800	13500	7023000	13359.51	13600
03/06/07	13650	14050	13650	7428500	13605.09	13850
03/07/07	14000	14250	13950	4127500	13899.79	14150
03/08/07	14150	14500	14000	3538500	14243.6	14500
03/09/07	14600	14600	14250	6365500	14194.48	14450
03/12/07	14550	14750	14550	4169500	14390.95	14650
03/13/07	14600	14650	14250	6069000	13998.02	14250
03/14/07	14000	14350	13850	8565500	13654.21	13900
03/15/07	14100	14200	13650	6950500	13408.63	13650

Date	Open	High	Low	Volume	Adj Close	Close
03/16/07	13650	13650	13300	5801500	13212.16	13450
03/20/07	13500	13550	13000	11459500	12868.35	13100
03/21/07	13200	13350	13050	10233000	13064.82	13300
03/22/07	13500	13650	13450	8881500	13310.4	13550
03/23/07	13600	13600	13050	14274000	12966.59	13200
03/26/07	13150	13300	13050	5353500	12917.47	13150
03/27/07	13150	13250	13050	9695000	12819.24	13050
03/28/07	13050	13050	12500	21210000	12426.31	12650
03/29/07	12550	13300	12500	13203500	13064.82	13300
03/30/07	13400	13550	13150	12369500	12966.59	13200
04/02/07	13200	13250	13050	10222000	12966.59	13200
04/03/07	13250	13800	13250	16152000	13555.98	13800
04/04/07	14000	14300	13900	23097500	13703.32	13950
04/05/07	13850	13900	13450	8987000	13261.28	13500
04/09/07	13700	13800	13600	2951500	13457.74	13700
04/10/07	13700	13700	13450	5108000	13310.4	13550
04/11/07	13600	13850	13600	5200000	13457.74	13700
04/12/07	13600	13650	13450	4701500	13310.4	13550

Date	Open	High	Low	Volume	Adj Close	Close
04/13/07	13550	13750	13450	3842000	13261.28	13500
04/16/07	13550	13600	13400	11583500	13310.4	13550
04/17/07	13600	13900	13600	7946000	13555.98	13800
04/18/07	13850	14400	13850	25222500	14047.13	14300
04/19/07	14200	14350	13750	19523000	13555.98	13800
04/20/07	14000	14150	13900	8410000	13752.44	14000
04/23/07	14200	14300	13900	6732000	13801.56	14050
04/24/07	14000	14250	14000	10664000	13948.9	14200
04/26/07	14700	15050	14700	23010000	14685.64	14950
04/27/07	14950	14950	14800	5767500	14636.52	14900
04/30/07	14900	14900	14350	8659000	14145.37	14400
05/01/07	14400	14800	14400	3629500	14489.18	14750
05/02/07	14900	15150	14850	13424000	14685.64	14950
05/03/07	15000	15150	15000	6833500	14832.99	15100
05/04/07	15300	15800	15300	13493500	15422.38	15700
05/07/07	15750	15950	15700	6088000	15569.73	15850
05/08/07	15850	15900	15700	12126500	15471.49	15750
05/09/07	15650	15850	15650	4984000	15520.61	15800

Date	Open	High	Low	Volume	Adj Close	Close
05/10/07	15900	16250	15900	14459500	15913.54	16200
05/11/07	16000	16200	15900	8482500	15667.96	15950
05/14/07	16100	16300	15850	3149000	15717.07	16000
05/15/07	16000	16000	15550	6513000	15324.15	15600
05/16/07	15550	16000	15550	3508000	15618.84	15900
05/21/07	16000	16200	15950	7323500	15864.42	16150
05/22/07	16100	16200	15950	3749000	15913.54	16200
05/23/07	16200	16850	16200	14376500	16552.04	16850
05/24/07	16800	16800	16500	1796000	16257.35	16550
05/25/07	16350	16350	16050	3968500	15864.42	16150
05/28/07	16250	16550	16100	2786000	16011.77	16300
05/29/07	16300	16400	16250	2665000	16011.77	16300
05/30/07	16250	16250	16100	2515500	15815.31	16100
05/31/07	16250	16500	16250	2868000	16110	16400
06/04/07	16500	16850	16500	4990000	16306.46	16600
06/05/07	16600	16900	16600	2388500	16453.81	16750
06/06/07	16750	16750	16550	2902500	16453.81	16750
06/07/07	16500	16500	16200	4117000	15962.65	16250

Date	Open	High	Low	Volume	Adj Close	Close
06/08/07	16000	16000	15850	6846000	15569.73	15850
06/11/07	15950	16100	15950	1292500	15717.07	16000
06/12/07	16100	16500	16100	3973000	16208.23	16500
06/13/07	16200	16250	16000	3029000	15864.42	16150
06/14/07	16250	16450	16250	4638500	16110	16400
06/15/07	16500	16500	16300	1506000	16500	16500
06/18/07	16500	16500	16300	2743000	16400	16400
06/19/07	16400	16500	16300	2405500	16450	16450
06/20/07	16450	16700	16450	6059500	16700	16700
06/21/07	16600	16650	16450	3973000	16550	16550
06/22/07	16500	16950	16450	2643000	16900	16900
06/25/07	16800	16800	16450	3237000	16750	16750
06/26/07	16750	17300	16700	7697500	17100	17100
06/27/07	17000	17000	16650	10190500	16700	16700
06/28/07	16800	16800	16500	5472000	16700	16700
06/29/07	16800	17000	16600	8284000	16900	16900
07/02/07	16900	17100	16900	4397000	17050	17050
07/03/07	17100	17250	17000	5291000	17000	17000

Date	Open	High	Low	Volume	Adj Close	Close
07/04/07	17000	17150	16900	3063500	16950	16950
07/05/07	17000	17100	17000	4776000	17000	17000
07/06/07	17000	18000	17000	11341000	18000	18000
07/09/07	18100	20050	18100	10198500	19150	19150
07/10/07	19200	19250	19000	8364500	19150	19150
07/11/07	19000	19000	18250	5715000	18350	18350
07/12/07	18500	18600	18100	2709000	18100	18100
07/13/07	18300	18500	17850	2862500	18000	18000
07/17/07	18050	18800	18050	11559000	18600	18600
07/18/07	18500	18600	18000	9313000	18600	18600
07/19/07	18300	18850	18300	6377000	18700	18700
07/20/07	18800	19300	18800	3931000	19250	19250
07/23/07	18900	19100	18900	2662500	19050	19050
07/24/07	19200	19600	19200	5439500	19450	19450
07/25/07	19050	19400	18850	7219500	18850	18850
07/26/07	19000	19050	18750	4836500	18850	18850
07/27/07	18300	18500	17900	8960000	18150	18150
07/31/07	18400	18750	18350	2905500	18750	18750

Date	Open	High	Low	Volume	Adj Close	Close
08/01/07	18600	18600	17800	8741500	18250	18250
08/02/07	18350	18750	18100	4007000	18500	18500
08/03/07	18600	18700	18350	2542500	18450	18450
08/06/07	18000	18000	17300	9288500	17400	17400
08/07/07	17600	17800	17000	6393500	17050	17050
08/08/07	17200	17800	17200	4929000	17800	17800
08/09/07	18100	18350	17750	3377500	17750	17750
08/10/07	17250	17450	17000	5338500	17350	17350
08/14/07	17250	17250	16950	7888500	17000	17000
08/15/07	16800	16800	15750	8908000	16250	16250
08/16/07	15600	15600	14800	10375000	15450	15450
08/20/07	16000	16400	15900	11135000	16200	16200
08/21/07	16200	16400	15900	8577500	16000	16000
08/22/07	16150	16500	15950	5714000	16500	16500
08/23/07	16700	17400	16700	7270500	17350	17350
08/24/07	17250	17400	17000	2761000	17300	17300
08/27/07	17500	17850	17400	2740500	17650	17650
08/28/07	17650	17650	17200	4477500	17400	17400

Date	Open	High	Low	Volume	Adj Close	Close
08/29/07	17000	17200	16700	7337000	17200	17200
08/30/07	17400	17550	17250	4083500	17450	17450
08/31/07	17500	17900	17450	3408000	17850	17850
09/03/07	17850	18050	17850	3201000	17900	17900
09/04/07	17900	18150	17900	6043500	18100	18100
09/05/07	18150	18150	17850	1330500	18000	18000
09/06/07	17850	18050	17800	5373000	17950	17950
09/07/07	17950	18000	17900	589000	18000	18000
09/10/07	17800	17900	17700	11980000	17800	17800
09/11/07	17800	17950	17700	2731500	17800	17800
09/12/07	17950	18000	17900	5907000	17950	17950
09/13/07	17950	18000	17900	845500	18000	18000
09/14/07	18000	18000	17950	4294500	18000	18000
09/17/07	18000	18750	18000	12077000	18500	18500
09/18/07	18400	18450	18250	4207000	18400	18400
09/19/07	19100	19250	18700	8141500	19150	19150
09/20/07	19250	19450	19100	3870000	19100	19100
09/21/07	18900	19450	18850	2515500	19300	19300

Date	Open	High	Low	Volume	Adj Close	Close
09/24/07	19350	19600	19300	3326000	19500	19500
09/25/07	19500	19500	19000	3739000	19000	19000
09/26/07	19100	19450	19100	2898500	19400	19400
09/27/07	19550	19600	19350	2406000	19500	19500
09/28/07	19500	19600	19250	2442500	19250	19250
10/01/07	19250	19400	19150	2934000	19300	19300
10/02/07	19400	20100	19300	10052500	20100	20100
10/03/07	20200	21200	20200	6201500	20450	20450
10/04/07	20250	20850	20250	4766000	20800	20800
10/05/07	20900	21600	20900	7297000	21000	21000
10/08/07	21100	21400	20350	4335000	20350	20350
10/09/07	20400	21350	20400	7414500	21250	21250
10/10/07	21400	22350	21400	5970000	22200	22200
10/11/07	22350	23050	22350	6034500	22500	22500
10/17/07	22500	23700	22400	3876000	23600	23600
10/18/07	23600	23600	22950	3446000	22950	22950
10/19/07	22900	23250	22500	2874000	22500	22500
10/22/07	21500	21950	21000	3116000	21550	21550

Date	Open	High	Low	Volume	Adj Close	Close
10/23/07	21900	23200	21800	6454000	22800	22800
10/24/07	23000	23300	22200	4824000	22200	22200
10/25/07	22400	24700	22400	6073000	24500	24500
10/26/07	24600	26000	24400	5948500	25500	25500
10/29/07	25700	26700	25700	1800000	26500	26500
10/30/07	26500	26800	26000	4224000	26650	26650
10/31/07	26500	26500	24900	8316000	25600	25600
11/01/07	26000	27250	26000	8708000	26550	26550
11/02/07	26000	27050	25700	3595500	26550	26550
11/05/07	26400	26550	24900	4174000	25150	25150
11/06/07	25150	25150	23800	11287000	24350	24350
11/07/07	24500	25250	24500	8047000	25100	25100
11/08/07	24800	24900	24300	3036500	24400	24400
11/09/07	24400	24400	24050	1567500	24350	24350
11/12/07	24000	24000	22750	7268000	23250	23250
11/13/07	23100	23150	22750	7626000	23000	23000
11/14/07	23500	24200	23500	8688500	24000	24000
11/15/07	23800	24200	23750	4360000	24050	24050

Date	Open	High	Low	Volume	Adj Close	Close
11/16/07	23700	24050	23350	1917000	23550	23550
11/19/07	23600	23900	23600	3082000	23850	23850
11/20/07	23000	24300	22850	5390000	24050	24050
11/21/07	24000	24000	22900	4798000	23000	23000
11/22/07	22800	22800	21800	5808000	22400	22400
11/23/07	22400	23200	22200	2515500	23200	23200
11/26/07	23500	24650	23400	4734000	24500	24500
11/27/07	24200	24550	23800	3423000	24350	24350
11/28/07	24350	25150	24350	7617500	24850	24850
11/29/07	25500	26400	25500	7661500	25800	25800
11/30/07	26200	26750	25000	4212000	25000	25000
12/03/07	25000	26950	24900	2652500	26800	26800
12/04/07	26800	28100	26800	4198500	27500	27500
12/05/07	27500	27500	26450	7979500	26700	26700
12/06/07	27000	27050	26250	4458500	26450	26450
12/07/07	27000	27100	26450	5175500	26850	26850
12/10/07	26850	28750	26850	8137000	28500	28500
12/11/07	28700	28750	27650	3318500	27750	27750

Date	Open	High	Low	Volume	Adj Close	Close
12/12/07	27300	28800	27250	5199000	28350	28350
12/13/07	28000	28000	27400	3166000	27450	27450
12/14/07	27500	27800	27300	2844500	27450	27450
12/17/07	27300	27350	26500	4147500	26550	26550
12/18/07	26400	26400	25500	7116000	25750	25750
12/19/07	26050	26400	26050	5073500	26100	26100
12/26/07	26700	26700	26100	4048000	26300	26300
12/27/07	26500	26800	26250	3012000	26800	26800
12/28/07	26800	27300	26500	3210500	27300	27300



